

CPSI10003 MICROCOMPUTER APPLICATIONS 1 - MICROSOFT ACCESS



LueCrecy Ragan
Arkansas State University-Beebe

CPSI10003 Microcomputer Applications 1 - Microsoft Access

This text is disseminated via the Open Education Resource (OER) LibreTexts Project (<https://LibreTexts.org>) and like the thousands of other texts available within this powerful platform, it is freely available for reading, printing, and "consuming."

The LibreTexts mission is to bring together students, faculty, and scholars in a collaborative effort to provide an accessible, and comprehensive platform that empowers our community to develop, curate, adapt, and adopt openly licensed resources and technologies; through these efforts we can reduce the financial burden born from traditional educational resource costs, ensuring education is more accessible for students and communities worldwide.

Most, but not all, pages in the library have licenses that may allow individuals to make changes, save, and print this book. Carefully consult the applicable license(s) before pursuing such effects. Instructors can adopt existing LibreTexts texts or Remix them to quickly build course-specific resources to meet the needs of their students. Unlike traditional textbooks, LibreTexts' web based origins allow powerful integration of advanced features and new technologies to support learning.



LibreTexts is the adaptable, user-friendly non-profit open education resource platform that educators trust for creating, customizing, and sharing accessible, interactive textbooks, adaptive homework, and ancillary materials. We collaborate with individuals and organizations to champion open education initiatives, support institutional publishing programs, drive curriculum development projects, and more.

The LibreTexts libraries are Powered by [NICE CXone Expert](#) and was supported by the Department of Education Open Textbook Pilot Project, the California Education Learning Lab, the UC Davis Office of the Provost, the UC Davis Library, the California State University Affordable Learning Solutions Program, and Merlot. This material is based upon work supported by the National Science Foundation under Grant No. 1246120, 1525057, and 1413739.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation nor the US Department of Education.

Have questions or comments? For information about adoptions or adaptations contact info@LibreTexts.org or visit our main website at <https://LibreTexts.org>.

This text was compiled on 06/12/2026

TABLE OF CONTENTS

Licensing

1: Microsoft Access - Setting a Foundation

- 1.1: Microsoft Access - Setting a Foundation Sections
 - 1.1.1: Introduction to Microsoft Access
 - 1.1.2: Microsoft Access- Main Features and Navigation
 - 1.1.3: Database Design and Four Main Objects
 - 1.1.4: Creating a Table
 - 1.1.5: Primary Keys and Table Relationships
 - 1.1.6: Maintaining Records in a Database
 - 1.1.7: Entering and Editing Records
 - 1.1.8: Querying a Database
 - 1.1.9: All About Queries

2: Microsoft Access - Building Proficiency

- 2.1: Microsoft Access - Building Proficiency Sections
 - 2.1.1: Microsoft Access and Data
 - 2.1.2: Advanced Queries in Microsoft Access
 - 2.1.3: Creating Forms in Microsoft Access
 - 2.1.4: Form Basics
 - 2.1.5: Customizing Forms
 - 2.1.6: Multiple Table Forms
 - 2.1.7: Creating Reports in Microsoft Access
 - 2.1.8: Customizing Reports
 - 2.1.9: Using Macros
 - 2.1.10: Microsoft Access Terminology

Index

Glossary

Detailed Licensing

Detailed Licensing

Licensing

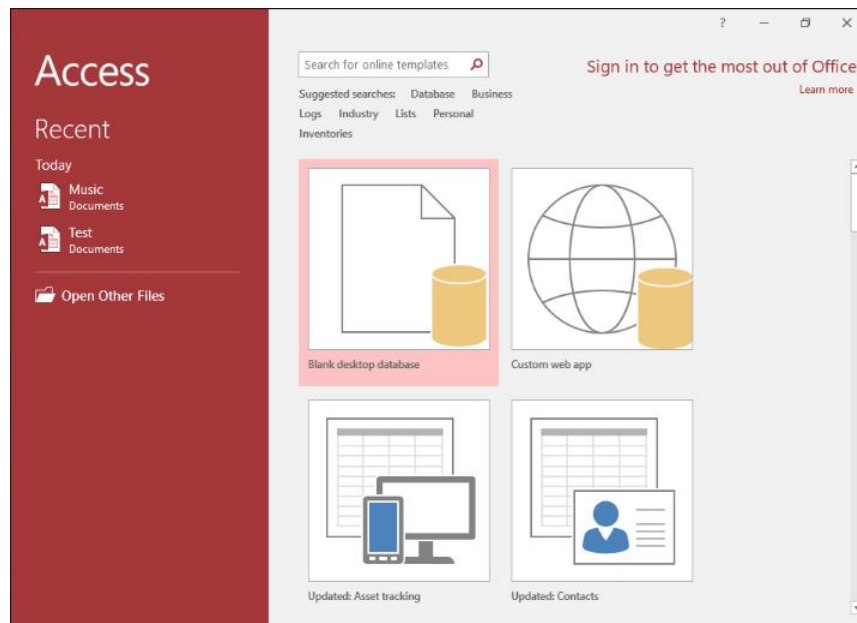
A detailed breakdown of this resource's licensing can be found in [Back Matter/Detailed Licensing](#).

CHAPTER OVERVIEW

1: Microsoft Access - Setting a Foundation

Learning Objectives

- Introduction to Microsoft Access
- Microsoft Access - Main Features and Navigation
- Database Design and the Four Main Objects in a Database
- Creating a Table
- Primary Keys and Table Relationships
- Maintaining Records in a Database
- Entering and Editing Records
- Querying a Database
- All About Queries



1.1: Microsoft Access - Setting a Foundation Sections

- 1.1.1: Introduction to Microsoft Access
- 1.1.2: Microsoft Access- Main Features and Navigation
- 1.1.3: Database Design and Four Main Objects
- 1.1.4: Creating a Table
- 1.1.5: Primary Keys and Table Relationships
- 1.1.6: Maintaining Records in a Database
- 1.1.7: Entering and Editing Records
- 1.1.8: Querying a Database
- 1.1.9: All About Queries

This page titled [1: Microsoft Access - Setting a Foundation](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [LueCrey Ragan - Arkansas State University - Beebe](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

1.1: Microsoft Access - Setting a Foundation Sections



Microsoft Access - Setting a Foundation

This chapter introduces Microsoft Access as an easy tool for organizing and analyzing data. It explains different types of databases and how Access connects with programs like Excel. The chapter covers basic design ideas, the four main parts (tables, forms, queries, and reports), and how to create and organize data using keys. It also shows how to enter, manage, and analyze data using different types of queries.

This page titled [1.1: Microsoft Access - Setting a Foundation Sections](#) is shared under a [CC BY-NC-SA](#) license and was authored, remixed, and/or curated by [LueCrecy Ragan - Arkansas State University - Beebe](#).

1.1.1: Introduction to Microsoft Access

[Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

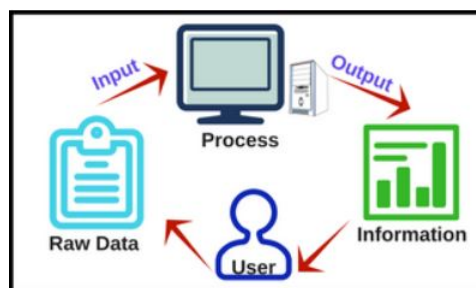
Access Scenario

Welcome to the Business Analysis Department at WorldCorp. We are happy you will be working here. Your role as an analyst is to work with data professionals who support various teams within WorldCorp. Most of the time, you will be tasked with supporting the sales and marketing team, though you may also work with data from other areas of the company. To get started, you will learn what a database is and how to use its data, maintain its records, and create data reports. We focus on Microsoft Access in this chapter, but other database programs have similar features and functionality.



Figure 13.1: Warehouses like this one require complex systems to manage their inventory. Microsoft Access is designed to handle and report on large amounts of data. (credit: modification of “Man Standing in an Aisle of a Warehouse Carrying a Box” by Tiger Lily/Pexels, CC0)

A **database** is a collection of related data that is organized so that it can be easily access, managed and updated. Databases can store details about people, products, orders, or anything else. **Data** can be defined as facts and statistics about people, objects, events and ideas. Organizations interpret, process and analyze data to turn the data into useful **information**. This information is used to make better decision-making.



Data process

In the past, many databases were paper-based. Paper records were stored in files within file cabinets. File cabinets still exist today, in large part due to the fears associated with the insecurity of electronic records being compromised. The individual documents (records) are categorized into different folders (tables) to streamline user access. However, this process is labor-intensive, susceptible to errors, and an office eyesore!

Databases are used for two major purposes: for processing transactions and analytical research. Transactional databases track daily operations. This data can be extracted into other software programs for further manipulation, summation and classification. This allows users to perform analytical research and prepare reports to help create strategic information.

Computerized databases organize data into rows, columns and tables, and is indexed to make it easier to find things quickly. Data gets updated, expanded and deleted as new data is added. Automated processes quickly manipulate the data, and data accuracy is enhanced through automated rules. The advancements in database software technologies have contributed to the Information Age that currently exists. The growth of the filing cabinet market cannot compare to the growth of electronic data contained in private and public networks, and web sites throughout the Internet!

While a spreadsheet can function as a database, with its utilization of columns and rows, the flat-file structure creates redundancies and inconsistencies which limit the ability to find and understand the data. In contrast, a relational database uses a tabular structure so that data can be easily reorganized and accessed in a number of different ways. Relational databases are made up of a set of tables with data that fits into predefined categories (fields). Each table has at least one data category (field) in a column, and each row has a certain data instance for the categories (fields) which are defined in the columns.

Microsoft's relational database software program available in Office 365 is called Access. Access only functions with Windows operating systems (not available on a Mac O/S), and is not available in all Office versions. Access is a tool to organize, store, and manipulate data, in order to report on the data stored in tables. The power of a database comes from the ability to link tables together to provide data processing efficiencies for large sets of data. Access' integration with the other Microsoft Office programs makes it a very popular and economical relational database solution.

Database Explained

Learning Objectives

By the end of this section, you will be able to:

- Discuss how databases are used in organizations
- Compare and contrast the features of a relational database system with other types of databases and data storage
- Describe the objects in a relational database
- Evaluate applications of relational database management systems in organizations

Data—bits of information about people, places, and things—is a key part of the world of business and technology. It can take the form of words or numbers and is frequently collected without active individual involvement. For example, the internet is constantly collecting information on what consumers look at, click on, and read. This information is then often passed on to a database or even parsed out to several databases. Other technology, including sensors and the so-called Internet of Things, is also constantly recording and storing data for businesses to use in a myriad of ways.

One of the challenges of managing and analyzing data is understanding what data is, what it can be, and how it can transform the world around us. Data on its own is not valuable or actionable. To make a difference, data needs to be processed and organized to give it context. When this occurs, we have information. Then, knowledge is generated from data that has been transformed into information and then finally analyzed in a way to give meaning and draw conclusions about the data.

No matter how much data is collected, it is useless without organization and structure. Think about all the things you interact with in a typical day. If those things were collected but had no structure (imagine bits of paper with single words or numbers thrown into a pile), you would not be able to make sense of them. Since the value of data lies in its ability to inform decisions and help us understand the world, both past and present, the structure and storage of data are critical.

A database stores a large amount of data in an effective way that will allow for efficient retrieval in a computerized environment. Databases may be physical objects, such as address books or file folders, each of which provides a place to keep data in an organized fashion so we can quickly locate what we need. Physical storage has limits, however, because its structure is easily overwhelmed when datasets get too large, and retrieval of meaningful information becomes impossible.

A digital database is a computerized file that not only stores data in a structured or organized way but also allows users to make changes to transform data. Digital databases are often contained within a program such as Access on a desktop computer or Microsoft Azure or an structured query language (SQL) server at the companywide level. Digital databases add new features and capabilities to business decision-making by combining the power of analytics with the structure of tables and rows.

Mac Tip

Access is only available on desktops and laptops with a Windows operating system. It is not available on Mac, Chrome, or Linux devices. Mac has a comparable desktop/laptop database called FileMaker.

Digital databases provide structure, organization, and access. But unlike physical objects, they provide a flexible structure that enables us to capture, store, and organize large, complicated sets of data, and that also gives us options for retrieving and using information. Using digital databases enables users to transform data, creating new ways of informing decisions. Because businesses must make decisions based on data every day, the key to good decision-making is creating a database that contains accessible, organized, actionable data.

The Role of Databases in an Organization

Databases drive nearly all business functions in organizations throughout the world. Any business that uses the internet has a digital platform or foundational technology that is built on databases. A digital platform is a place where users can interact with and explore the products and services offered by a company. Digital platforms often come in the form of an app, website, or social media. The term *database* is ubiquitous in organizations, where you can assume that every function and every employee or customer are interacting with many databases.

Organizations use databases to store information about customers, employees, vendors, products, supply chain, transportation, maintenance, sales, inventory, and much more. Anywhere an organization is working with data, some type of database will be involved. Often the database is the foundation for the work, and the people who interact with the database may not even be aware that it is there.

Data in organizations comes from a variety of inputs, such as from direct input by employees or customers. Some examples include entering new customer information, recording human resources information, creating maintenance requests, purchasing goods for company use, and engaging in sales transactions. Data can also come from technology that records information related to the business—for example, technology that tracks items such as inventory levels, building management (temperature, air quality, occupancy), traffic patterns (employee use of resources), resource use by customers, and customer contact (frequency and method of contact, as well as the results).

Sales and marketing departments collect a great deal of data about their operations and customers. In addition to typical customer information and sales history, a business is also likely to look at daily performance data on marketing and sales operations. This can include sales force performance, customer service, and experience data; social media promotion; and web platform performance. Automatic data collection tends to generate a very large quantity of data that needs to be connected (or related) and then aggregated (or summarized) to be understandable and useful.

Businesses also capture public data from other sources. For example, sales and marketing departments need to look at public data, which is available from the government and other organizations. Public data includes demographics and financial information and may even include climate change and environmental data that affects how businesses work. These datasets are free, very large, constantly changing, and challenging to locate and use. A clear plan for identifying and using this data benefits a sales and marketing department.

All of this data has to go somewhere, and that is the digital platform and the databases it comprises. It is not enough to have data—data needs to be organized to be useful. Since data is key to decision-making and action in most organizations, the digital platform is mission critical.

Data Collection and Storage in Organizations

In a typical business, a production database contains live data that the business uses to produce transactions such as customer requests, inventory management, and sales transactions. Because production databases are in constant use, they should not be used for testing or analysis. Any change to a production database makes the system unavailable for some period of time, which is not desirable when the database is the interface between the company and its customers, suppliers, and other important people outside the company. Instead, when an organization needs to work with a production database, it moves its data to either a development database (used for testing) or an analytical database (used for analysis).

Spotlight on Ethics - Working with Live Data

One of the best practices for working with databases is that users do not work directly with the production system—that is, the production database containing the live data that actively run the business. But why? It seems that working directly with the production system would be the most efficient choice. In this case, however, following best practices prevents users from making mistakes that can seriously compromise the business. “Don’t work with the production system” is an ethical issue—part of the code of conduct that all analysts are expected to follow. Watch this [video using a case study to discuss this problem](#) to learn more about why this has become a standard in database work.

Access is a desktop-based relational database program that can be used to work with a wide variety of data quickly and easily, regardless of its source. Users can request data from the department in charge of the data warehouse, which is a type of data management system that businesses use, then download the dataset to work with to provide actionable information. Let’s see how this might look at WorldCorp.

John works as a data analyst for WorldCorp. The company has a robust data warehouse that stores vast amounts of customer data, sales records, and inventory information. The data warehouse is managed by the Data Management Department (DMD) within WorldCorp. (Other companies may choose to contract out their data management department to a third party.)

John is tasked with analyzing the sales trends of a particular product category to identify potential areas for improvement. To do so, he needs access to a specific dataset from the data warehouse. John follows the established procedure and sends a formal request to the DMD, specifying the dataset he requires and providing relevant details about his analysis objectives. The request outlines the specific parameters, such as the time frame, region, and product category of interest.

Upon receiving the request, the DMD acknowledges John’s inquiry and begins processing it. They retrieve the requested dataset from the data warehouse, ensuring that it contains the relevant information needed by John to perform his analysis. After completing the data retrieval process, the DMD notifies John that the dataset is ready for download. John receives an email with a link to access the dataset securely.

John clicks on the provided link, which directs him to a secure portal where he can download the dataset. He verifies his credentials and proceeds to download the dataset onto his local machine. With the dataset now available on his computer, John starts working with the data using specialized analysis tools. He cleans the dataset, performs statistical analyses, and generates insightful visualizations to gain a better understanding of the sales trends in the specified product category.

After extensive analysis, John identifies certain regions where the product category is underperforming. He also discovers opportunities for strategic marketing campaigns and potential adjustments to the product assortment. John compiles his findings into a comprehensive report, highlighting actionable insights and recommendations based on the analysis of the downloaded dataset. He presents this report to the company’s management team, enabling them to make informed decisions to improve the performance of the product category.

In this example, the user (John) requests data from the department in charge of the data warehouse (DMD), downloads the dataset, and uses it to generate actionable information through analysis.

Data can come in a variety of formats and configurations. The way in which data is stored can affect how the data can be used, so it is important that a business carefully consider how its data will be stored. There are advantages and disadvantages to each storage medium, but based on the type and volume of data being stored, some storage formats are more suitable than others.

Data Warehouses

When we discuss data storage, we are typically talking about the data warehouse—a data management system that stores and manages data, especially data related to business intelligence operations and activities. Data warehouses are built for storage, accessibility, and retrieval, as Figure 13.2 illustrates, and often integrate data from a variety of sources within the organization. You may also hear the term data mart, which is a subset of the data warehouse that is optimized for specific workgroups or functions. Typically, data flows into the warehouse on a predetermined schedule, adding to the existing data available for analytical work.

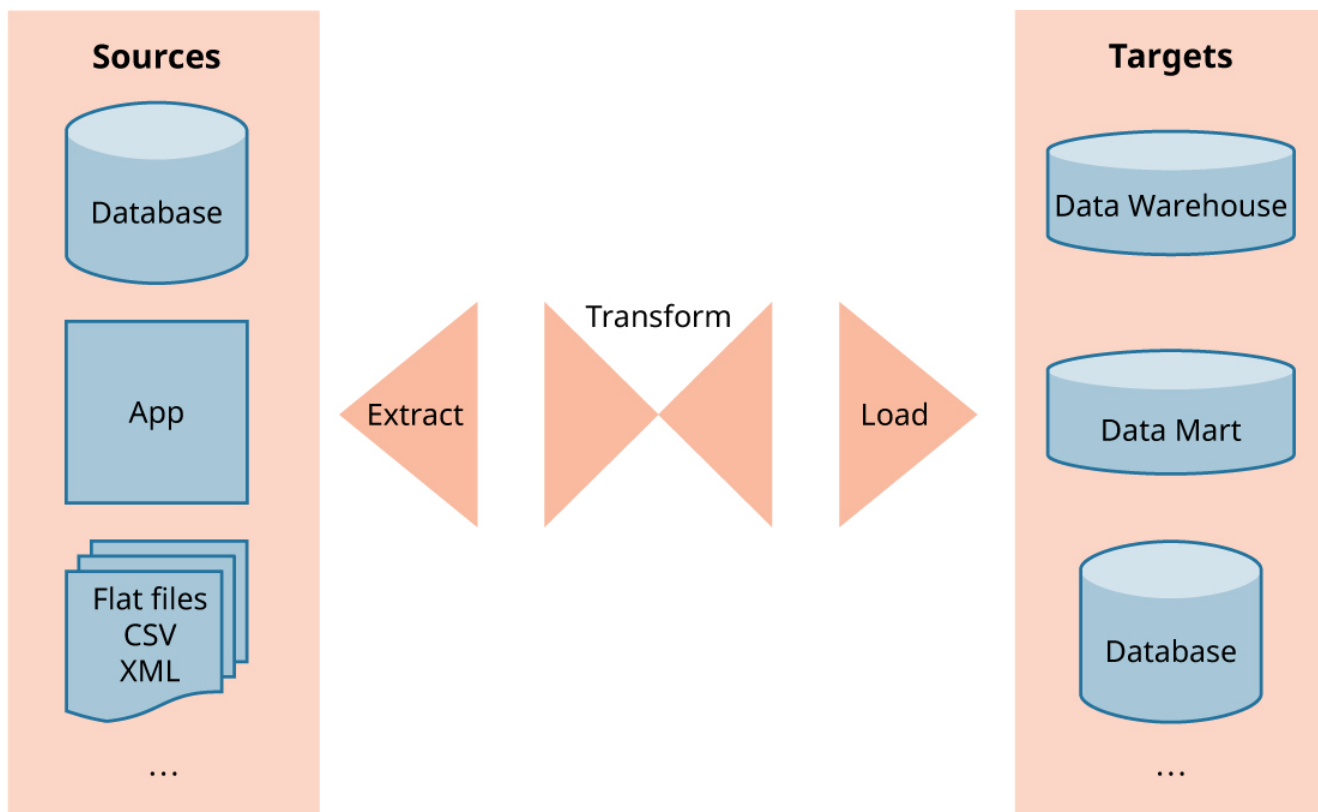


Figure 13.2: Data can be transformed in the database through organizing and processing. (credit: modification of “Extract, Transform, Load Data Flow Diagram” by Bamyers99/Wikimedia Commons, Public Domain)

A data warehouse is also a type of database, but it differs from the type of database you will use in Access. As an organization’s central data repositories, data warehouses hold a lot of information and use a complex structure. Typically, a department or team will include someone who is trained to interact with the data warehouse, using business intelligence tools or other analytics technology. That employee can then use Microsoft products such as an Access database or an Excel spreadsheet for analysis and preparation for management to aid in decision-making.

Flat Files

Many businesses store their data in flat files—individual files that have a tabular structure, including single spreadsheets, other tables, and stand-alone lists of information. Flat files have no set links to other information. In fact, if there are any links, they will be obvious to employees but typically not in documents or parts of the file itself. Flat files tend to look like tables, with headers at the top and a row representing each record, as Figure 13.3 shows.

Flat File Model

	Route No.	Miles	Activity
Record 1	I-95	12	Overlay
Record 1	I-495	05	Patching
Record 1	SR-301	33	Crack seal

Figure 13.3: Flat files are mostly static repositories of data, usually represented in tables, that workers access directly as needed.

Flat files are easiest to use when datasets are small and when the individual using them understands the data well. But even flat files that are quite large, containing thousands or hundreds of thousands of records, can be usable if all records follow a consistent pattern. Flat files are typically created by departments and individual employees in the course of their work, and most organizations

have a huge number of these files, which are stored on personal computers and may not be updated or even available beyond the person who created them. In this way, they are a form of dark data—data that is not used or actionable for the organization.

Flat-file databases are also used in other ways, including the transfer of information from one system to another. For example, suppose a sales organization wants to examine inventory records from a warehouse. In this case, the warehouse may send them a flat-file database that follows a uniform pattern, with columns for fields and rows for records. This type of flat file merely stores information; it does not have an indexed system or any links to other systems, and it is typically in a text format. Common examples of file formats that contain plain text include CSV (comma-separated values, where data items are separated by commas), txt (text, where items are separated by space in the file), or tsv (tab-separated values, where items are separated by tabs). Often businesses must consider the other systems that their database will interact with before deciding on a data format. Plain text is one of the most universally supported formats for data, making it simple to move data between business systems.

Relational Database Management Systems

The power in a digital database comes from the ability to connect data from one area to another. A relational database management system (RDBMS) is software that houses, manages, accesses, and retrieves data in its database. The term *relational* refers to the way that data links together by creating relationships between the tables that hold the data itself. In this way, relational databases can hold much more information that is complex and yet easily accessible.

There are many different types of RDBMS in use today. Access is a desktop system that holds information that is typically generated in other, larger database systems running on dedicated database infrastructure servers. For example, a data analyst might use Access at their computer at their desk at headquarters but will pull the data from the more powerful relational database systems running in WorldCorp's dedicated data center. Still, Access is also an RDBMS, with the power to manage relationships between tables, making it possible to create complex reports.

A relational database is a type of database that stores and provides access to data. Relational databases hold a collection of items, with preset relationships between them. Sometimes people use data, database, and relational databases interchangeably, though you may need to pay attention to context to understand if a person is talking about the data, the database, or the system itself.

A relational database and RDBMS are designed to give structure to data and to ensure that data is stored and used effectively. Examples of large-scale RDBMS include MySQL, IBM DB2, Oracle DB, and Microsoft SQL Server. Although each of these larger databases support querying and generating reports, in many small to medium-sized businesses, data is often exported and brought into an Access relational database for further refinement, analysis, and reporting or brought into Microsoft Excel for more complex statistical analysis.

Relational databases, running in an RDBMS, allow us to join or connect tables, perform complex analysis, and run reports efficiently. A join is a function within an RDBMS that sets up the relationship between records in two tables. In a data warehouse environment, relationships (or joins) that connect tables make it possible to store vast quantities of data and still be able to retrieve them quickly and efficiently.

Access, as a desktop RDBMS, uses the same idea of connecting tables through joins to make data retrieval quick and efficient and has the added benefit of using a graphical user interface. To work with data in Access, you do not need to know a language such as Java, Python, R, or SQL. Since we often need to analyze information from a larger data store, like the data warehouse, being able to work with relationships makes it possible to efficiently access information from multiple parts of the database.

Storing Data in an RDBMS

An RDBMS stores data in a database, which is a series of connected tables. The RDBMS provides the software and system structure for the data; the database is the holding area of the data. In practice, the term *database* is usually used for both of these, but it is important to understand that the data is separate from the program housing it. As a result, you can move data from one software to another program, such as from Access to Outlook or Word for communication distribution.

A database can hold a variety of data types in its data tables. A relational database that uses structured data can recognize and process data based on its type; for example, it can understand the difference between text and numbers and process them accordingly. Data types in Access are shown in Table 13.1.

Table 13.1: Data Types in Access These are some data types typically housed in a relational database.

Data Type	Description
Text	Alphanumeric data (in either short or long format)

Data Type	Description
Number	Numerical data
Date/time	Date and time
Currency	Monetary data with four decimal places
Aut/nonumber	Unique value set by Access
Yes/No	Data stored as yes/no (true/false)
OLE object	Pictures, graphics, or other files that are stored
Hyperlink	Link to a document (online, local computer, or network)
Attachment	Full file attachment stored in the database
Calculated	Mathematical expression (or calculation)
Lookup Wizard	Field that uses the contents of another table to populate the field

Data Manipulation

One of the strengths of a database is that it enables users to manipulate data, changing it in a way that makes the data more useful and actionable. An example of data manipulation is the creation of new fields based on combining and calculating existing fields within the database. WorldCorp often manipulates its data as new products are added to its portfolio or old products are updated with new features. Data manipulation allows users to set up the data in a way that helps them and their audience make sense of it and transform it into business intelligence.

Objects and Relationships

Objects and relationships are key concepts in a database. Objects are the items that make up a database, including tables, queries, forms, and reports. Relationships link objects together in the database, as Figure 13.4 illustrates, which in turn improves the performance of the database itself.

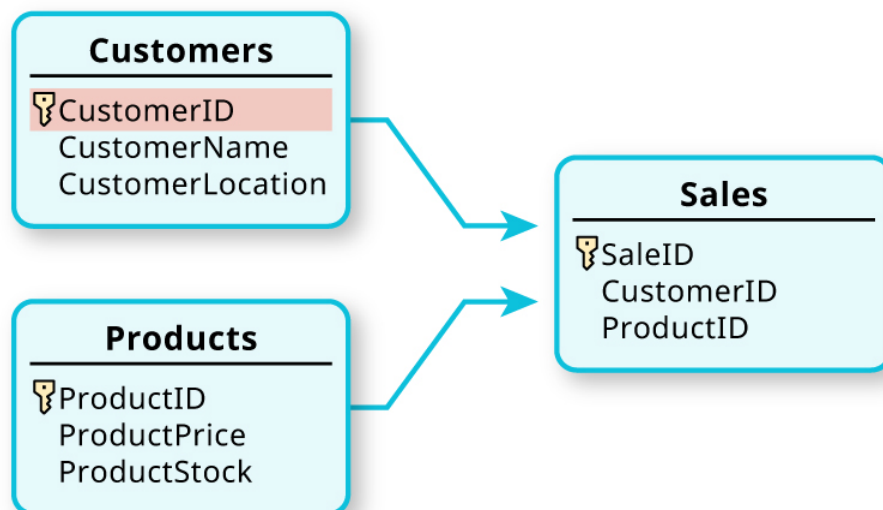


Figure 13.4: Relational databases use tables with defined relationships between them to organize and manage data, while flat-file databases typically do not have this level of relational structure.

A database object is used to store, reference, or interact with the database. There are four main types of objects in an Access database:

- Tables: the structures that hold individual data records
- Queries: structured data requests

- Forms: formatted screens for inputting or viewing data
- Reports: formatted summaries of specific information in the database

A relationship (also known as a join) tells the database that two records have a connection. This means we can break up a table into many tables, which in turn keeps screens manageable and speeds up processing. Relationships are created by using a key field, which is a field in a table that serves as a unique identifier for an individual record in a table. For example, in a database about employees, an employee ID number would be a key field and would act as the link between tables that hold employee information, payroll information, and even performance information in a sales department. Instead of trying to hold all this data in one extremely large table, we can have many smaller tables, each specializing in a particular type of data and linked to other tables by key fields, such as primary or foreign keys.

Tables

Tables hold the data within a database organized into rows and columns in two dimensions. A table consists of columns, each of which is a field, and rows, each of which is a record. The column or field names apply to each record in the table and create the structure the database uses to organize information. A database can include many tables that are joined together by relationships between fields. This structure makes relational databases fast and efficient when retrieving data.

Each table typically contains a key field. A primary key field is a unique field in the table, which serves as a unique identifier for the individual record you are looking at. A table can also have a foreign key field, which refers to the primary key field in another table. For example, an invoice number for a sales record is a primary key field—it is unique—but it can also appear as a foreign key field in other tables. In this way, the primary key field prevents duplication in a sales invoice table and also provides a link to the sales invoice list as a foreign key when it appears elsewhere in the database. In Access, primary key fields have an image of a key next to them when viewed in Design View, as shown in Figure 13.5.

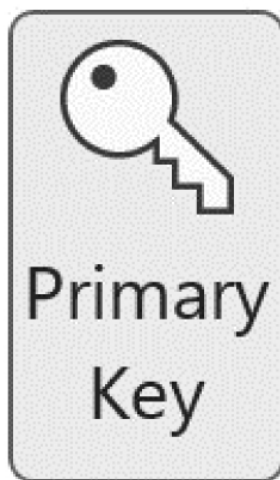


Figure 13.5: The primary key is a unique identifier within a database table that ensures each record is distinct and provides a reliable way to access, reference, and link data across tables. (Used with permission from Microsoft)

Key fields make it simple to gather useful information about a single data point from multiple tables. WorldCorp might use a model number as a key field in a database that stores the features, availability, and price of their products. In another database, WorldCorp can use a customer identification number to track various customer metrics such as location, purchase history, and contact information across various tables.

Queries

In a large database, it can be difficult or even impossible to find the information you need by simply scrolling through the records. It is even more difficult when a database contains many tables linked by joins. Instead, you can design or build a query, a structured request that allows you to set up criteria instructing the database to return what you are looking for.

For example, suppose your boss at WorldCorp asks you to gather sales transactions on a specific date or all transactions for a certain dollar amount. Or you might want to be more specific, asking for all sales of a specific product in a specified time period that exceeded a certain dollar amount. In each case, you can use queries to ask the database to provide the information you need. In many relational databases, structured query language (SQL) is used to create and run queries. SQL is a system of written

commands used to manipulate data in a database; it was developed by IBM in the 1970s to allow its engineers to interface with database systems more efficiently. SQL is in Access; however, users can create queries without knowing SQL.

Forms

A form is a database object that simplifies the use of information within the database. A form creates a user interface—a way for individuals to interact with the database—without directly using the table, which can be slow, overwhelming, and risky (because data in the table can be inadvertently changed). A form can be used to input information into a database (e.g., new customer information that customers enter themselves). A form can also be used to retrieve information (e.g., when a customer service representative needs to look up a specific customer record to resolve an issue). Examples of sales forms include customer information, sales invoices, and sales records.

Reports

A report is a formatted output of a database, ready for use and action. A report summarizes data in a way that matches the needs of the audience. For example, a product listing on a website or in print is a format ready for presentation to a customer or someone else who needs to see all products at a glance. A listing of sales, summarized by month, is also ready for presentation if the audience needs to know monthly sales figures for a decision or action. Reports are typically one of the final outputs of a database, as they are directly linked to the actionable goal or business intelligence activities of a specific workgroup, team, or project. An example of a sales report may cover sales by product or sales over a time period, such as a month or year.

Applications of RDBMS

Databases offer unique functionality that goes beyond the calculations and statistics provided by other programs such as Excel. This is because of the RDBMS functionality. A relational database provides structure for data, includes relationships that connect pieces of data in meaningful ways, and allows controlled access through queries, forms, and reports. Because of this advanced functionality, RDBMSs are used in a variety of use cases across multiple industries.

As organizations work to transform themselves to meet the needs of the twenty-first century, companies are shifting how they view customers, employees, and work in general. Databases play an integral role in this shift. Digital transformation relies on data, and for companies to transform, they need not only good data but actionable data, available when they need it, where they need it. The core of this change is the data management system, and an organization's data landscape is a constantly growing and evolving part of the business plan. Since the digital transformation of a business or organization is about leveraging technology to change the way a business is run, an organization's collection and use of data is an ongoing process, with the focus on people—not technology—and how we enable people to be more innovative, responsive, and successful.

Large organizations typically collect data through transactions and sensors, which then store the data. Eventually, data is transferred to a data warehouse, which stores data for historical purposes. From the data warehouse, organizations can bring datasets to a relational database such as Access for analytics and use. Data warehouses and Access are built on relational database models, which provide benefits, including the following:

- Data accuracy: The use of primary and foreign keys improves accuracy.
- Easy access to data: Data is held in tables that are easily accessed through queries, improving speed and efficiency.
- Flexibility: A RDBMS easily scales to include more data and tables as needed by the organization, without affecting the existing structure or the data in the database.
- Security: Because data is held in separate tables, it is possible to create a variety of user roles and permissions, allowing individuals to access some data without needing full system access.
- Adaptability: Databases can be easily modified as business requirements change.

Because data in an organization is created in production systems and then brought to the analytical system for work, an RDBMS focuses on keeping the data organized and ready for retrieval. The RDBMS also allows the analyst to create database objects that make working with the data more accurate, effective, and efficient.

Examples of Database Use

How do companies apply the information they get from databases? They do so in many ways, most commonly through their marketing and customer service initiatives, for example. WorldCorp's Sales and Marketing Department depends on databases to inform marketing strategy and customer service processes and procedures and to understand sales trends and future opportunities.

Example: Marketing Strategy

Recall that WorldCorp manufactures health-care products. Its leadership has been looking at their online promotion of allergy medications. They have a variety of advertisements online, served by social media platforms. While they can easily look at the data to see the click-through rate (which is the number of times a consumer clicks an online ad in relation to how many times the ad is shown) and the purchase rate on specific partner sites, they also want to know if they can better understand the customers who are considering purchasing but have not yet done so.

Using the data warehouse, you as the WorldCorp employee can request a report of all shopping carts that were populated with a particular medication from an online shopping site partner. Since the data warehouse stores both the WorldCorp information and key information on sales from major partners, you are able to ask for a year's worth of sales data, with customer demographics including location, family structure, and other related purchase information such as, say, other antihistamine and allergy comfort items. You receive the data, import it into Access, analyze it to identify who was likely to add the item to the cart and what they did after placing it in the cart (purchase the item, purchase something else, or another action), and prepare this data for the strategy meeting.

Example: Customer Service

WorldCorp supplies medication to its vendor partners, including online distributors (both over-the-counter and online pharmacies) and physical distributors, as well as health-care organizations. They are looking to improve the customer experience for their vendors. Now you are the analyst working with the customer service team in the Sales and Marketing Department. You need to understand where vendor needs pain points are to prepare a report for the leadership team. You request a list of all vendor support contacts, which includes the vendor name and contact information, the request itself (including date, time, and method of request), and the resolution. You analyze them to determine the most common types of questions and requests, where these requests originate from, and how they were resolved. You use the dataset to create a report for the leadership team that summarizes and highlights the top issues over the past eighteen months.

Example: Sales Trends

Next, the sales team at WorldCorp is looking at where to focus their attention for the upcoming year in terms of promoting their medication for obstructive airway diseases such as chronic obstructive pulmonary disease. The team wants to identify areas in the world where their medications may be needed and prepare for the demand. Working with their public health experts, they know that obstructive airway diseases are more common in people who smoke or breathe in secondhand smoke or who live in areas with high pollution. They also learn that people are often over age 40 years when symptoms appear, that asthma increases the chances of developing these diseases, and that they occur more commonly in individuals with certain jobs or who have had a high number of respiratory infections in childhood.

You, again as the business analyst, want to identify the geographic areas that are most likely to increase the risk of obstructive airway diseases and look at sales trends in these areas over the past five years. You work with the Data Science Department to ensure that the public data that WorldCorp has available and the internal sales data (including the data from the vendor) are available. You then create reports that show specific geographic areas and risks and the sales trends for these areas over the past five years.

NoSQL Databases

In addition to Microsoft Access and other databases that define the relationships between data stored in them, which are the main topics of this chapter, there are other ways that organizations can capture and store data: nonrelational databases. A NoSQL database (NoSQL stands for “not only SQL”) stores data in a different way than the databases we are discussing in this chapter. A NoSQL database, also known as a nonrelational database, is a type of database management system that provides a mechanism for storing and retrieving data that is different from traditional relational databases. Unlike relational databases, which store data in structured tables with predefined schemas, NoSQL databases store data in a variety of ways and are designed to handle large volumes of unstructured or semistructured data. NoSQL databases have gained popularity due to their flexibility, scalability, and high performance.

NoSQL databases often store documents and images and manage large amounts of data in their original formats, rather than in tables; this is also known as unstructured data. NoSQL tends to be used when an organization needs to store huge volumes of both structured and unstructured data and also needs the ability to quickly develop and scale data initiatives. Scaling data initiatives is the process of expanding and advancing data-related projects, strategies, and operations to handle larger volumes of data, accommodate increased complexity, and support the growing needs of an organization. It involves implementing measures and techniques to effectively manage, process, and analyze data on a larger scale, often in terms of volume, velocity, and variety. For

example, WorldCorp uses a NoSQL database to record logistics for shipping its products. Because the data involved with logistics has unique structure and is greatly varied, a NoSQL database is an ideal candidate for managing this data.

Data Lakes

You may also hear about data lakes in your organization. A data lake is a centralized repository that allows for the storage and analysis of vast amounts of diverse data in its original, raw, unprocessed format. A data lake can store both structured and unstructured data and make it available for use and analysis as needed. In a data lake, data is collected from various sources, such as databases, applications, sensors, social media feeds, and more, without the need for up-front structuring or transformation. This raw data is stored as is, preserving its original format and granularity. This includes structured data (e.g., relational tables), semistructured data (e.g., JSON, XML), and unstructured data (e.g., text documents, images, videos). Data lakes offer several advantages, including the ability to store and analyze large volumes of diverse data, support for flexible data exploration and discovery, and the potential to uncover valuable insights and patterns. However, they also present challenges related to data quality, data integration, and ensuring proper governance practices, as the raw, unstructured data may require additional effort for data preparation and cleansing.

Although many organizations use NoSQL databases and data lakes, RDBMSs are still the most common type of databases used to power transactional systems, such as managing and tracking customer profiles at WorldCorp. At present, Access does not have a built-in mechanism to directly interface with NoSQL systems or data lakes, although data that comes from a NoSQL database or data lake can be formatted and sent to you for analysis.

1.1.1: Introduction to Microsoft Access is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by LibreTexts.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.0: Chapter Scenario** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **6.1: What is a Database?** by Marcus Lacher is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.
- **13.1: What Is a Database?** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

1.1.2: Microsoft Access- Main Features and Navigation

 [Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Demonstrate navigation of the Microsoft Access screen interface
- Identify the main features of Access
- Understand how to close Access

The benefit of Microsoft Access is that its design and structure enable it to communicate with other systems that hold data. Unlike larger database systems that run on dedicated servers, Access runs on a regular desktop computer. Because it is a desktop database system, the user can work within the database without connecting to those larger systems.

As a Microsoft software package, Access follows many of the conventions of Microsoft Office. This provides a consistent, familiar interface for work and ensures that file transfer is seamless between Microsoft Office software, for example, between Access and Excel.

What Is Access?

Access is a RDBMS that can import and manipulate data from other systems on your desktop computer. While data science professionals work directly with the data warehouse and can create queries using programming languages such as R and Python, individuals who need to analyze data as part of their jobs within a department often find Access to be a user-friendly software package that has the power to access, retrieve, and report data using a Microsoft interface. Access also integrates well with Microsoft Excel, providing even more analytical and reporting power that can inform decision-making.

In small businesses and organizations, Access can act as the main data store; however, it is much more likely that you will use data that is collected and stored in other systems and then imported into Access for analysis.

The Role of Access in an Organization

Access is one of the many tools that make up an organization's data landscape—the entirety of data storage and sharing in an organization. An organization may have both production databases and analytical databases.

Production databases are systems that handle daily transactions related to an organization's core business functions, such as sales, inventory, human resources, payroll, and purchasing. Because production databases provide value to the organization, companies devote significant resources to ensure they remain online and active. Companies do not work directly with a production system when analyzing a business because this may interfere with regular operations for customer transactions; instead, companies use a data warehouse or analytical database for building new applications or testing new features. This type of database is entirely separate from the production database, so changes to the analytical database will not modify the production database until the developer is ready to push the changes to the production database after thorough testing.

Analytical databases (data warehouses) serve as an organization's single point of truth (SPOT) in which a primary copy of a data element is stored. Changes made to a data element outside of the data warehouse are not considered final until the primary data element in the data warehouse is also updated. These large databases contain a vast number of records, each of which stores historical information about the organization itself. A data warehouse is not a production system; rather, it is filled with data from the production system, collected over time and organized in a way that makes retrieval and use easier for business use. We can request information from the data warehouse, which is then exported to a file we can use in our analytical software, such as Access or Excel. The ultimate goal of using data is to make good decisions based on the data and not on emotions.

An organization's data landscape can also contain numerous flat files held within teams and workgroups, other systems that run on databases that should be connected to the data warehouse, and even information from vendors and public sources that can inform data. The data landscape may include information scraped from the web, downloaded and processed from public sources, and environmental data. The extent of an organization's data landscape is based on the size and role of the company; larger companies

will often have more complex data landscapes, as will organizations that are undergoing digital transformation and either have or are building a core platform for all business functions.

Access and Excel

Access and Excel are both software packages in Microsoft Office. They are complementary: Although they have some functions in common, each package is designed for specific uses. Sometimes, the overlap or commonalities between them can make it difficult to know which is most useful. Both Access and Excel can store large amounts of data, run powerful queries, use analysis tools to manipulate data, and perform complex calculations. However, each software has its own strengths, and choosing the right software will make your work easier, as Table 13.2 outlines.

Table 13.2: Choosing Excel or Access Excel and Access each have different strengths when handling information.

Use Excel to:	Use Access when you:
Analyze a specific dataset	Need to track and record data regularly
Perform complex numerical calculations	Need to ensure data integrity
Work with a table for calculations	Expect data to be accessed by multiple users
Perform what-if analysis to run scenarios	Store data in multiple tables
Create advanced pivot tables and charts	Need to connect to other data sources to refresh data (either manually or on a schedule)
Create visualizations	Secure data at the user level (each user can have different roles and access to specific data)

Access and Excel can be used together. It is possible to move data from the data warehouse to Access and then, as necessary, query and export data from Access to Excel for further analysis and visualization. Transferring data from one program to another is one of the benefits of using a desktop database program; as part of the Microsoft suite of products, both Access and Excel can communicate effectively, so users can also import flat files from Excel and convert them to relational tables.

Features of Access

Access is a cost-effective database solution. It runs on a desktop computer and provides an easy-to-use interface that is familiar to users of other Microsoft software. Access interacts well with Excel for deeper analysis, with Word for reporting, and with PowerPoint for presentations. It can work with data from external sources, such as a data warehouse, or with exported files from the web and other systems.

Microsoft Access:

- Creates the framework or structure to store small bits of information (data) in a database
- Allows users to open tables and scroll through records
- Provides an easy-to-use interface for adding, modifying, and retrieving records
- Has a process for creating queries to select, combine, and calculate data from multiple tables
- Provides user-friendly screens and support for creating queries, forms, and reports that allow access and use of the data held in the database

One difference between Access and other Microsoft 365 software is in its introduction or opening screens. Access loads one database at a time, and all screens focus on the open database. When you first open Access, the screen will be blank, Figure 13.6 until you load a database. Once loaded, the screen shows all available commands.

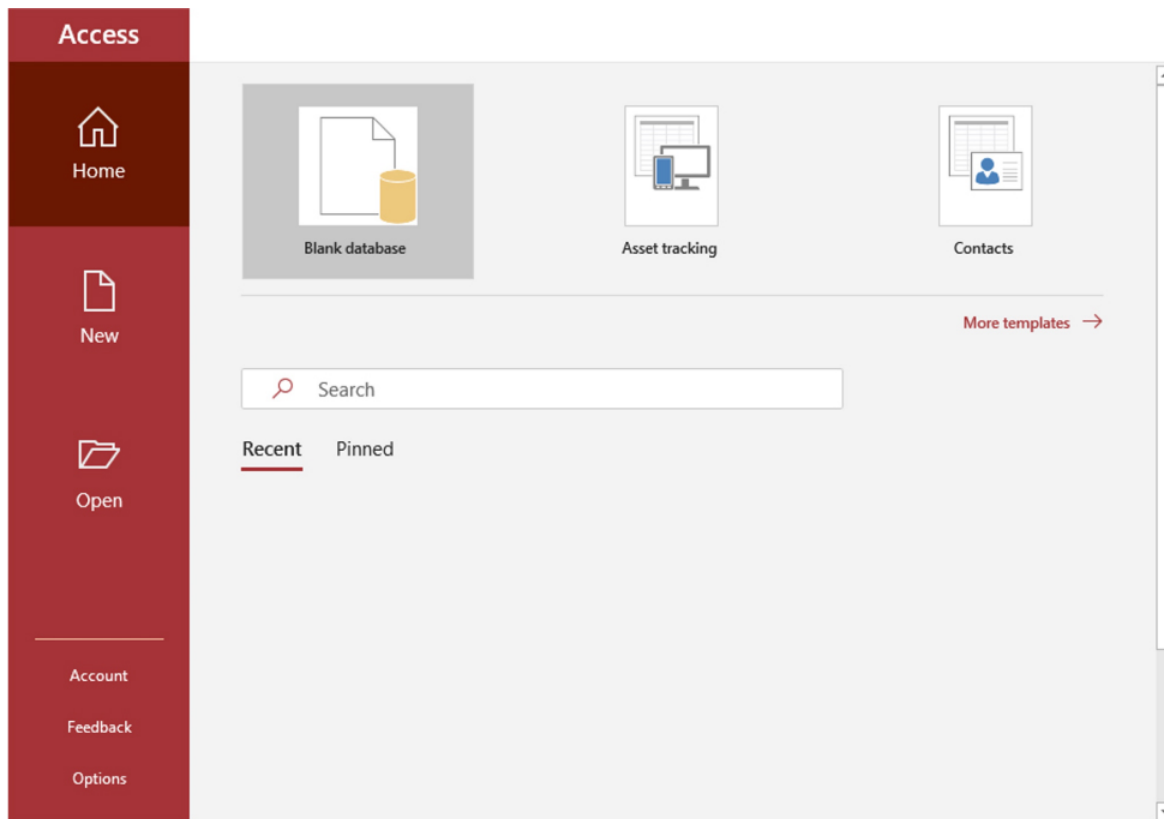


Figure 13.6: The Home screen will give you the options to create a new blank database, choose from preloaded templates provided by Access, or search for an existing database file. (Used with permission from Microsoft)

As you move through Access, different commands will become available based on the active objects. For example, you will see different commands when working in tables versus working with queries. The main working area in Access is the center of the screen, with navigation on the left and commands across the top.

The Opening Screen

When Access is open without a database, most commands either do not appear or are unavailable (grayed out). Without a database, the software is inactive. When a database is open on the screen, you will see commands available on the ribbon along the top (Figure 13.7) and a navigation area along the left edge of your screen (Figure 13.8). Only one database can be open at a time in Access, and all software is focused on supporting your open database.

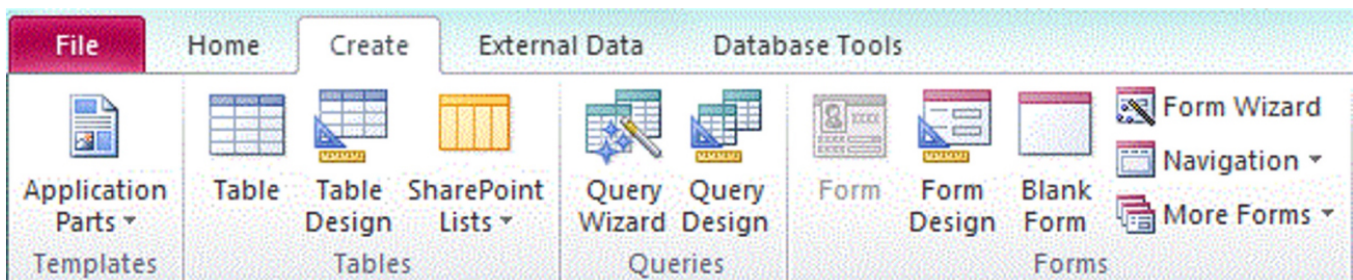


Figure 13.7: The Create tab provides tools and options for creating various database objects such as tables, forms, queries, reports, and macros to design and manage a database system. (Used with permission from Microsoft)

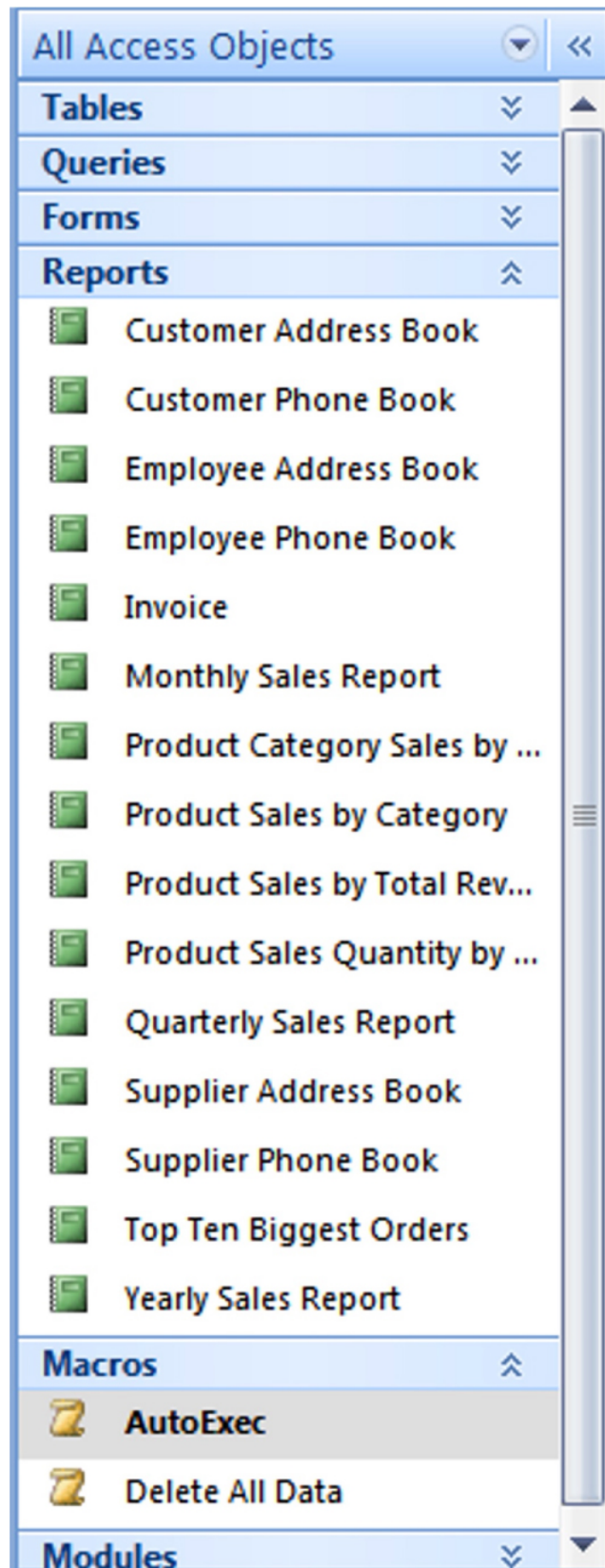


Figure 13.8: Access has many reports and models already built and modifiable; you might want to consider these before creating one from scratch. (Used with permission from Microsoft)

As you click through the ribbon, notice that some commands are not available unless you have specific database objects open, and you can move between objects in your database by using the navigation area on the left side of your database screen.

Although there are many different database software programs in use, most of them have the same parts. Access, as a RDBMS, follows the typical database conventions, or standards, in use in all relational database systems.

You can think of Access as a container that holds database objects, which, recall, are items in a relational database. Each object plays a role in managing the database and the data it stores. Because the database is electronic, it can grow to hold many objects and can create links and new uses for each over time.

Like many databases, Access databases usually contain tables, forms, reports, and queries. (They can also contain macros and modules, which you will learn about in Advanced Database Use.)

Locating and Opening Database Objects

Once you open the database or the container that holds database the object, you will see a navigation area open on the left side of the screen. This area has headings for each type of object that is present in the database, followed by the names of each object you have. To open any object, double-click on its name, and it will open in the main work area on the right side of the screen (Figure 13.9). To close an object, click on the X next to the name of the object.

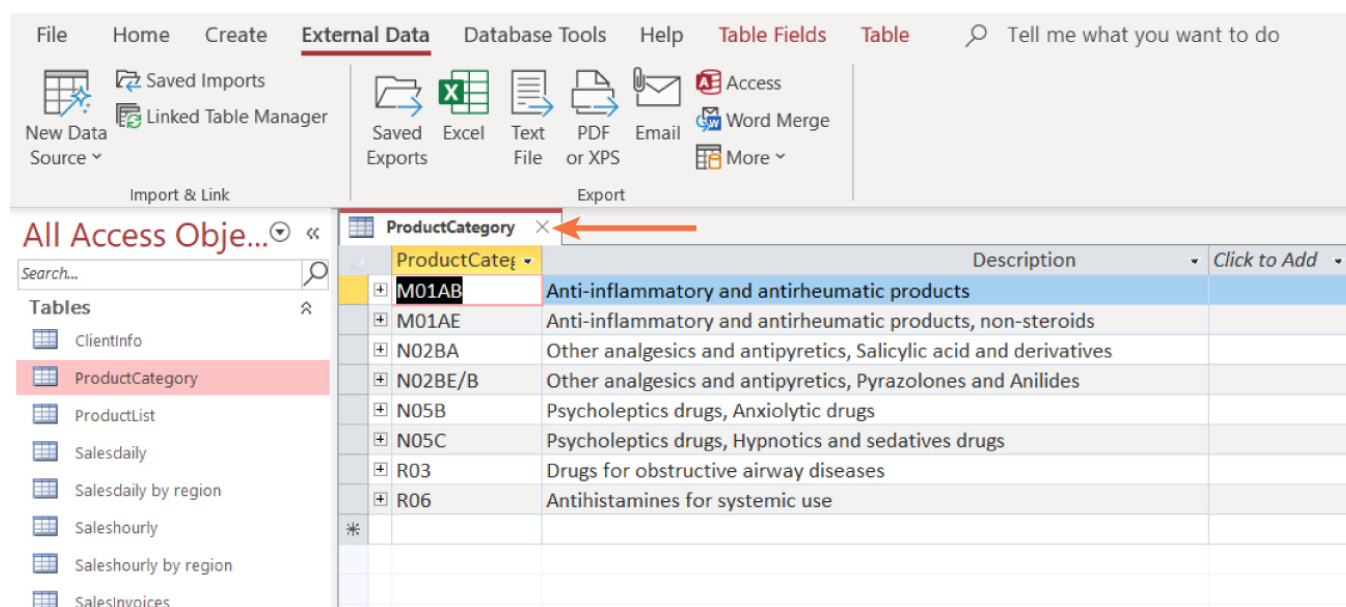


Figure 13.9: Open an object from a database in Access. (Used with permission from Microsoft)

Tables

Tables hold the data in a database. One database can contain many tables, each of which holds a different record (or set of related information) or parts of a record. One benefit of a relational database is its ability to split large sets of information about a specific item, person, or idea over multiple tables. This action speeds up processing and gives the database a better organizational structure.

As you learned in What Is a Database?, tables are built with rows and columns. The rows of a table consist of individual records or lines of related data. The columns represent fields or headings referring to information that is likely to be included in all the records. This structure, which characterizes all relational databases, makes it possible to share datasets between different database software packages. An example of table structure is presented in Table 13.3.

Table 13.3: Table Structure This structure characterizes all relational databases, allowing the sharing of datasets between different database software packages.

ID	FirstName	LastName	MiddleInitial
1234	Mary	Nanas	A
2345	Jose	Garcia	I
3456	Andre	Williams	C
4567	Lisa	Estrich	J
5678	Pat	Johnson	E

ID	FirstName	LastName	MiddleInitial
6789	Stevie	Larose	Y

This small table contains four fields—ID, FirstName, LastName, MiddleInitial—and six records, which contain the information for each field for six different employees.

Queries

Queries are a way of asking questions of a dataset. Recall that a query is a structured request for information, allowing you to choose what information appears, the order in which it appears, and how it is organized. Queries can also be used to calculate information based on fields. Typically, when you interact with the database, you will be creating queries to satisfy business requirements.

Like other database objects, queries appear in the navigation area of the database. To open a query, double-click it, and it will appear in the work area of your screen. In Access, you can change how you view a query, choosing either Datasheet View, which shows the results of the query in table form, or Design View, which shows the structure of the query and allows you to modify it. You will learn more about these views in Querying a Database.

Reports

A report shows the results of a query, formatted in a business-friendly way. Creating a report allows you to choose the precise information you want to include and the format in which you would like your audience to view it. Since tables and queries typically return results in the form of tables, reports allow you to focus on the message your audience will need for decision-making, not on the data. One advantage of creating reports is that you can set up standard reports that you will need to run frequently (e.g., a report of sales totals) and then open them and view the latest results as often as necessary.

Like other objects, reports in Access can be created in a Wizard that walks you through the basics, and then you can adjust the report in Design View. Building a useful report can take time and care to ensure it summarizes and displays information in an actionable way that supports the work group's goals.

Forms

Recall that a form in a database allows users to work with records in an easy-to-use format. Forms limit the individual's view of the database to what is necessary for a specific task or information request. On most websites, we interact with forms that are built on the basis of queries: They request and present data taken from the underlying table and query structure. Forms are often based on predefined queries that run over and over again and, as with a search for a product, may prompt the user for a response.

Forms appear in the navigation area of the database and can be opened in Layout or Design View, which allows you to modify the form itself (Figure 13.10), or in Form View (ready for use) (Figure 13.11).

The screenshot displays the LibreTexts Design View interface for an 'Inventory Data Sheet'. The interface includes a top toolbar with various icons for text, images, and layout. Below the toolbar, a 'Controls' panel shows the 'Inventory Data Sheet' form. The form is structured into three main sections: 'Form Header', 'Detail', and 'Form Footer'. The 'Form Header' section contains the title 'Inventory Data Sheet'. The 'Detail' section contains various input fields and controls for data entry, including 'Manufacturer:', 'Model #:', 'PCN:', 'Mfg Serial #:', 'Express Service Code:', 'Mfg Date:', 'Received:', 'Checked Out To:', 'Equipment Removal', 'Method of Removal:', 'Date of Memory Removal:', and 'Condition:'. The 'Form Footer' section contains navigation buttons: 'Previous', 'Save', and 'Next'. The form is displayed in a grid-like structure with a dotted background.

Figure 13.10: Design View allows users to modify and customize the structure and properties of database objects, such as tables, queries, forms, and reports, providing a detailed and precise way to design their database elements. (Used with permission from Microsoft)

Figure 13.11: Form View is the user-friendly interface for interacting with and entering data into a database, presenting records in a user-readable format and allowing data input and editing. (Used with permission from Microsoft)

Link to Learning

Sometimes the best way to learn is to explore on your own. Visit [Microsoft's featured Access templates page](#) to choose a template and then download, open, and explore it in Access.

Closing Databases

In Access, only one database can be open at a time. As you work, all on-screen commands will focus on this open database, and you will need to close the database itself to end your working session. When you are ready to close the database, Access will either save objects automatically or prompt you to save objects before closing. In general, Access will automatically save records when you are done with them (i.e., when you leave the row) but will prompt you to save changes to objects, such as tables and queries, so you can decide whether you want to save them.

If you want to make sure an object is saved in Access before you close the database, you can use any typical Save command shortcut (e.g., File>Save, Ctrl+S) to request that Access save your changes. Access will prompt you if you need to take any additional steps.

When you are done working with your database, click on File>Close to close all objects and exit the Access software. Be sure to close your database every time you are done using it; this will ensure that all objects are closed and that data is correctly written to the records within the database.

This page titled [1.1.2: Microsoft Access- Main Features and Navigation](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.2: Microsoft Access- Main Features and Navigation** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

1.1.3: Database Design and Four Main Objects

[Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Database Design

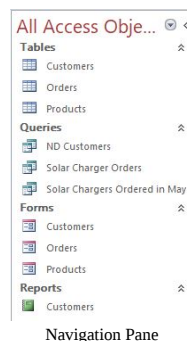
A properly designed database will divide data into tables to eliminate duplicate information (also known as *data redundancy*). **Redundancy** occurs when data is repeated several times in a database. Redundant data is bad, because it wastes space and increases the likelihood of errors and inconsistencies. A good database prevents redundant data when, for example, customer data (ID, Name, Address, etc) is stored in one table and then other tables store just the customer ID to refer to the customers. Another important design consideration to maintain is data accuracy. A common motto when performing data entry is “garbage in, garbage out”. The correctness and completeness of information is important if the user wants to rely on their data. If your database contains incorrect information, any reports that pull information from the database will also contain incorrect information. As a result, any decisions you make that are based on those reports will then be misinformed.

Creating a new database involves determining the database schema, which is a description of the data and the organization of the data into tables. Designing a schema consists of the following steps:

1. Determine the purpose of the database and which data to include in the database. It is a good idea to write down the purpose of the database on paper — its purpose, how you expect to use it, and who will use it.
2. Divide information into related groups to create tables and then give each table a descriptive name. Data should be grouped to ensure there is little to no redundancy. Each table must have data related to at least one other table through a table relationship.
3. Describe the fields and determine the primary key for each table.

The Four Main Objects in a Database

While Microsoft Access is made up of seven components, this text will focus on the main objects: tables, forms, queries and reports. Together, these objects allow users to enter, store, analyze and compile data in various ways.



Navigation Pane

When an Access database is opened, the objects are illustrated in the **Navigation Pane** on the left side of the window. The standard view in the Navigation Pane displays all objects in the database organized by object type. The Navigation Pane is used to run, open, and manage the objects of a database. Double-clicking an object will open the object in a document window.

A **table** is the database object that stores data organized in an arrangement of columns and rows. Each row is a **record** and each column is a **field** within each record. A record is all of the data pertaining to a person, place, thing or event. Each record will have one or more fields and every database must have at least one table. A relational database must have more than one table.

The other three objects are typically based off of the tables, so take the time to set up the tables correctly, the first time! A **form** object is typically used for entering new records into a table, or edit/deleting/display existing records in a table. Forms are the primary interface that most users of a database enter and display data. A **query** object retrieves specific data from other database objects and then displays only the data that the user specifies. Queries are the equivalent of asking questions about the data. The **report** object summarizes the records from a table or query into an easy-to-read format suitable for printing or exporting to other software programs.

Access objects have several views. Each **view** gives users different perspectives and offer different functionality. The two most important views are Datasheet view and Design view. **Datasheet view** displays a table in a datasheet where fields are in columns and records are in rows.

Customers								
CustomerID	First Name	Last Name	Phone	Street Address	City	State/Prov	Zip/Postal C	Click to Add
1	Devinder	Patel	218-555-5641	212 Ash Street	Aurora	MN	59990	
2	Maria	Escobar	218-555-5146	212 Birch Street	Lake Cosmos	MN	59991	
3	Ahmed	Rashad	218-555-1658	212 Cedar Street	Park Falls	MN	59992	
4	Arnold	Hammer	218-555-8811	212 Elm Street	Aurora	MN	59990	
5	Susan	Wong	(218) 555-0182	212 Fir Street	Aurora	MN	59990	
* (New)								

Datasheet view of a Customer table

This view is useful for comparing records and is often used when sorting or filtering data. This view is also practical for printing data since many of the records can fit on a single sheet of paper. Records can be added in datasheet view, but forms are a more typical object for data entry. Nonetheless, the asterisk (*) that appears to the left of the row below the last record indicates where a new record will appear. The grey box to the left of each record is a record selector. Clicking a record selector will make the record active. Right-clicking the record selector will allow users to delete the selector record or add a new record to the table. **Design view** allows users to setup and edit the fields in an object.

This page titled [1.1.3: Database Design and Four Main Objects](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher \(Minnesota Libraries Publishing Project\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

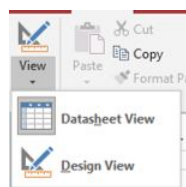
- [1.1.1: Introduction to Microsoft Excel](#) by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- [6.3: Database Design](#) by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

- 6.2: The Four Main Objects in a Database by Marcus Lacher is licensed CC BY-NC-SA 4.0. Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

1.1.4: Creating a Table

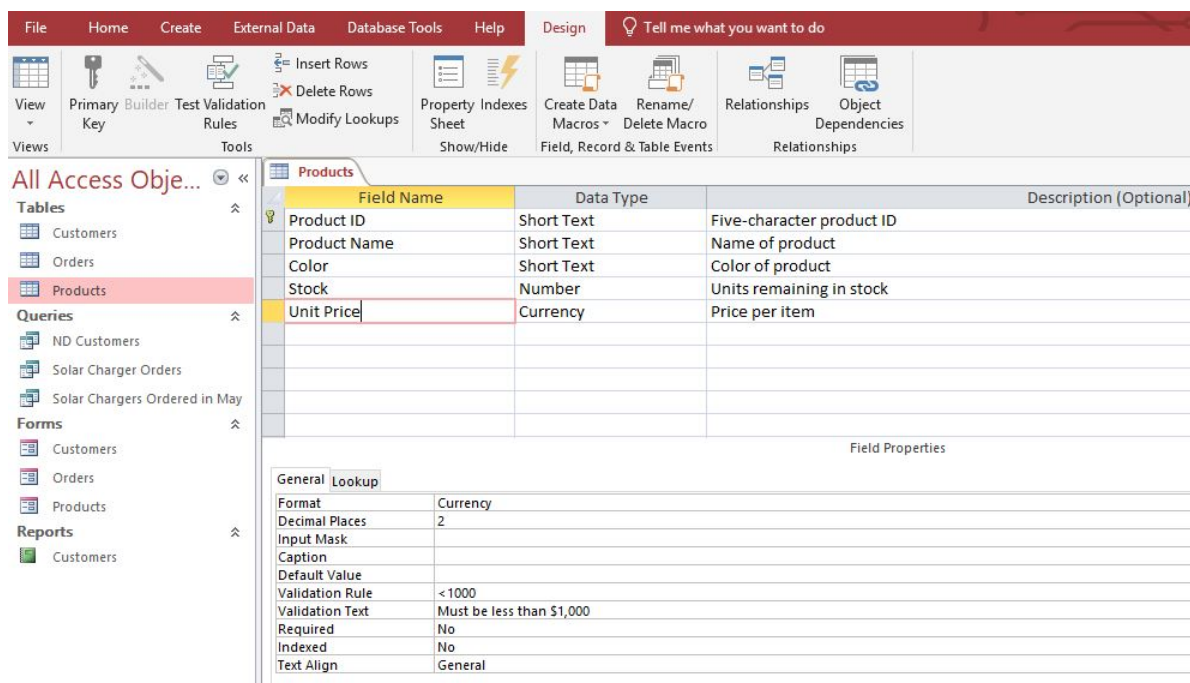
[Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

As previously stated, a database cannot exist without at least one table. Therefore, when a new database is created in Access, the first object that opens is a table in datasheet view.



Access View tab

Switching the view to Design View allows the user to make many design decisions in a much more intuitive view. Many users like to use Datasheet View to paste data copied from Excel. However, setting up the fields in Design View makes it easier to define each field's properties and data type.



Design View

Design View doesn't display any data. Therefore, there's more space available to display other settings. In Design View, you can see the data type listed next to each field. When a user clicks on a field (in the top frame), the bottom frame displays the properties for that field. The user can then change these properties as desired.

Creating a table in Design View allows the user to define advanced settings, like data types and properties, for each field in the table. However, the first step is defining the field names. Here are a few guidelines when creating field names:

- Names can include numbers, UPPER and lowercase letters, and any combination. Avoid special characters (exclamation points, periods, brackets, quotes, etc.)
- No leading spaces. Spaces between words is acceptable, but not common practice, because:
- Make the names as short as possible without excessive abbreviation. Formatting column widths for displaying in printouts can be challenging with excessively long field names. For example, Order # or Order Number are better than Customer Order Number.

- Ensure the names are unique per table. If a table has fields for Customer ID and Product ID, neither field should be named simply ID. Likewise with the word Name. It is too generic and needs a prefix identifier.
- Break data into multiple fields. For example, instead of one field for Customer Name, create fields for First Name and Last Name. This will allow sorting and filtering on single fields. Another example would be separating an address into multiple fields for Street Address, City, State/Province and Zip/Postal Code.

Every field has a data type. A field's **data type** indicates the kind of data that the field stores, such as large amounts of text or attached files. A field's data type determines what other properties the field has including its size and formatting options. Field data types are defined in the table design grid, where as field properties are defined in the Field Properties pane in the bottom of the Table Design View window. Data types include:

Data type	Description	Size
Short Text	Use for text or combinations of text and numbers (i.e. Phone Numbers).	Up to 255 characters
Long Text	Use for larger amounts of alphanumeric data, including sentences and paragraphs. Note: You cannot sort a memo field. However, they are searchable.	Up to 1 GB
Number	Numeric values.	4 bytes
Large Number	Numeric values. Stores a non-monetary, numeric value and is compatible with the SQL_BIGINT data type in ODBC. Use this data type to efficiently calculate large numbers.	8 bytes
Currency	Use for monetary data. Holds up to 15 digits of whole dollars, plus 4 decimal places. Tip: You can choose which country's currency to use.	8 bytes
AutoNumber	Unique value generated by Access for each new record, usually starting at 1.	4 bytes
Date/Time	Use for dates and times.	8 bytes
Yes/No	A logical field can be displayed as Yes/No, True/False, or On/Off. In code, use the constants True and False (equivalent to -1 and 0). Note: Null values are not allowed in Yes/No fields.	1 byte
Ole Object	Can store pictures, audio, video, or other BLOBs (Binary Large Objects).	Up to 2 GB
Hyperlink	Contain links to documents or other files, e-mail addresses, or web pages.	Up to 8 bytes
Attachment	Attach pictures, documents, etc. Can contain an unlimited number of characters per record. Access will compress the file.	Up to 2 GB
Calculated	Use to create an expression that uses data from one or more fields. You can designate different result data types from the expression.	Dependent on the data type of the result type property
Lookup Wizard	When selecting this entry, a wizard starts to help define either a simple or complex lookup field. A simple lookup field uses the contents of another table or a value list to validate the contents of a single value per row. A complex lookup field allows you to store multiple values of the same data type in each row.	Dependent on the data type of the lookup field

After creating a field and setting its data type, users can set additional field properties. The field's data type determines which other properties that can be set. The **field size** is the number of characters or the type of number it can store. For Number and Currency fields, the Field Size property is especially important, because it determines the range of field values. For example, a one-bit Number field can store only integers ranging from 0 to 255. In the **Field Properties** pane, on the General tab, enter the new field size in the Field Size property. Users can choose from the following values:

- **Byte** — For integers that range from 0 to 255. Storage requirement is a single byte.
- **Integer** — For integers that range from -32,768 to +32,767. Storage requirement is two bytes.

- **Long Integer** — For integers that range from -2,147,483,648 to +2,147,483,647. Storage requirement is four bytes.
- **Single** — For numeric floating point values that range from -3.4×10^{38} to $+3.4 \times 10^{38}$ and up to seven significant digits. Storage requirement is four bytes.
- **Double** — For numeric floating point values that range from -1.797×10^{308} to $+1.797 \times 10^{308}$ and up to 15 significant digits. Storage requirement is eight bytes.
- **Replication ID** — For storing a GUID that is required for replication. Storage requirement is 16 bytes.
- **Decimal** — For numeric values that range from $-9.999... \times 10^{27}$ to $+9.999... \times 10^{27}$. Storage requirement is 12 bytes.

Numeric and date/time fields can have a specified format to define how the data is displayed. Short text, long text, and hyperlink fields usually have no format. Predefined numeric field formats include:

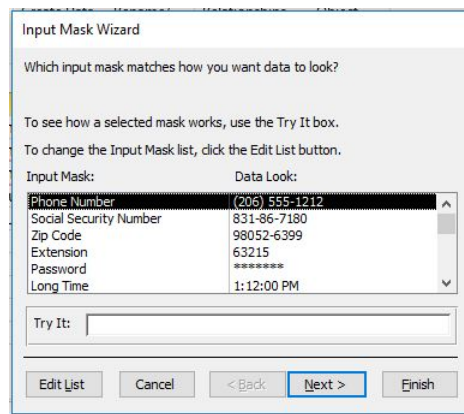
- **General Number** – (Default) Display the number as entered.
- **Currency** – Use the thousand separator; follow the settings specified in the regional settings of Windows for negative amounts, decimal and currency symbols, and decimal places.
- **Euro** – Use the euro symbol (€), regardless of the currency symbol specified in the regional settings of Windows.
- **Fixed** – Display at least one digit; follow the settings specified in the regional settings of Windows for negative amounts, decimal and currency symbols, and decimal places.
- **Standard** – Use the thousand separator; follow the settings specified in the regional settings of Windows for negative amounts, decimal symbols, and decimal places.
- **Percent** – Multiply the value by 100 and append a percent sign (%); follow the settings specified in the regional settings of Windows for negative amounts, decimal symbols, and decimal places.
- **Scientific** – Use standard scientific notation.

When it comes to date/time fields, Access automatically displays the date and time in the **General Date** and **Long Time** formats. The dates appear as, mm/dd/yyyy in the U.S. and as, dd/mm/yyyy outside the U.S. where mm is the month, dd is the day, and yyyy is the year. The time is displayed as, hh:mm:ss AM/PM, where hh is the hour, mm is minutes, and ss is seconds. Here are some other predefined formats for date/time fields:

- **Long Date** – Displays only date values, as specified by the Long Date format in your Windows regional settings. i.e. Monday, May 13, 2018.
- **Medium Date** – Displays the date as dd/mmm/yy, but uses the date separator specified in your Windows regional settings. i.e. 13-May-2018.
- **Short Date** – Displays date values, as specified by the Short Date format in your Windows regional settings. i.e. 5/13/2018.
- **Medium Time** – Displays hours and minutes followed by AM or PM. Access uses the separator specified in the Time setting in your Windows regional settings. i.e. 9:23 AM.
- **Short Time** – Displays only hours and minutes. Access uses the separator specified in the Time setting in your Windows regional settings. i.e. 9:23.

Other popular (not all inclusive) field properties that appear based on certain data types include:

- **Decimal Places** – Specifies the number of decimal places to use when displaying numbers.
- **Input Mask** – Controls how data is entered and is useful when text or date entries should follow a certain format. Access has several predefined masks. To open the Input Mask Wizard, click the ellipses (...) at the end of the Input Mask field. Popular existing input masks include the Phone Number and Zip Code masks. Typically, clicking Next to accept the default options throughout the wizard will create the most common input masks. Custom masks can be created by clicking the Edit List button. Mask characters include the digits 0 and 9. A 0 placeholder means that a digit is required, whereas a 9 means that a digit is optional. For example, the last four characters of a Zip Code are often optional, so the mask would display as 00000-9999.



Input Mask Wizard

- **Caption** – The label text that is displayed for this field by default in forms, reports, and queries. If this property is empty, the name of the field is used. Any text string is allowed.
- **Default Value** – Automatically assigns the specified value to this field when a new record is added. Popular with State or Country fields for data entry in regional uses.
- **Validation Rule** – Helps ensure accurate data through creation of an expression criterion that must be true whenever the value is added or changed in the field. Validation rules are popular for numeric and date entries, and are formed with one or more relational operators and/or logical operators. A typical rule in an Order Entry table/form might require the Quantity be at least one. The rule in the Quantity field could be ≥ 1 .

Relational Operators	
<	less than
>	greater than
<=	less than or equal
>=	greater than or equal
<>	not equal to
Logical Operators	
And	requires an entry to match both criteria
Or	requires an entry to match one criteria
Not	requires an entry not match the criteria

Operators

- **Validation Text** – If an entry breaks the rule, it is cleared from the entry box and the Validation Text is displayed in a dialog box. Highly recommended to be used in conjunction with the Validation Rule so the data entry person knows why their entry is not acceptable.
- **Required** – Specifies if data must be entered or not. Choice is Yes or No. Default is No.
- **Indexed** – Specifies whether the field has an index. There are three available values: Yes (No duplicates) – Creates a unique index; Yes (Duplicates OK) – Creates a non-unique index; No – Removes any index on the field. If users often search a table in Access or sort its records by a particular field, users can speed up these operations by creating an index for the field. Access uses indexes in a table as people use an index in a book: to find data, Access looks up the location of the data in the index. In some instances, such as for a primary key, Access automatically creates an index. At other times, it may be necessary to create an index manually.

This page titled [1.1.4: Creating a Table](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher \(Minnesota Libraries Publishing Project\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **6.4: Creating a Table** by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

1.1.5: Primary Keys and Table Relationships

 [Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Primary and Foreign Keys

Fields that are part of a table relationship are called keys. A key usually consists of one field, but may consist of more than one field. There are two kinds of keys:

- **Primary key** – To ensure that no two records in a table are identical, one field in each table is designated the primary key. A primary key consists of one or more fields that uniquely identify each record that you store in the table. Often, there is a unique identifier, such as an ID number (Employee ID, Customer ID, etc), a serial number, or a social security number that serves as the primary key. Always choose a primary key whose value will not change. A key icon appears next to the name of the primary key field. AutoNumber data type fields make good primary keys.
- **Foreign key** – A table can also have one or more foreign keys. A foreign key contains values that correspond to values in the primary key of another table. For example, you might have an Orders table in which each order has a customer ID number that corresponds to a record in a Customers table. The customer ID field is a foreign key of the Orders table.

Table Relationships

Access is a relational database management system. In a relational database, users divide information into separate, subject-based tables. Keeping data separated in related tables produces the following benefits:

- **Consistency** – Because each item of data is recorded only once, in one table, there is less opportunity for ambiguity or inconsistency. For example, storing a customer's name only once in a table about customers versus storing it repeatedly (and potentially inconsistently) in a table that contains order data.
- **Efficiency** – Recording data in only one place results in less disk space used. Additionally, smaller tables tend to provide data more quickly than larger tables. Finally, not using separate tables for separate subjects will introduce null values (the absence of data) and redundancy into the tables, both of which can waste space and impede performance
- **Comprehensibility** – The design of a database is easier to understand if the subjects are properly separated into tables.

In relational database terms, a **relationship** is a situation where multiple tables can contain related data that is linked by a common field. A relationship consists of a parent table and a child table. The child table references the parent table by having a field that matches a field in the parent table. The child's field is the foreign key, whereas the parent's field is the primary key. In a relationship, any data entered into the child's foreign key field must match a value from the parent's primary key field. By ensuring that the foreign key's data matches data in the primary key, Access ensures that all records in the child table will have an associated record in the parent table. There are three types of relationships:

- **One-to-One** – A row in table A can have only one matching row in table B, and vice versa. i.e. A baseball player can have only one Jersey Number.
- **One-to-Many** – A row in table A can have many matching rows in table B, but a row in table B can have only one matching row in table A. i.e. A baseball player can have many games played.
- **Many-to-Many** – A row in table A can have many matching rows in table B, and vice versa. This is achieved through the use of a third table (commonly called a *junction table*) that contains lookup data for both tables. Multiple players can play multiple games.

Determining the relationships between tables helps ensure that the right tables and columns exist. When a one-to-one or one-to-many relationship exists, the tables involved need to share a common column or columns. When a many-to-many relationship exists, a third table is needed to represent the relationship.

This page titled [1.1.5: Primary Keys and Table Relationships](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher](#) (Minnesota Libraries Publishing Project) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

- **6.5: Primary and Foreign Keys** by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.
- **6.6: Table Relationships** by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

1.1.6: Maintaining Records in a Database

 [Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Examine existing records in a database
- Construct new records to add to a database
- Update existing records in a database
- Demonstrate the process to delete records from a database

Most of the time, when employees maintain and work with a database, they access data that has been provided to them from another data storage area, such as the company's data warehouse. The tasks in the analysis are to import the data, review the data to ensure that it is brought into the database correctly, and then use the data. Sometimes, however, users need to complete other tasks to ensure that the database processes the data correctly.

Link to Learning

Data integrity is a cornerstone of the digital organization. Without data that can be trusted, companies cannot transform themselves to be successful in the digital world. Transformation comes with two data-related problems—how companies store and access data and the quality of that data. For an overview of these issues, read this [article on data integrity in business](#) and think about the ethical requirements for the data professional and organizational leadership and how they need to balance the quality, security, and accessibility of data within the organization.

Reviewing Records in a Database

It is possible to view records in database tables but be sure to not change the records.

In a table, records are shown on a datasheet, with each column equaling a field name (a value that all records have) and each row representing an individual record. On the datasheet, there are commands that let you move between records. You can also use the scroll bars to look at different parts of the table. Although using the datasheet to view records can be useful in limited situations, databases typically hold a great deal of data, so in general, querying the table will be a much more efficient way of finding the records you need (Figure 13.30).

ProductID	ProductCategoryID	Status	Click to Add
5B03	N05B	active	
5B04	N05B	active	
5B05	N05B	active	
5B06	N05B	on hold	
5B07	N05B	retired	
5B08	N05B	active	
5B09	N05B	active	
5B10	N05B	active	
5B11	N05B	on hold	
5B12	N05B	retired	
5B13	N05B	active	
5B14	N05B	active	
5B15	N05B	active	

Figure 13.30: Reviewing records in a datasheet provides a broad overview of the data, while querying allows you to filter and retrieve specific information that answers particular questions or meets specific criteria. Queries are used to extract precise and targeted data from a database, whereas a datasheet presents a more comprehensive view of the data. (Used with permission from Microsoft)

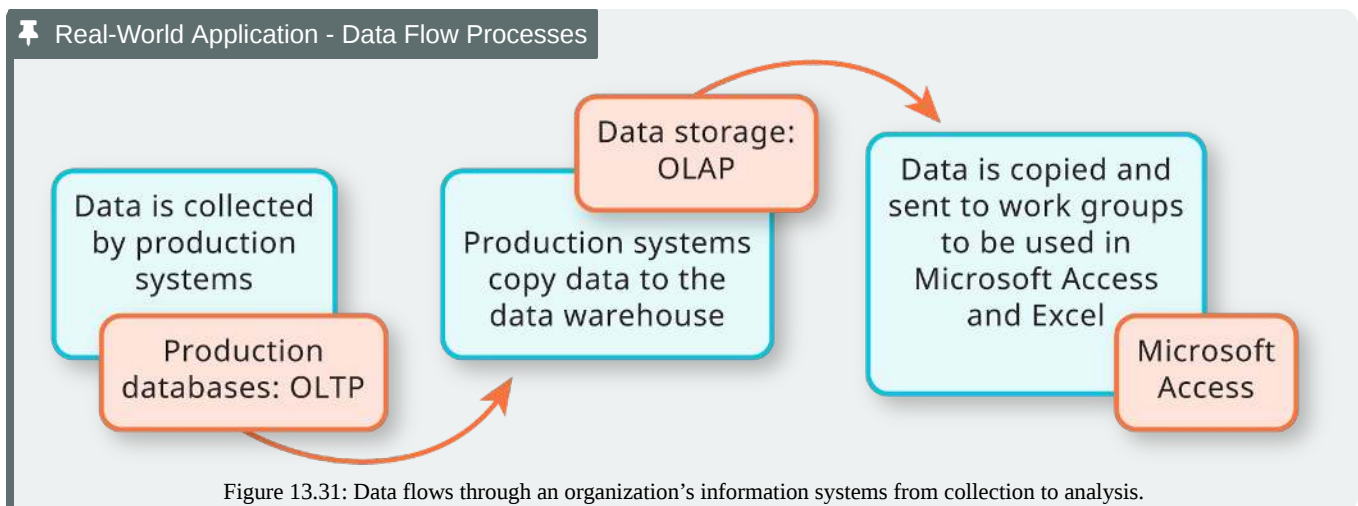


Figure 13.31: Data flows through an organization's information systems from collection to analysis.

Data moves across the data landscape from transactional production databases, including sales, customer service, human resources, and other systems, to storage databases such as the data warehouse, which can hold historical information (online transaction processing (OLTP)). Once housed in a data store, datasets can be made available to databases like Access and spreadsheets like Excel for local use and analysis (online analytical processing (OLAP)). Why would it be important to keep in mind how and when data was collected from a production system when performing analysis on that data in Access?

If you need to look at records in Datasheet View, there are commands at the bottom of the screen that will help you move between them. When a datasheet view is open, these commands appear automatically in Access, on the bottom left side of the screen.

The record navigation buttons open any time that records are shown. As you work in the database, you will see these buttons in Datasheet View and in certain form views. The command buttons include jumping to the first or last record in the series, navigating

one record ahead or back, and adding a new record, as you can see in Figure 13.32. You can also add a filter or use the search feature to find a specific field value.

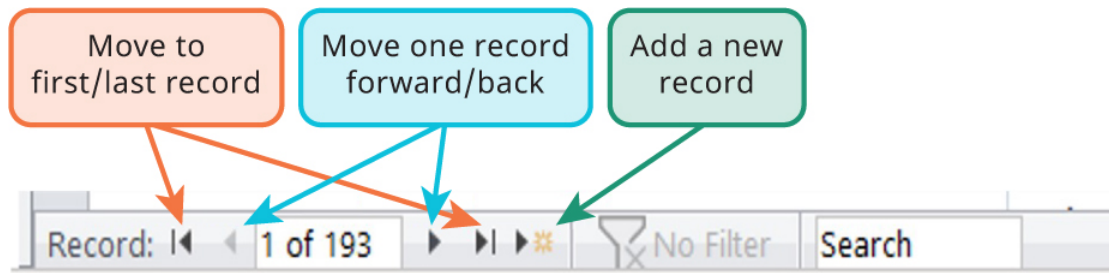


Figure 13.32: Using the record navigation buttons to progress through a datasheet can save you a lot of time. (Used with permission from Microsoft)

Constructing New Records

Databases are historical in that they record past details about an organization, but they also need to be updated as time goes on. There are a variety of ways to update a database, including both manual and automated updates. Most of the time, when you use Access, you will import a dataset or file from another system, such as the organization's data warehouse.

Adding a Record Manually

If you need to add a record into a database, you can either add the record to the table manually or import it from another database. To add a record manually:

1. Open the table in Datasheet View.
2. On the Home tab, click the command New in the Records group.
3. Fill in the information, using Tab or the left and right arrow keys to move through the record.
4. Once you are done, use Tab or the up and down arrows to move to the next record, or press Shift+Enter (or any version of Save that saves a record).

Importing a Record

Creating files that can be imported to other systems is a hallmark of a digital organization. When working with Access, we are likely to import files from other, larger systems. Data is collected in business units across an organization, and using that data is key to an effective business. Business units routinely transfer files between systems, which in turn lets units analyze and create actionable information.

Most systems can create comma-delimited (CSV) files, which use commas as delimiters or breakpoints to tell Access (or any other program) where data should be separated. CSV files can be read into most Microsoft products and other popular software packages. There are also other formats designed for importing and exporting files, which use another type of formatting to delimit, or separate, data.

Access can import a variety of file types, including Excel, CSV, and XML, as well as files specific to databases and other systems or software products. Even so, CSV is one of the most common formats for data importing (Figure 13.33).

Before importing a file, be familiar with the file and its contents. To make the import process easier, you may want to rename fields so that they match in the file and the database.

Once your file is ready, you will likely want to import it to a new table so you can catch and correct any errors without corrupting the data in your other data tables. Depending on the specific type of work you are doing, you can either set up a table ahead of time or import the data into a new table.

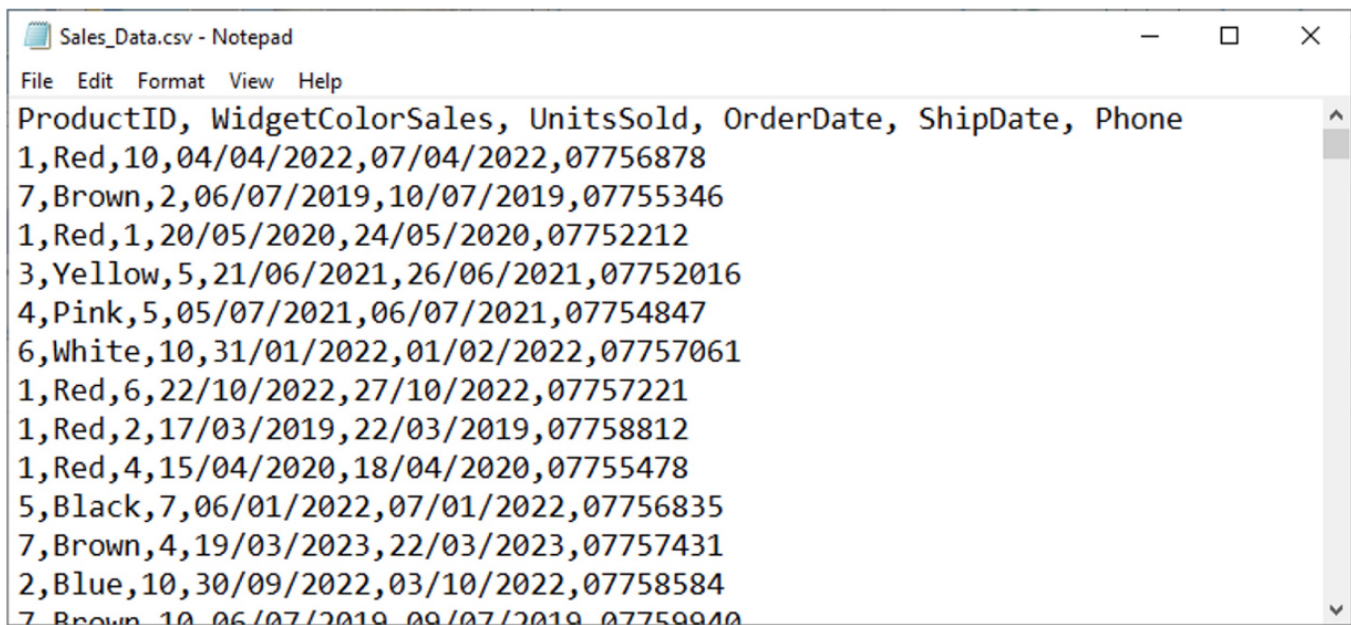
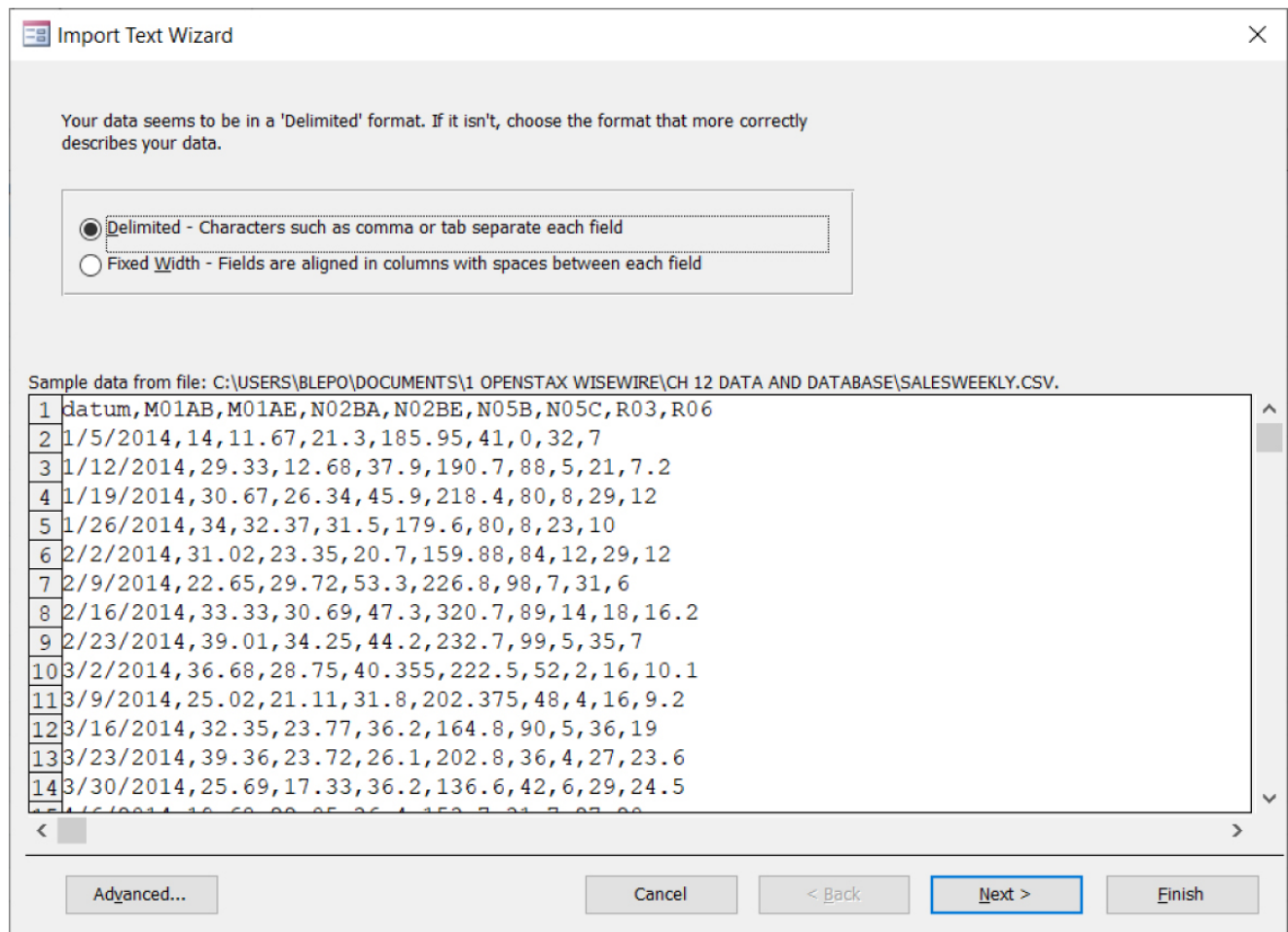


Figure 13.33: A CSV (comma-separated values) file is a plaintext file format commonly used for storing tabular data, such as spreadsheets or databases. (Used with permission from Microsoft)

Note: Access has the ability to import data directly into existing tables manually through the Import command and by connecting automatically with a database or system.

Follow these steps in importing a file:

1. With your database open, select the External Data tab in Access.
2. Choose the New Data Source button.
3. Choose the file type you want to work with—for example, text. A dialog box will open.
4. Select the file you want to import.
5. Choose the import type that fits best with your project. If you are concerned about an import, try importing the data into a new table to be sure you have all of the fields and parameters set correctly.
6. Click on OK; the import screen will open (see Figure 13.34).



Import Text Wizard

Your data seems to be in a 'Delimited' format. If it isn't, choose the format that more correctly describes your data.

☒ Delimited - Characters such as comma or tab separate each field
☐ Fixed Width - Fields are aligned in columns with spaces between each field

Sample data from file: C:\USERS\BLEPO\DOCUMENTS\1 OPENSTAX WISEWIRE\CH 12 DATA AND DATABASE\SALESWEEKLY.CSV.

1	datum, M01AB, M01AE, N02BA, N02BE, N05B, N05C, R03, R06
2	1/5/2014, 14, 11.67, 21.3, 185.95, 41, 0, 32, 7
3	1/12/2014, 29.33, 12.68, 37.9, 190.7, 88, 5, 21, 7.2
4	1/19/2014, 30.67, 26.34, 45.9, 218.4, 80, 8, 29, 12
5	1/26/2014, 34, 32.37, 31.5, 179.6, 80, 8, 23, 10
6	2/2/2014, 31.02, 23.35, 20.7, 159.88, 84, 12, 29, 12
7	2/9/2014, 22.65, 29.72, 53.3, 226.8, 98, 7, 31, 6
8	2/16/2014, 33.33, 30.69, 47.3, 320.7, 89, 14, 18, 16.2
9	2/23/2014, 39.01, 34.25, 44.2, 232.7, 99, 5, 35, 7
10	3/2/2014, 36.68, 28.75, 40.355, 222.5, 52, 2, 16, 10.1
11	3/9/2014, 25.02, 21.11, 31.8, 202.375, 48, 4, 16, 9.2
12	3/16/2014, 32.35, 23.77, 36.2, 164.8, 90, 5, 36, 19
13	3/23/2014, 39.36, 23.72, 26.1, 202.8, 36, 4, 27, 23.6
14	3/30/2014, 25.69, 17.33, 36.2, 136.6, 42, 6, 29, 24.5

Advanced... Cancel < Back Next > Finish

Figure 13.34: Importing a file using the Import Text Wizard allows the user to select the correct format for their file. (Used with permission from Microsoft)

- The Import Wizard, which is a part of the import process, will open and ask you to choose how your text file is structured. (Hint: Most text files are delimited, and CSV files use commas as the delimiting or separating character.) Once you choose whether the file is delimited or fixed width (broken up based on a certain number of characters), click Next to continue.
- On the next screen, set your delimiter (comma for a CSV file) and click the first row that contains field names (assuming your first row is titles or labels). Then preview the data and, when you are sure it is correct, click Next to move to the next screen. See Figure 13.35.

Figure 13.35: For files that use a delimiter, the Import Text Wizard requires that the user define which delimiting character is used. (Used with permission from Microsoft)



Figure 13.36: The Import Text Wizard allows you to specify various field properties, such as name and data type, before importing the file to Access. (Used with permission from Microsoft)



Microsoft Access recommends that you define a primary key for your new table. A primary key is used to uniquely identify each record in your table. It allows you to retrieve data more quickly.

☒ Let Access add primary key.
 ☐ Choose my own primary key.
 ☐ No primary key.

ID	datum	M01AB	M01AE	N02BA	N02BE	N05B	N05C	R03	R06
1	1/5/2014	14	11.67	21.3	185.95	41	0	32	7
2	1/12/2014	29.33	12.68	37.9	190.7	88	5	21	7.2
3	1/19/2014	30.67	26.34	45.9	218.4	80	8	29	12
4	1/26/2014	34	32.37	31.5	179.6	80	8	23	10
5	2/2/2014	31.02	23.35	20.7	159.88	84	12	29	12
6	2/9/2014	22.65	29.72	53.3	226.8	98	7	31	6
7	2/16/2014	33.33	30.69	47.3	320.7	89	14	18	16.2
8	2/23/2014	39.01	34.25	44.2	232.7	99	5	35	7
9	3/2/2014	36.68	28.75	40.355	222.5	52	2	16	10.1
10	3/9/2014	25.02	21.11	31.8	202.375	48	4	16	9.2
11	3/16/2014	32.35	23.77	36.2	164.8	90	5	36	19
12	3/23/2014	39.36	23.72	26.1	202.8	36	4	27	23.6
13	3/30/2014	25.69	17.33	36.2	136.6	42	6	29	24.5
14	4/6/2014	19.68	22.05	36.4	153.7	31	7	27	20

Advanced...

Cancel

< Back

Next >

Finish

Figure 13.37: You should select a primary key for any imported tables. This primary key can serve as a unique identifier for each record in your database. (Used with permission from Microsoft)

The final Import Wizard screen will allow you to name your new table. Be sure to use a descriptive name that distinguishes it from the tables you already have in the database.

When you are done, click Finish to view your new table. If your table has errors (e.g., if data cannot be imported because of mismatches in the field type), a new table containing the errors will appear in your list, as Figure 13.38 shows. If you find any errors, you may want to delete your imported tables and walk through these steps again.

Import & Link Export

All Access Objects

Search...

Tables

- Agent
- Booked viewings
- Buyer ID
- Property ID
- Staff_2014_ImportErrors**

Forms

- Moving On Up-Property

Error	Field	Row
community	ID	1
Null value in a	ID	1
Type Conversion	ID	2
Null value in a	ID	2
Type Conversion	ID	3
Null value in a	ID	3
Type Conversion	ID	4
Null value in a	ID	4
Type Conversion	ID	5
Null value in a	ID	5
Type Conversion	ID	6
Null value in a	ID	6
Type Conversion	ID	7
Null value in a	ID	7
Type Conversion	ID	8
Null value in a	ID	8

Record: 1 of 16 No Filter Search

Figure 13.38: A table with errors will still be imported, but the errors will be identified in a new table with the phrase “ImportErrors” in the title. (Used with permission from Microsoft)

Link to Learning

Data management is a bigger concern than just tables and fields—data professionals at all levels need to understand the data landscape and how organizations can manage it. This [article on data lineage](#) provides an overview of some of the challenges and considerations in data management.

Editing or Deleting an Existing Record

Once you locate the record you want to update, you can edit it. To edit a record:

1. Open the table in Datasheet View (or Form View).
2. Click on the field you would like to edit.
3. Place your cursor where you want the information to appear.
4. Enter your text.
5. Use Tab to move across your record.

As you work, Access will save your changes whenever you move your cursor from one field to another. If you want to save the record manually to be sure your data is saved, you can either click the Home tab then click Save in the Records group, or press Shift+Enter on the keyboard.

When you are updating a record, you may see symbols appear that share the status of that record, as Figure 13.39 shows. As you manually edit records, pay attention to these data entry symbols.

Symbol	Description
	Record that is being edited and is currently saved
	Record that is being edited and is not saved
	New record ready for information
	Primary key field
	Locked record; cannot be edited (often due to being in use already)

Figure 13.39: These are some of the common icons you will use in data entry. (Used with permission from Microsoft)

Deleting a Record

If you need to delete a record or set of records for any reason, you can do so through either the datasheet or the form.

1. Open the table in Datasheet View or Form View.
2. Select the record by clicking on the record selector. To select multiple records, either click and drag across record selectors or use Ctrl+Click.
3. Press Delete or select Home>Records>Delete.

This page titled [1.1.6: Maintaining Records in a Database](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.4: Maintaining Records in a Database** by [OpenStax](#) is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

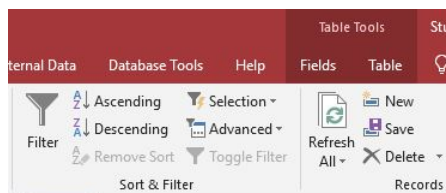
1.1.7: Entering and Editing Records

[Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Entering records directly into Datasheet View is probably the most intuitive way to enter data into a database. In Access, this is common because many Access databases are considered single-user databases. With multi-users databases, data integrity policies would discourage most users from having editing rights directly to the tables. Therefore, in these databases, the more common method for entering and editing records in a database would be via a form. Nonetheless, entering records directly into Datasheet View has some advantages, including:

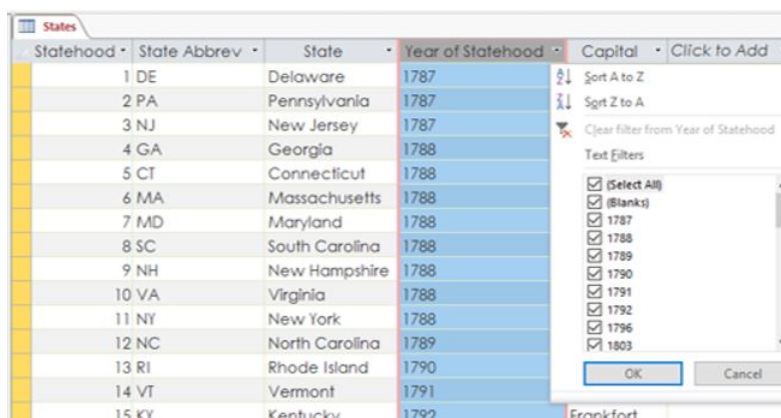
- Users can look at their data in columns and rows in a datasheet without using any special formatting.
- When a user opens a table in Datasheet View it resembles an Microsoft Excel worksheet, and the user can paste or add data in one or multiple fields. Data can be copy and pasted from a Word table or Excel worksheet into Access.
- Printing a table's datasheet in Portrait or Landscape orientation. Use Print Preview to change the orientation and margins before printing. Output can also be sent to a PDF file.
- Users don't need to explicitly save their data. Access commits user's changes in the table when they move their cursor in the same row towards a new field, or if they move their cursor towards another row.
- Sorting and filtering data is very similar to Microsoft Excel.

Access allows users to **sort** data in Datasheet View one iteration at a time in either **Ascending** or **Descending** order. Ascending order is also referred to as *alphabetical* order when the sort is based on a text field, *chronological* order when the field is a date/time data type or *numerical* order when the data type of the field is Number. Sorting a datasheet does not affect the order that the records are displayed in a form.



Sort & Filter

Displaying records based on specified criteria is called **filtering**. When a filter is applied, records that do not meet the specified criteria are hidden from view until the filter is removed. To filter records based on the data in a field, click in the field to base the filter on and then click on the Filter button on the ribbon or click the drop-down triangle to display a menu similar to the graphic below.

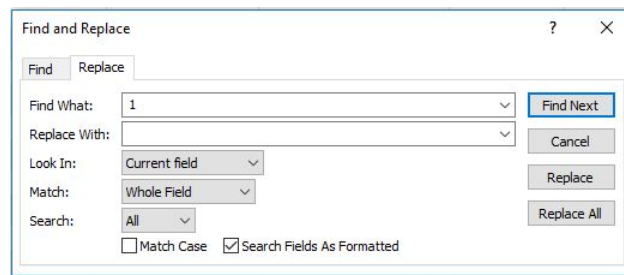


Statehood	State Abbrev	State	Year of Statehood	Capital	Click to Add
1	DE	Delaware	1787		
2	PA	Pennsylvania	1787		
3	NJ	New Jersey	1787		
4	GA	Georgia	1788		
5	CT	Connecticut	1788		
6	MA	Massachusetts	1788		
7	MD	Maryland	1788		
8	SC	South Carolina	1788		
9	NH	New Hampshire	1788		
10	VA	Virginia	1788		
11	NY	New York	1788		
12	NC	North Carolina	1789		
13	RI	Rhode Island	1790		
14	VT	Vermont	1791		
15	KY	Kentucky	1792	Frankfort	

Filtered table

To remove a filter, clear the check boxes, and click OK or click the Filter  button again or click the Advanced button on the ribbon and choose Clear All Filters.

Most databases contains hundreds, if not thousands, of records. Therefore, when a table becomes so large that the records and data do not fit in the datasheet window without scrolling, the Find feature becomes very practical.



Find and Replace

In Datasheet view, use the Find command, from the Home tab, to quickly locate specific records using all or part of a field value entered in the Find What: field. If in the States table above, if the user entered the value North in the Find What field, Access would find two records for North Carolina and North Dakota, but only if the Match: value was changed from Whole Field to Any Part of Field or Start of Field. Using **wildcard characters** can be extremely helpful if the user does not know the exact spelling of the value they are looking for.

Wildcard Character	Usage
?	Used to match any single character; to search for a three-letter word that starts with "d" and ends with "g", use d?g.
*	Used to match any number of characters; to search for a word that ends in "son", use *son.
#	Used to match any single numeric character; to search for a three-digit number that starts with "3", use 3##.

Wildcards

A wildcard character can replace a single character or multiple characters, which can be both text and numbers. Clicking the Replace tab will allow the user to replace the data in the Find What: field with the data entered in the Replace With: field. This could be very useful when updating name changes, such as when Kentucky Fried Chicken changed their name to KFC in 1991.

This page titled [1.1.7: Entering and Editing Records](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher \(Minnesota Libraries Publishing Project\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **6.7: Entering and Editing Records** by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

1.1.8: Querying a Database

 [Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Explain the uses of queries to explore data in a database
- Build and edit a query in Access through the query grid
- Establish a relationship between tables with a simple query

WorldCorp collects data constantly in all its departments. The Human Resources Department maintains employee records; the Maintenance Department monitors equipment, repair, and inspection records; and the Accounting Department keeps records on all purchases, to name just a few departments. The data collected in all of these areas can inform decisions, as long as it is accessible and analyzable.

The WorldCorp product and sales division constantly updates customer records, processes sales, and maintains inventory levels. It has a data warehouse that powers its transactions, which is continuously working in real time. This information is key to its success, but it cannot work with it directly because that would mean turning the system off. Instead of taking a production system offline for analysis, you will work with a dataset based on last week's work.

Working with a dataset can be complicated, and as the datasets get more complex with more records, you will need to be able to query or ask questions using Access. Because Access runs on your computer against a database file, it is quick and responsive and integrates well with other Microsoft programs like Excel for visualization and Word and PowerPoint for communication.

Understanding Queries

Queries allow you to ask questions of a database or take an action within a database. To write a query, you need to know the question you want to answer. For example, you might need a listing of all products sold over the past quarter, a list of equipment that needs to be inspected, or a list of products in the company's inventory. Once you establish what you want to know, you can then write a query that will ask the database to share the correct information.

A query is really a question written in computer syntax. The query collects data that meets your specifications by running against the table(s) in your database. A query can be simple—for example, returning all values in a field in the table, such as product ID number. A query can also be complex, limiting the dataset and results to precisely what you are looking for. Expressions that a database management system uses to limit the results of a query based on matching field values are called criteria. Adding criteria enables you to control what data you see so that you can identify the records you need and omit the ones that do not meet your needs. In turn, this makes it easier to answer questions and visualize the data you need.

Before you create a query in Access, you should have a strong understanding of what you want from your dataset. For example, WorldCorp could be interested in discovering how their latest product has been selling based on data in their transactional database. To accomplish this, you might need to look for sales that match certain criteria, such as sales for a specific product identification number. Knowing this in advance and putting it in writing will help you craft useful queries. When planning a query, you will want to identify the following:

- The fields you want
- The value(s) you expect in each field
- Any calculations or changes you want from the data
- How you want the data displayed (e.g., in order by date)

Although the steps are the same in building both simple and complex queries, the planning process becomes more important the more complex your queries are.

Queries can be designed for a single use or built for ongoing work. Ad hoc queries typically answer a specific question and are needed only at the moment you create them. Once you are done with an ad hoc query, you will not use it again, so it does not need to be stored or saved. Other queries, called predefined queries, are set up and saved. Predefined queries are used to automate data

management tasks that meet specific organizational needs. An example of a predefined query is a sales report or top-selling product report that you frequently run without change (or with only minor changes). In this case, you will want to save the query to make your work easier. You might even want to set up a screen with all of your predefined queries.

The Purpose of Queries in Databases

Queries are an essential part of making a database accessible. Imagine trying to make sense of hundreds, thousands, or even more records without a tool to help you. It is simply not possible. The more complex your dataset, the more important it is to use queries and reports to retrieve data. Most databases have too many records to work with unless you have well-written queries. A well-written query is specific, organized, and comprehensive, meaning that it contains everything you need.

Spotlight on Ethics - Protecting the Privacy of Data

The Target data breach in 2013 was a significant security incident, resulting in the theft of forty million credit and debit records and seventy million customer records. The breach demonstrated that such incidents can severely undermine customer trust and confidence in data security. Customers concerned about the potential leakage of their data became wary of Target, leading to decreased sales reputational damage.

The breach also highlighted the need for companies to ensure the security of their vendors. Target's compromise, for example, originated from a third-party vendor. Cybersecurity is a weakest link proposition: Organizations must consider the security of all parties involved in their data ecosystem.

Although Target's handling of the breach included prompt notification to customers, it was clear that additional precautions could have been taken that would have made the attack more challenging for cybercriminals. The incident highlights the need for proactive security measures and robust disaster preparedness plans. Furthermore, the breach had financial implications that went beyond the settlement costs; Target experienced reduced earnings and a decline in customer patronage. The ultimate cost of the breach was estimated to be well over \$200 million.

The Target data breach serves as a reminder for organizations to prioritize data security, protect customer information, engage in proactive defenses, and have strategies in place to restore customer trust if a breach occurs.

See this [coverage of the Target data breach](#) and think about how your data may be stored and used by companies and websites *and* also about how you, as a professional, need to manage data to protect it and the people who have shared it with you and your company.

You are most likely to use queries to retrieve a specific record or set of records, but queries have other functions as well. For example, they can summarize data and perform calculations. You can even set up queries that show you the most current data in the database each time you refresh your records.

Designing Queries

Before building a query, you need to take some time to think through how the query will work and what it will accomplish. Start with a question, such as one of the following examples:

- What does inventory look like right now?
- What sales were made last week?
- What were the top products sold, based on volume, last quarter?

These questions demonstrate in detail how to structure the query. Once we have the question in mind, we can then consider other items that inform our query design: what data is available, where it is located, how it may need to be processed (e.g., sorted, filtered, or calculated) to get to the answer, and how the output should be organized (e.g., sorted by date).

This design, or model, tells you how to create the query in your software. Once you have created the query in Access, you will need to test it by running it a few times to make sure it returns the values you are looking for. You will likely revise your queries as you test them. Remember that the more complex a query is, the more revision and testing you will need to do.

A query based on your question or need for information contains specific components. In Access, these are as follows:

- The name of the query (if you will be saving it), which appears at the top of the query screen
- The field in which your data is stored
- The table in which the field is located

- Sort, which determines the column that will be used to sort the query (e.g., in numerical order by product ID)
- Criteria (text that can be used to limit the results of a query)
- Or, which allows you to create more complex statements

When you create a query in Access through the Query Wizard (Figure 13.12), these areas will be filled in by the wizard. If you need to modify a query or want to create one without using the wizard, you will use the query grid to set up the query. You learn more about this later in the section on Editing Queries with the QBE Grid and in the chapter on Advanced Database Use.

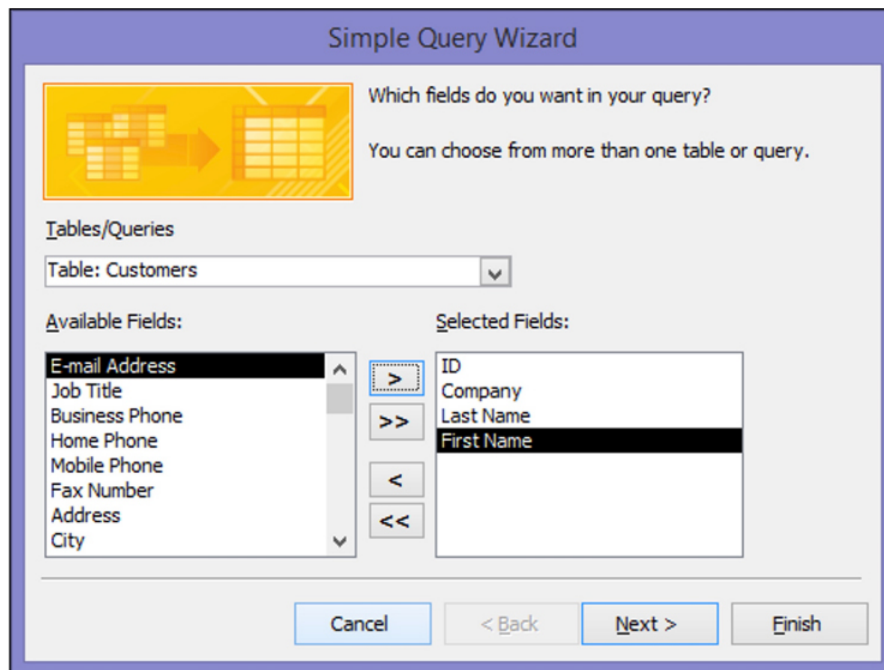


Figure 13.12: The Query Wizard guides users through the process of creating database queries step by step. It provides a graphical interface to help users select tables, fields, criteria, and sort options, making it easier to generate complex queries without writing code manually. (Used with permission from Microsoft)

Using these areas of the query allows you to accomplish multiple tasks, including choosing the data you want to include, omitting data that is not of interest to you, sorting data, and even combining data from multiple tables.

Running a Simple Query

A simple query is one that finds and displays information. Very simple queries may show all the data in the database, in columns, or all the information in one specific field or column. Once a query has been created and saved, you can run the query whenever it is needed to return the updated results. Each saved query has a name associated with it, and typically these names follow a specific pattern that helps you to differentiate between them. For example, a version of a customer list sorted by last name and the same list sorted by state would be named qryCustomerLastNameState.

In the Access database with queries, you will see a list of saved queries in the navigation section of your database. For example, in the database shown in Figure 13.13, you can see four queries under the Queries heading: ProductCategoryList, ProductList with client information, ProductsACTIVE, and ProductsALL. These simple queries are already created; to run them, simply double-click on the name of the query you want.

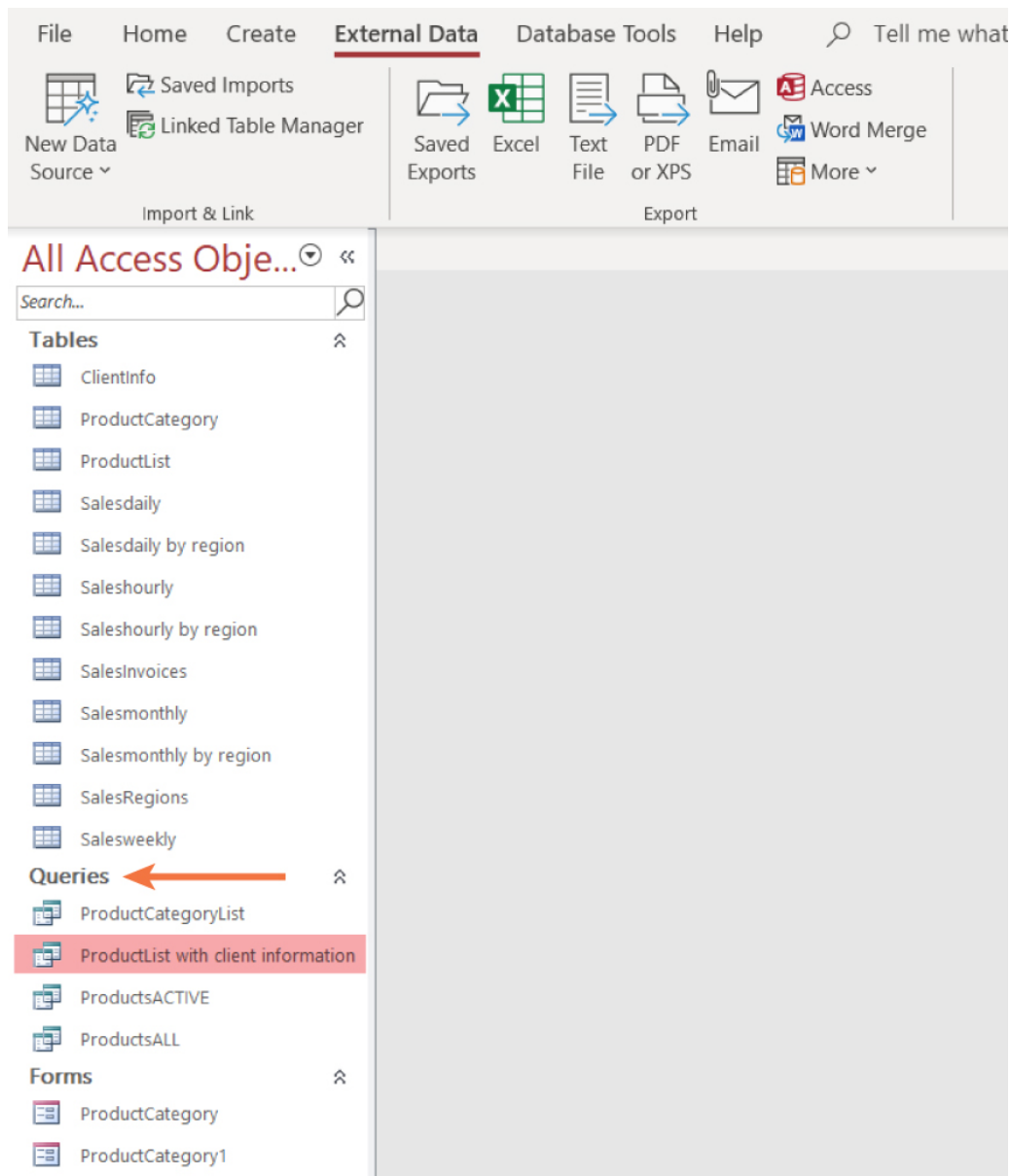


Figure 13.13: Use Saved Queries to prevent repetitive query design in Access. Here, the saved query ProductList with client information is selected. (Used with permission from Microsoft)

Once you double-click on a query name, it will open and show you the results table. The query will generate updated results every time it is opened, allowing users to use a simple query over and over, even as data is refreshed in your database.

When queries—or any database objects—are open, you will see the name of the object on a tab above your work area. To close a query, click on the X to the right of the object's name.

Switching Views

Access can switch between the Design View of the query and the results of the query. For queries, you will frequently change views to move back and forth between the underlying query structure and the results of running the query. When you look at the structure in Layout View, you can modify how the query works. When you go to SQL View, you are also able to modify the query structure (see Figure 13.14).

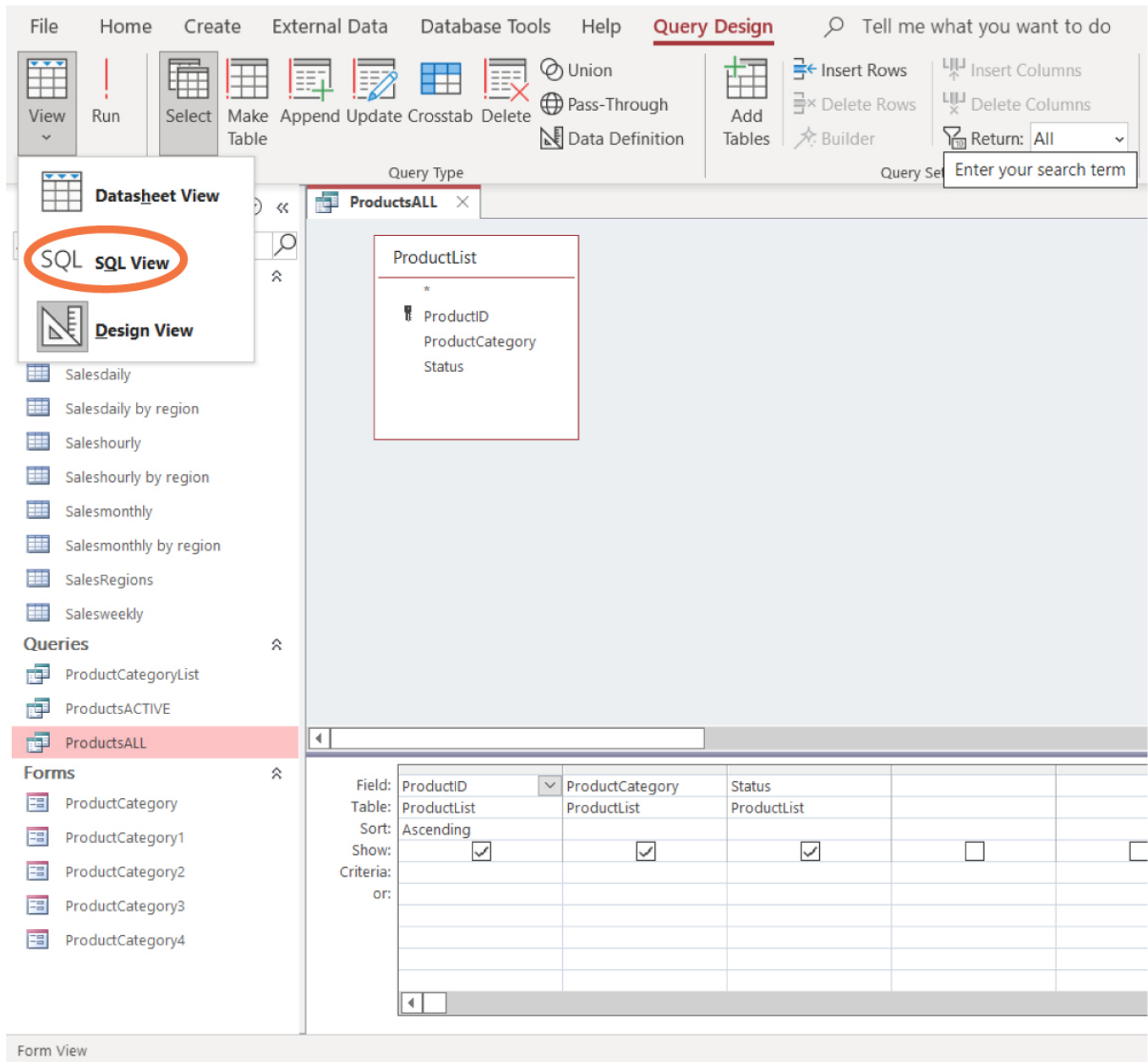


Figure 13.14: Viewing a query in SQL View allows you to see and edit the underlying SQL code that defines the query's structure and criteria. (Used with permission from Microsoft)

To change views, first click on the View command on the ribbon, and then choose the view you would like to see. You can choose from three views:

- Datasheet View shows you the results of the query in table form.
- SQL View shows the query in structured query language form.
- Design View shows the query in the form of a QBE (query by example) grid.

Using these views, you can see your query in multiple ways, first visualizing the query and then seeing the results quickly.

Query Walkthrough

In today's fast-paced business landscape, staying ahead of the competition requires a deep understanding of your products and their performance. WorldCorp's state-of-the-art sales and marketing database is designed to maximize their sales potential. At the core of this powerful database lies the ability to perform queries that deliver invaluable insights. A query in WorldCorp's sales and marketing database returns a list of products, including the product ID code, product category code, and notes (if any are present). Notice that there is only one table, called the Product List, and that all three fields are used in the query.

First, Figure 13.15 shows the query in Design View. Note that the list is sorted in ascending order based on the product ID. The result of the query is shown in Figure 13.16.

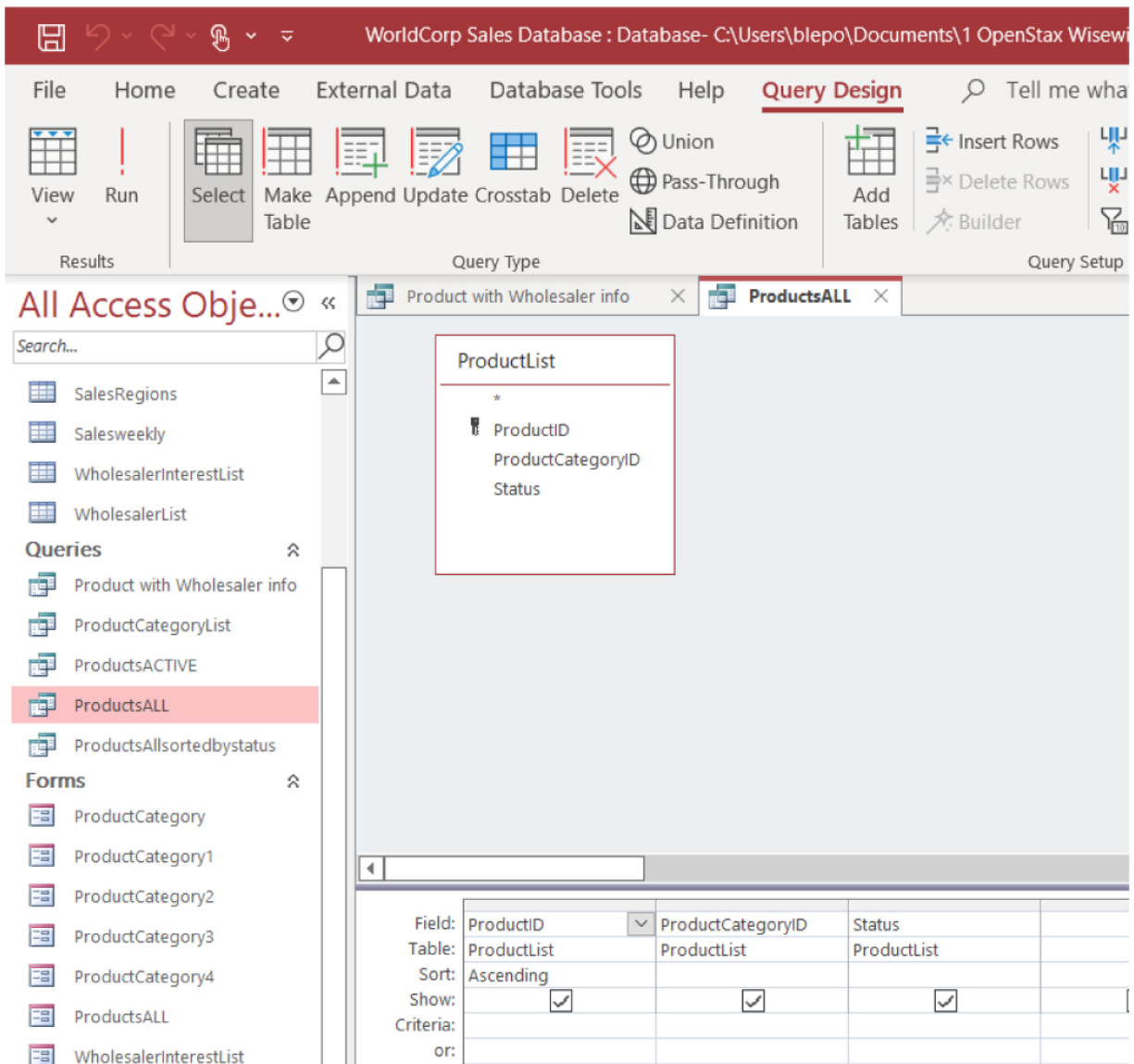
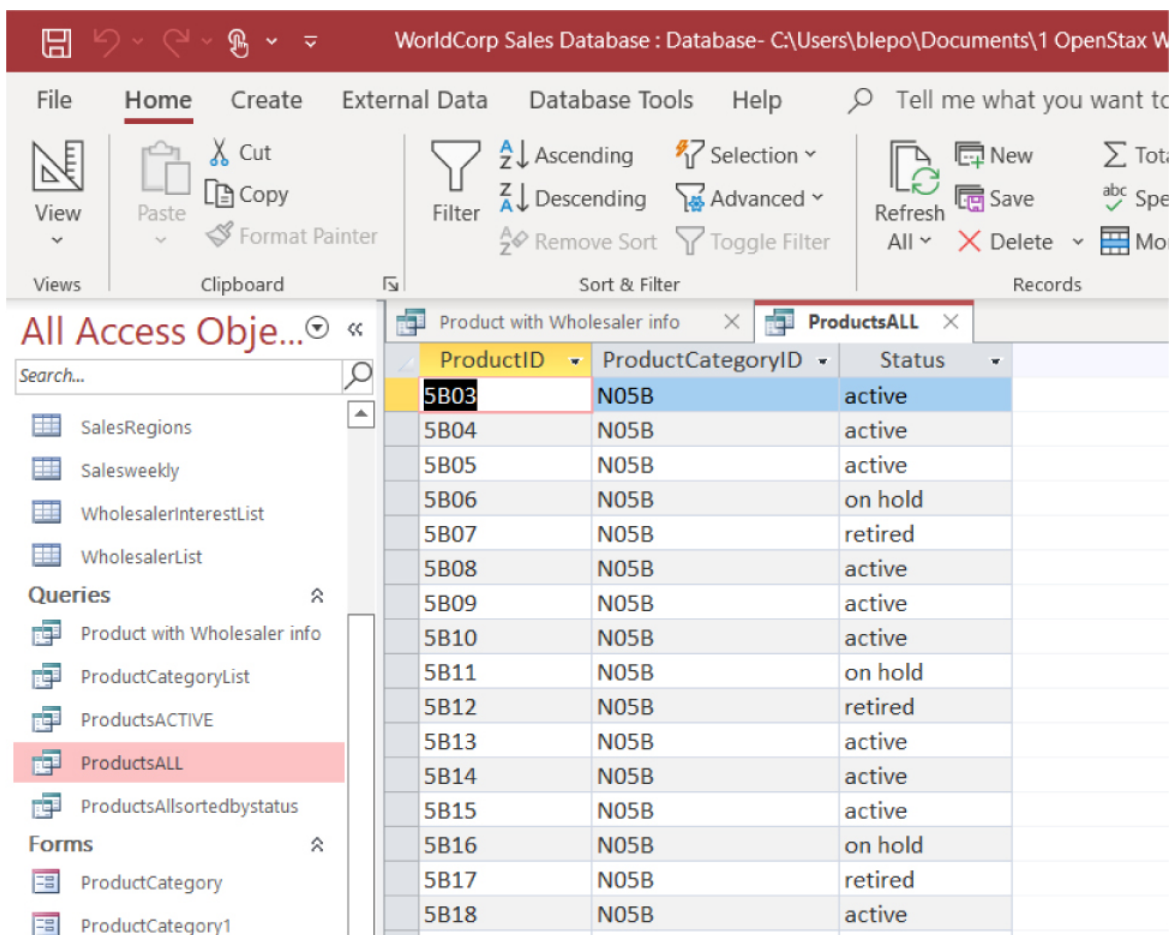


Figure 13.15: Using a query creates a list of all products found in a database. (Used with permission from Microsoft)



ProductID	ProductCategoryID	Status
5B03	N05B	active
5B04	N05B	active
5B05	N05B	active
5B06	N05B	on hold
5B07	N05B	retired
5B08	N05B	active
5B09	N05B	active
5B10	N05B	active
5B11	N05B	on hold
5B12	N05B	retired
5B13	N05B	active
5B14	N05B	active
5B15	N05B	active
5B16	N05B	on hold
5B17	N05B	retired
5B18	N05B	active

Figure 13.16: The result of the query will return everything in the database that fits the criteria you included in your query fields. (Used with permission from Microsoft)

Now, let's look at a query that uses more than one table. In this example, the query returns a list of products with information about wholesalers who have showed interest in those products. Three tables are used:

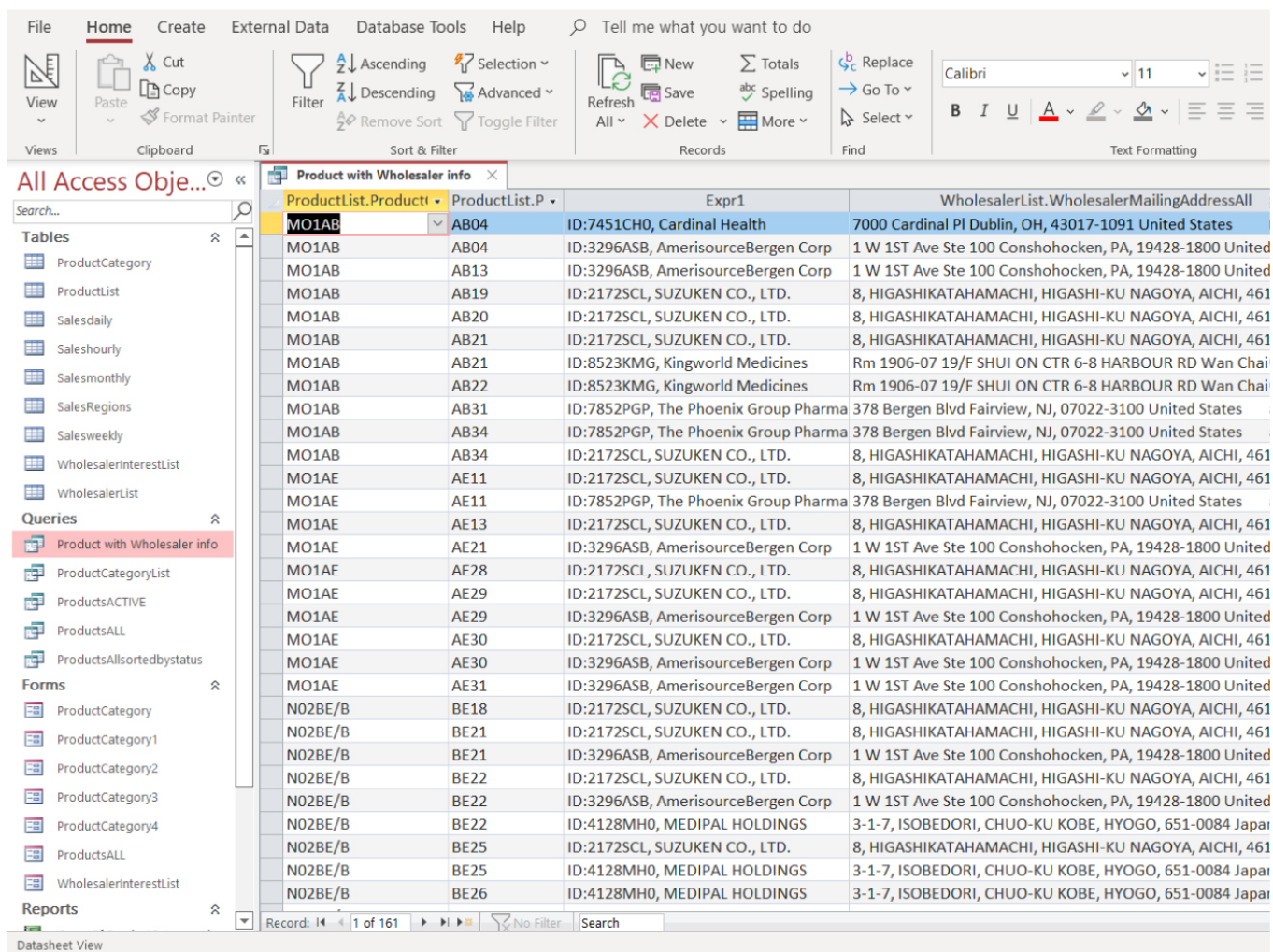
- ProductList: list of products that WorldCorp offers
- WholesalerList: list of wholesalers that WorldCorp works with
- WholesalerInterestList: table that includes ProductID (from ProductList) and WholesalerID (from WholesalerList)

See that the tables are joined by the fields ProductID and WholesalerID. If you used only the WholesalerInterestList table, you would not be able to display the product category or the wholesaler name—information needed to make the results of the query useful for the sales and marketing team. If you look at the grid in Figure 13.17, you will also see that the table uses sorting and an expression (a combination of two fields displayed together). The results are shown in Figure 13.18.

The screenshot shows the Microsoft Access Query Design view. The title bar indicates the current database is 'Product with Wholesaler info'. The ribbon at the top is set to 'Query Design'. The design grid contains three tables: 'ProductList', 'WholesalerInterestList', and 'WholesalerList'. 'ProductList' has fields 'ProductID', 'ProductCategoryID', and 'Status'. 'WholesalerInterestList' has fields 'WholesalerID' and 'ProductID'. 'WholesalerList' has fields 'WholesalerID', 'WholesalerName', 'WholesalerRegion', 'WholesalerAdd1', and 'WholesalerAdd2'. Relationships are shown as lines connecting 'ProductID' in 'ProductList' to 'ProductID' in 'WholesalerInterestList', and 'WholesalerID' in 'WholesalerInterestList' to 'WholesalerID' in 'WholesalerList'. The field list at the bottom shows the following fields being added to the query:

Field:	Table:	Sort:	Show:
ProductCategoryID	ProductList	Ascending	☒
ProductID	ProductList	Ascending	☒
Expr1: "ID:" + [Wholesale			☒
WholesalerMailingAddr	WholesalerList		☒
Status	ProductList		☐

Figure 13.17: A query across several tables can create a sorted list of products from the database by combining data from more than one table. (Used with permission from Microsoft)



The screenshot shows the Microsoft Access interface with a query named 'Product with Wholesaler info' displayed in Datasheet View. The query results are shown in a table with the following columns: ProductList.ProductID, ProductList.ProductName, Expr1, and WholesalerList.WholesalerMailingAddressAll. The data is sorted by ProductID in ascending order.

ProductList.ProductID	ProductList.ProductName	Expr1	WholesalerList.WholesalerMailingAddressAll
MO1AB	AB04	ID:7451CH0, Cardinal Health	7000 Cardinal PI Dublin, OH, 43017-1091 United States
MO1AB	AB04	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
MO1AB	AB13	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
MO1AB	AB19	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AB	AB20	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AB	AB21	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AB	AB22	ID:8523KMG, Kingworld Medicines	Rm 1906-07 19/F SHUI ON CTR 6-8 HARBOUR RD Wan Chai
MO1AB	AB22	ID:8523KMG, Kingworld Medicines	Rm 1906-07 19/F SHUI ON CTR 6-8 HARBOUR RD Wan Chai
MO1AB	AB31	ID:7852PGP, The Phoenix Group Pharma	378 Bergen Blvd Fairview, NJ, 07022-3100 United States
MO1AB	AB34	ID:7852PGP, The Phoenix Group Pharma	378 Bergen Blvd Fairview, NJ, 07022-3100 United States
MO1AB	AB34	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE11	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE11	ID:7852PGP, The Phoenix Group Pharma	378 Bergen Blvd Fairview, NJ, 07022-3100 United States
MO1AE	AE13	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE21	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
MO1AE	AE28	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE29	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE29	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
MO1AE	AE30	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
MO1AE	AE30	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
MO1AE	AE31	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
N02BE/B	BE18	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
N02BE/B	BE21	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
N02BE/B	BE21	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
N02BE/B	BE22	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
N02BE/B	BE22	ID:3296ASB, AmerisourceBergen Corp	1 W 1ST Ave Ste 100 Conshohocken, PA, 19428-1800 United
N02BE/B	BE22	ID:4128MMH, MEDIPAL HOLDINGS	3-1-7, ISOBEDORI, CHUO-KU KOBE, HYOGO, 651-0084 Japar
N02BE/B	BE25	ID:2172SCL, SUZUKEN CO., LTD.	8, HIGASHIKATAHAMACHI, HIGASHI-KU NAGOYA, AICHI, 461
N02BE/B	BE25	ID:4128MMH, MEDIPAL HOLDINGS	3-1-7, ISOBEDORI, CHUO-KU KOBE, HYOGO, 651-0084 Japar
N02BE/B	BE26	ID:4128MMH, MEDIPAL HOLDINGS	3-1-7, ISOBEDORI, CHUO-KU KOBE, HYOGO, 651-0084 Japar

Figure 13.18: The results will display only the columns that you chose to show in your query. (Used with permission from Microsoft)

Notice that this query combines several different techniques, including the use of multiple tables, sorting, criteria, and even an expression that combines fields in the query (in this case, the wholesaler ID and name). This is just a small demonstration of the power of queries.

Real-World Application - Using Data to Make Informed Decisions

Netflix collects and analyzes vast amounts of user data, including viewing history, ratings, and preferences, to drive their content strategy and personalized recommendations. By analyzing this data, Netflix can determine which TV shows and movies are popular among different segments of their user base, make data-driven decisions on content acquisition and production, and provide personalized recommendations that enhance the user experience. This data-driven approach has contributed to Netflix's success in delivering highly relevant and engaging content to their subscribers.

Netflix also uses data to make informed decisions in other aspects of its business, including the following:

- **Content acquisition and production:** Netflix analyzes viewership data, user preferences, and market trends to help it make informed decisions about acquiring rights to existing content or investing in original content production.
- **Audience segmentation and targeting:** Netflix leverages user data to segment its audience based on viewing habits, preferences, and demographics.
- **Pricing and packaging strategies:** Netflix analyzes data on user subscriptions, engagement levels, and pricing experiments to assess the impact of different pricing and packaging strategies.

- User experience and interface design: Netflix utilizes data on user interactions, browsing behavior, and feedback to continuously improve its user interface design and user experience. Netflix runs experiments, conducts A/B tests, and gathers user feedback to refine features, layout, and content presentation, ensuring an intuitive and engaging viewing experience.
- Customer service and support: Data analysis helps Netflix identify patterns and trends in customer support interactions, enabling it to improve response times, identify common issues, and enhance customer satisfaction. Netflix can also proactively address potential issues by analyzing data on streaming quality, buffering, and device compatibility.

Using this data-driven approach, Netflix can make informed decisions that drive customer acquisition, retention, and overall business growth, allowing it to stay ahead of the competition, continuously improve service, and deliver a highly personalized and satisfying entertainment experience to millions of subscribers worldwide.

Saving and Closing the Query

The first time you create a query, you need to give it a name. After that, if you modify the query, you will need to save it, and doing so will update the copy you have. To save the query, you can click on the Save icon on the quick access toolbar or use Ctrl+S on the keyboard. Access will also prompt you to save when you close a query that you modified.

If you do not plan on using a query again or if it is extremely straightforward, you may not want to save it, but if you are likely to reuse the query, you will want to give it a descriptive name. It makes sense to create a naming schema that you can continue. For example, if you are querying for a specific project team, you may start with the team name, like ProductTeam, to help you identify the query later on.

Building and Creating Queries

WorldCorp needs to quickly gather data related to sales records of their new health-care product in the market of Southeast Asia. Although you could design a query to gather relevant information from scratch, Access has a convenient wizard that can help you design a suitable query. Wizards are small programs that run within Access to help you set up the objects in your database. A wizard will prompt you to add the information you need. For example, the Query Wizard will ask questions that will help you decide what table (or tables) you want to work with, the fields you will use, and how you will display the output. Once you have created a query with the wizard, you can modify it however you like.

To access the Query Wizard in Access:

1. Click on the Create menu and locate the command for Query Wizard.
2. Choose the type of query you want to build. In this case, we choose the Simple Query Wizard, as you can see in Figure 13.19, which selects and displays specific records. This is the most commonly used query.

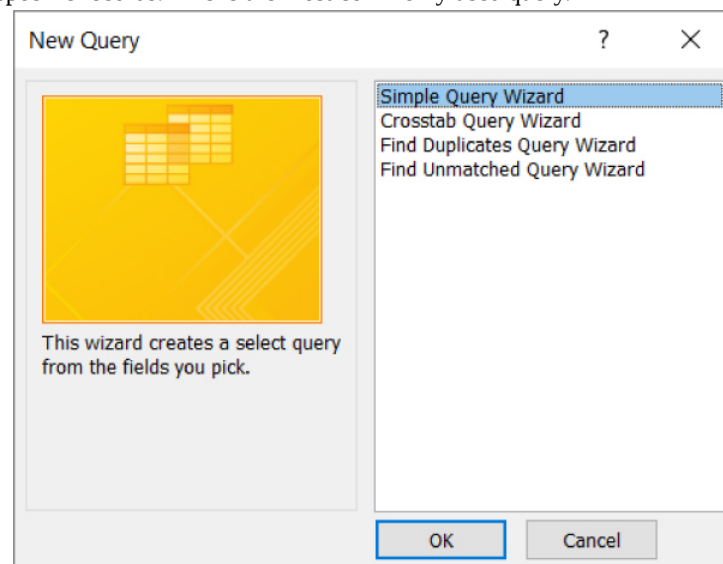
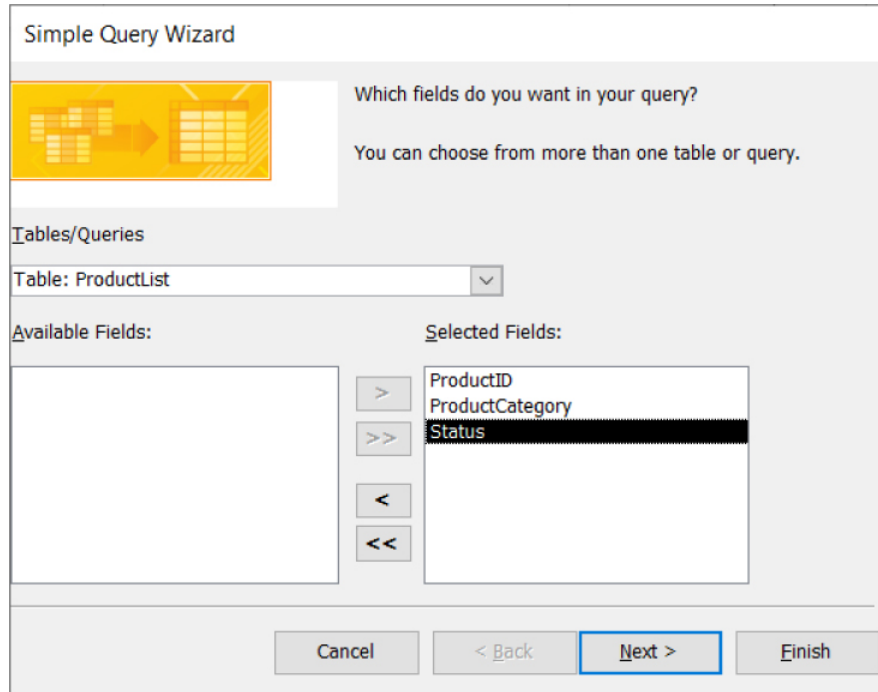


Figure 13.19: The Query Wizard can be used to create simple queries, crosstab queries, or queries to find duplicate or unmatched records. (Used with permission from Microsoft)

- Use the **>>** button to choose the table or queries and the fields you want to use in the query. You can add from different tables or queries by changing the Table/Queries drop-down menu to choose a different object to work with. When you are finished, click Next to move to the next screen (see Figure 13.20).



Simple Query Wizard

Which fields do you want in your query?

You can choose from more than one table or query.

Tables/Queries
Table: ProductList

Available Fields:

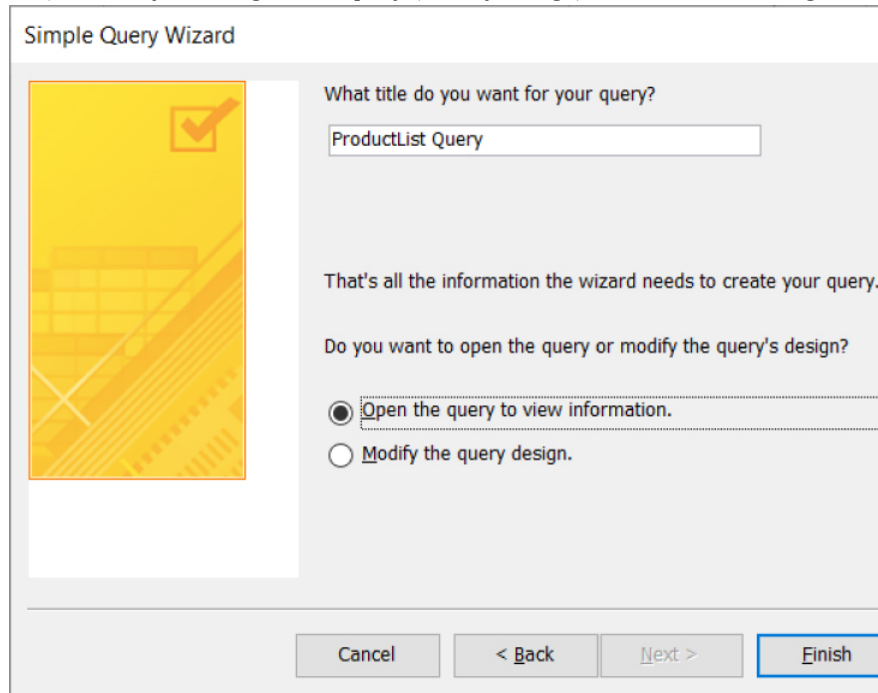
Selected Fields:

- ProductID
- ProductCategory
- Status

Buttons: Cancel, < Back, Next >, Finish

Figure 13.20: The Query Wizard allows you to select individual fields you want to include in your final query. (Used with permission from Microsoft)

- Once you have set up your query, you can name it on the last screen and choose whether you would like to see the results of the query (View Information) or modify the design of the query (Modify Design). Then click Finish Figure 13.21 to build the query.



Simple Query Wizard

What title do you want for your query?

ProductList Query

That's all the information the wizard needs to create your query.

Do you want to open the query or modify the query's design?

☒ Open the query to view information.

☐ Modify the query design.

Buttons: Cancel, < Back, Next >, Finish

Figure 13.21: The Query Wizard allows you to title your query before finishing it. (Used with permission from Microsoft)

Editing Queries with the QBE Grid

Once a query is built, you can edit it using the QBE grid, which gives you the ability to look at a query and rearrange, modify, and add to it, as well as the ability to run the query and see the results.

To open a query in the QBE grid, right-click the query in the navigation area of the screen, or to open the query, click on the View command, and choose Design View, as Figure 13.22 shows.

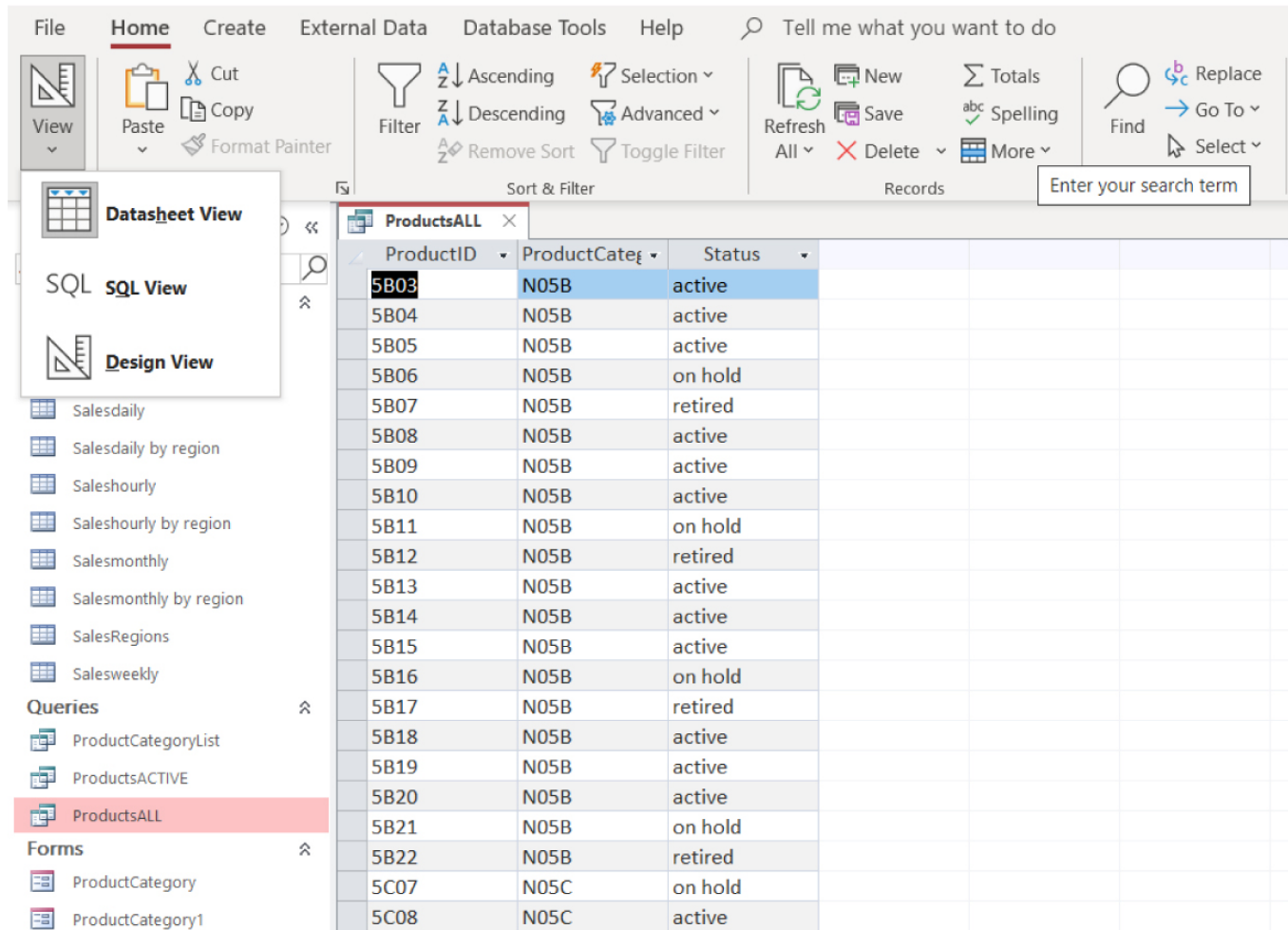


Figure 13.22: The View command gives you a few different options for viewing a query. (Used with permission from Microsoft)

The QBE grid (Figure 13.23) allows you to interact with a query visually. It places the elements of the query on screen in order from left to right and gives you the ability to modify the format of the query. You can do the following using the QBE grid:

- View, add, or change the tables used in the query
- Add or reorder fields
- Sort values returned by the field
- Choose whether a field is shown in the query (e.g., you may want to sort or use a field but not have it appear in the results)
- Add criteria to limit the records returned by the query

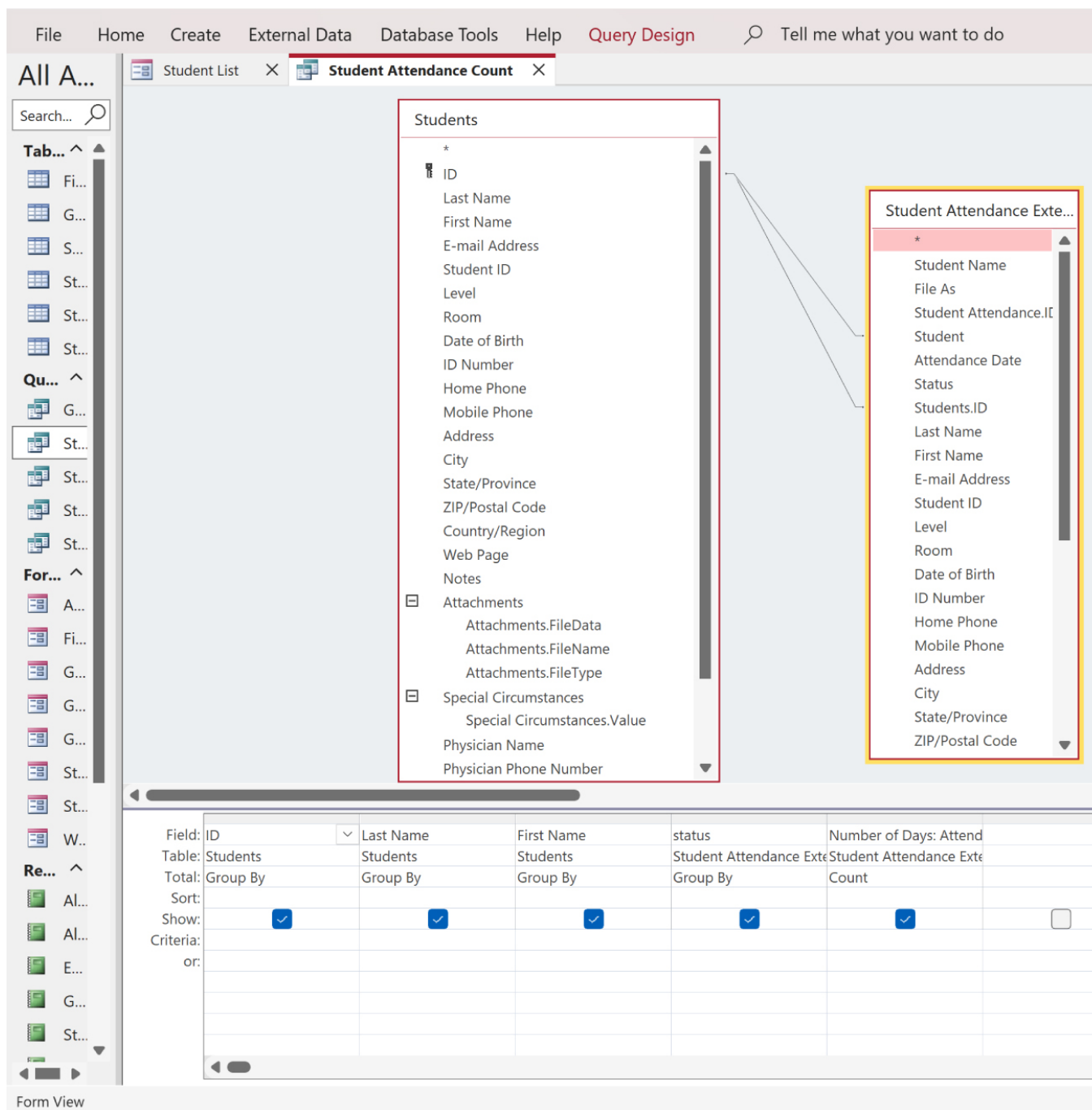


Figure 13.23: Use the QBE grid to interact with or modify a query. (Used with permission from Microsoft)

Building a Query with the QBE Grid

Although the Query Wizard is a useful tool and a good way to quickly start a query, using the QBE grid gives you more options for building complex queries. The QBE grid consists of a table that makes commands available.

In setting up the query, follow these steps:

1. Add tables.
2. Choose fields, one per column header.
3. Choose the sorting criteria.
4. Run the criteria to see results.
5. Return to Layout View to continue working.

To set a sort order, which will control the order of records displayed in the query, follow these steps:

1. View the query in Design View.
2. Locate the column you want to sort.
3. Click on the down arrow in the Sort row and choose the sort order (ascending, descending, or not sorted). By default, the column is not sorted.

You can also set criteria here to control which records are displayed. Criteria are commands that are written using symbols together with a text string that tells Access what you want the query to do. A criterion statement has two parts: the criterion/operator and descriptive text. For example, in the criterion statement

= 100

the equals sign serves as the operator, and 100 is the descriptor.

There are several different criteria you can use within your queries. Some of the operations (criteria) are displayed in Table 13.4.

Table 13.4: Display Criteria Using criterion statements adds power to a query. They allow you to build complex queries that organize data in interesting and useful ways.

Criterion	Result
=	Returns values that are equal =99 will return all values that are 99
>	Returns values that are greater than >99 will return all values greater than 99
<	Returns values that are less than <99 will return all values less than 99
Is null	Returns records with empty cells (Note: “null” signifies empty, <i>not</i> 0)
“ ”	Return records that are an exact match to whatever is between the “ ”
Like	Returns records that contain the value between “ ” “Like gp” will return all values containing gp
Not	Returns values that do not match the criteria “Not 100” will return all values that are not 100

Databases can hold hundreds, thousands, and even hundreds of thousands of records. One way to limit the results to what is relevant to your project or to the decision you need to make is to use criteria to limit the data that a query returns. Limiting the results makes it easier for you to work with the data and helps ensure that your decisions are both accurate and effective.

Setting criteria can help you select specific items in an Access database by showing only items that match the criteria you have set in creating the query. Individual criteria can be used in expressions, or statements, that specify values to select or omit.

Using Tables in Queries

The query grid will open with a blank gray area, a blank table across the bottom of your screen, and a window in which you can add tables. To add a table to the screen to work with it, double-click the table name or click and drag it into the gray area (Figure 13.24).

You will likely want to work with more than one table at a time, as one of the reasons we use relational databases is to be able to split our data into multiple tables and then combine them as needed, giving us enhanced performance and flexibility. To work with multiple tables, drag them onto the screen or double-click them.

If you have a query open in the QBE grid, you can add tables to the screen by right-clicking anywhere in the table area and choosing the Show Table command.

Figure 13.24: Adding an additional table to the query grid view allows you to create queries that involve multiple tables, enabling you to retrieve and analyze data from multiple sources simultaneously. (Used with permission from Microsoft)

Joining Tables

Splitting data into tables in a relational database like Access enhances database performance, which is a key issue in working with very large datasets. It also increases flexibility by allowing us to chunk, or separate, our data into smaller sets, which are easier to work with. Splitting up the data also enables us to avoid creating many null (empty) fields. (Imagine using a single spreadsheet to manage all your information—it would not be useful, as it would have too many fields, and many of those fields would be blank.)

When using a relational database to separate data into smaller, ID-focused tables (e.g., a table including only salespersons, another for clients), you need to join these tables based on key fields. In this example, you would likely create a primary key, SalesRepID, on the sales rep table, and then include this same field as a foreign key on the client table. You can then create queries that will join these tables by making sure that Access understands the connection between the fields.

In fact, the ability to create and define relationships between tables is one of the most important functions of a relational database. If you consider a database a set of data tables with relationships between them, Access needs to be able to connect related fields between one table and another. It does so by means of the connection or relationship known as a join.

A primary key, or unique key, is a field that uniquely identifies a record. As with an ID number, no two records can have the same primary key. A foreign key field is a primary key field that appears in another table. We can join foreign keys to primary keys to create a relationship between the two fields.

Recall that a join refers to the relationship between two fields in different tables. Access will create a relationship or join when it sees a primary key field in one table and the same primary key used in another. This is because Access understands that these fields have a relationship.

Joins appear between two fields in two or more tables in the query design screen. They are shown as lines connecting the fields, which represent the relationship between the two tables as a one-to-one, one-to-many, or many-to-many join. Access will create a join automatically when it recognizes the same field in two tables. At times, however, Access will not recognize the relationship, or you may want to relate two fields for the purpose of a specific query. In these instances—when we realize that two fields are related but that Access does not automatically understand this—we can also create joins manually. For example, this happens when different field names apply to the same field: A person may realize that CustomerNumber and CustomerID are the same, but Access likely would not recognize this. In this case, it is our job to add the relationship, letting Access know that these fields should be joined when we perform a query, as Figure 13.25 and Figure 13.26 show.

Figure 13.25: Using joins in a query allows you to establish connections between fields in different database tables, enabling you to retrieve and display related data by specifying how the tables are linked based on shared values or keys. (Used with permission from Microsoft)

Figure 13.26: The Join Properties dialog box enables users to specify the type and criteria for joining tables in a database query, defining how records from multiple tables are related and included in the query results. (Used with permission from Microsoft)

Types of Joins

There are two main types of joins: inner and outer. The type of join controls what data will display when the query runs. In an inner join, Access looks at the common fields and data between the two tables and only displays the matches, omitting any records from either table that do not match. The inner join is the most common type of join and is the default type for Access.

In an outer join, Access looks at the common fields and displays the related (matched) data plus the rest of the records from a single specified table. You can use the Join Properties box to choose which table will have all its records displayed.

As an example of an inner join, the sales and marketing database has a table with client information and a table with sales invoices. Both tables include the field ClientID. These tables are related using the default join type, the inner join.

Suppose the team at WorldCorp needs to see a list of all products with the clients that have purchased them so they can share new sales information appropriately. The tables in Figure 13.27 include the fields ClientInfo and SalesInvoices.

Figure 13.27: Access uses lines between tables to denote that a relationship exists between them. (Used with permission from Microsoft)

After looking at what the team actually needs for the process, you decide to add the fields from ClientInfo, including ClientID, ClientName, and ClientMailingAddressAll, and to use two SalesInvoices table fields—ClientID and ProductID—to narrow the results to just the clients who have purchased specific products.

After creating a new query and adding both tables to the grid, you realize that Access did not automatically create the join. You drag ClientID from the Client table to the matching ClientID in the SalesInvoice table. This creates an inner join, which will return only the details for the records that match.

You then add the fields for the query into the query grid. Add the ProductID field first because it should appear first on the screen. Then add the ClientID field, followed by ClientName and ClientMailingAddressAll.

Using these fields, as Figure 13.28 shows, you created a query that returns a list of each product, followed by the clients that purchased it. By adding ClientID, ClientName, and ClientMailingAddressAll, the team can send the correct mailings to the customers that have purchased these specific products in the past.

Figure 13.28: A query can use joins to return lists with data from multiple tables. (Used with permission from Microsoft)

Creating Joins

To create a join in Access, you need to have both tables open on the query design screen. With the tables open, click on the field you want to join in one table and drag it to the matching field on the other table. Access will usually create an inner join, displaying only the records that match on both tables.

If you want to display the data differently, you can double-click the join line and change the type of join. To change a join, double-click on the join itself in the query grid. The Join Properties box (Figure 13.29) will open. In the box, you can change the tables or columns of the join and can also change the type of join relationship. Sometimes it is helpful to modify the join on a dataset to see the results. The Join Properties box is helpful because it gives you a quick reminder of the results you can expect from the join, using the names of your tables.

Figure 13.29: You can change a join in the Join Properties box. (Used with permission from Microsoft)

If you decide you do not want or need a join for any reason, you can click on the join line in the query layout view and click Del on your keyboard. The join will be removed.

This page titled [1.1.8: Querying a Database](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.3: Querying a Database** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

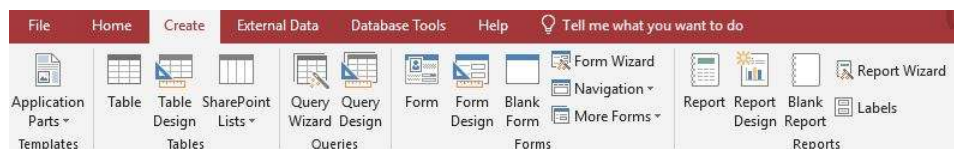
1.1.9: All About Queries

[Click here to view Microsoft Access - Setting a Foundation Chapter Sections](#)

Query Basics

Queries are far more powerful than the simple searches or filters that might be used to find data within a table. This is because queries can draw their information from multiple tables. The real power of a relational database, like Access, lies in its ability to quickly and easily retrieve and analyze data by running a **query**. Access queries allow users to pull information from one or more tables based on a set of search conditions the user defines. A common synonym for the word query is: question or inquiry. Queries essentially ask questions of databases. Queries do not store data. Therefore, a major benefit of queries is the ability to save a query to be re-ran at a later time after table data has been updated. This eliminates the need spend time recreating all of the fields and criteria desired to answer a popular question, such as a customer calling frequently to check on their latest account balance.

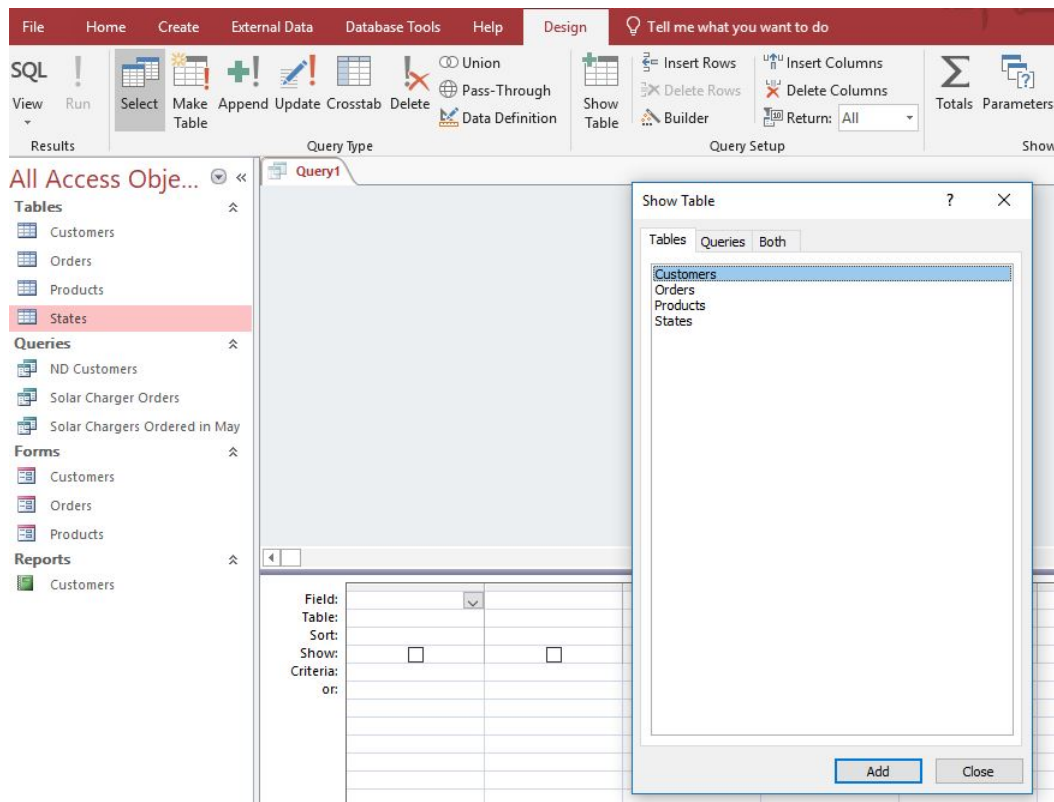
Queries can be something simple, like list all names in a table, or something a little more complex, like list all products that are priced over certain amount. A query can be very simple or very complex, depending on the user's requirements. As tables grow in size they can have hundreds of thousands of records, which makes it impossible for the user to pick out specific records from that table without a query. The tricky part about queries is that one must understand how to construct one before one can actually use them.



Access Create tab basic

When a query is ran, the results are presented in a table, but when a query is designed a different view is used. This is called **Query Design View**, and it lets users see how the query is put together. While the Simply Query Design Wizard can be used to display fields from one or more tables or queries with the option to choose a detailed or summary report, the wizard is limited by its inability to select data criteria. This is why most Access queries are created in Query Design View, as this allows for far more complexity to be added to create complex simple to complex queries.

After clicking the Query Design button on the Create tab, the Query Design View opens and by default the Show Table dialog window appears with a list of available tables and queries that can be added to build the query. Select one or more tables that contain the data desired to produce the answers to the potential questions. However, only add tables that contain fields that you intend to add to the query design. Tables added to the query design that do not have fields in the query grid can skew the query results. The exception to this rule is if the table is needed to join two or more other tables that are part of the query design. After the Show Table window is closed, it can be re-opened by clicking the Show Table button in the Design contextual tab.



Query Design

The next step to creating a query is to add the fields from the tables/queries into the query design grid (at the bottom of the window). Fields can be added by double-clicking the fields from the tables in the Object Relationship pane. A user can also drag a field from the table to the grid. A field can also be selected from the Field drop-down list in each column of the query design grid. If the fields are added in the wrong order, the column can be deleted or moved by clicking the top of the column and dragging the field to the appropriate location.

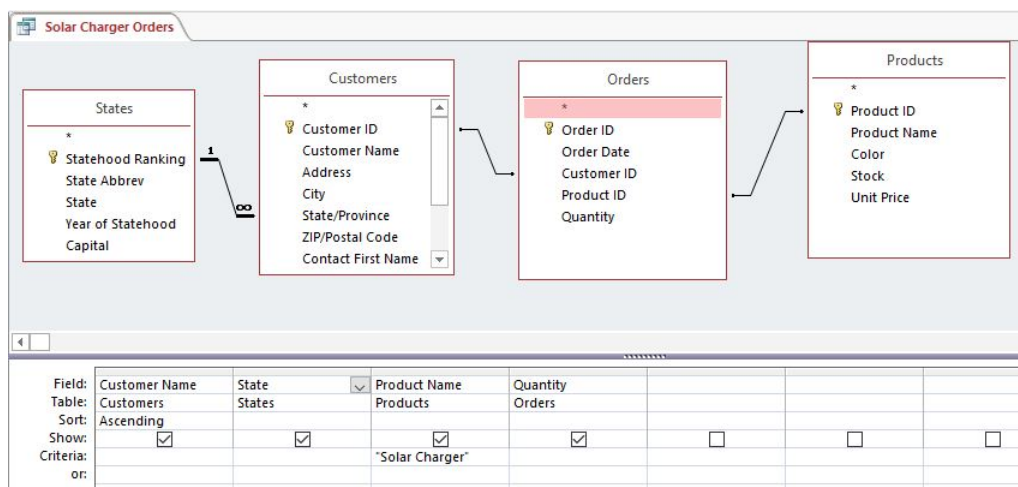
Types of Queries

Numerous queries from simple to complex can be created. A few of the more popular queries include:

- Single-Table Select query
- Multiple-Table Select query
- Range query
- Complex query
- Totals query
- Action query
- Parameter query
- Crosstab query

The simplest query is a **Single-Table select query**. A single-table query is based on only one table. It is essentially the same as an advanced filter applied to a table. To create a query in Design View follow these steps:

1. **Add the data source(s).** This can include table(s) and/or queries from the Show Table dialog window. For a single-table query, only one table will be selected.
2. **Join related data sources:** Access does not automatically create joins between queries and tables that are not related. If Access does not create joins when data sources are added, joins should be created manually. Data sources that are not joined to any other data source can cause problems with the query results. To add a join, drag a field from one data source to a corresponding field on another data source. Access displays a line between the two fields to show that a join has been created. **Note:** the graphic below is NOT an example of a Single-Table query. A Single-Table query would only have one table, so a join would not be applicable.



Query Design tables

3. **Add output fields** – Double-click or add a field from a data source in the upper pane of the query design window down to the **Field** row of the design grid, in the bottom pane of the query design window. To quickly add all fields down to the Field row of the query design grid, double-click the table or query name from the upper pane to highlight all the fields in that source and then drag them all down to the design grid at the same time. Alternatively, double-click the asterisk in the table.
4. **Specify criteria** – (Optional) Use the Criteria: and/or Or: rows in the grid to limit the records that the query returns based on whether data from the fields meet the specific criteria. Non-Select queries utilize this row extensively.
5. **Execute (Run) the query** – To see the results of the query, click the Run button on the Query Tools Design tab. Access will display the query results in Datasheet view. To make additional changes to the query, click the View button to return the Query Design View to make necessary edits before re-running the query. If the query is complete, click the **Save** command in the Quick Access toolbar. When prompted to name it, type a descriptive name and click OK. The new query should now appear in the Navigation Pane in the Queries object group.

As previously stated, a relational database must have multiple tables. Therefore, **Multiple-Table Select queries** are far more common than Single-Table queries. The key to these queries is adding tables or queries that are related through joins via a common field. This may require a little planning before composing at the computer. Determine what information is needed for output, and then pinpoint which tables contain the fields needed to produce the information. Further analysis may reveal that criteria might be needed to streamline the results. The query in the previous graphic uses fields from four related tables.

A select query with criteria that matches a range of values is called a **range query**. This type of query relies on relational operators to restrict the query results. When a range query compares text, alphabetical order is used to determine which data matches the criteria. For example, to return all used vehicles less than \$10,000 from a database of used vehicles, the criteria would be < 10000. Note the lack of a dollar symbol or comma delimiter? Adding a \$ in front a number would change the datatype from a Number to a Text data type. Another example using the States table that tries to return results that do not include North Dakota results would use a criteria such as <> "ND". Using the quotation marks indicates that the criteria is of the Text data type.

A complex query is any query that uses criteria in multiple fields. Criteria can include relational operators as well as logical operators. If criteria exists in the same row, the And operator is implied. The Or operator can be used in the Criteria row or using the Or: row below the Criteria: row.

Relational Operators	
<	less than
>	greater than
<=	less than or equal
>=	greater than or equal
<>	not equal to
Logical Operators	
And	requires an entry to match both criteria
Or	requires an entry to match one criteria
Not	requires an entry not match the criteria

Operators

Complex queries often utilize wildcard characters to add additional complexity. Complex queries sometimes include the same field twice to accommodate additional criteria. In these situations, the Show checkbox is often cleared to prevent the field from display twice in the query results. The graphic above illustrates a query that includes AND and OR criteria, as well as wildcard characters. The results should produce a list of music CDs that include any Classic Rock CDs and any Country CDs from Artists that include the name “Brooks” in any combination. This would include an Garth Brooks CDs as well as any Brooks and Dunn CDs.

Field:	Artist	Title	Genre
Table:	Music	Music	Music
Sort:			Descending
Show:	☒	☒	☒
Criteria:			"classic rock"
or:	Like "*Brooks*"		"country"

Complex Query criteria

There are times when setting simple criteria won't generate the desired results , especially when working with numerical values. The objective may be to see the query results grouped or counted in some way. A **Totals query** uses a type of function called an aggregate function to sum the data in a query result set. Aggregate functions perform a calculation on a column of data and return a single value. Access provides a variety of aggregate functions, including **Sum**, **Count**, **Avg** (for computing averages), **Min** and **Max**. To sum data, add the Sum function to the query, to count data use the Count function, and so on. The following example will illustrate how many of various bakery menu items have been ordered. The query datasheet results at right display the quantity of various cakes ordered on separate order transactions.

Product Type	Product Name	Quantity
Cakes	Black Forest	1
Cakes	Black Forest	1
Cakes	Black Forest	1
Cakes	Black Forest	5
Cakes	Black Walnut	1
Cakes	Black Walnut	1
Cakes	Black Walnut	3
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	2
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Buche de Noel (Christmas Cake)- Winter	1
Cakes	Carrot Cake	2

Total Query Before

Creating a Total query will group all identical menu items from separate orders, and then by using the Sum function, Access will add values in the Quantity field to calculate the total number ordered for each menu item. To create the Total query, the first step is



to click the Totals button from the Show/Hide group of the Query Tools Design tab. This will add a Total: row to the query design grid.

Field:	Product Types	Product Name	Quantity
Table:	Categories	Products Table	Order Items
Total:	Group By	Group By	Sum
Sort:	Ascending	Ascending	
Show:	☒	☒	☒
Criteria:	"Cakes" Or "Pies"		
or:			

Total row query design grid

The default values in the row are set to Group By. Clicking the drop-down arrow in the Quantity column of the Total: row, and selecting **Sum** will accumulate (add) the quantities of products ordered. Executing the query by clicking the Run button will display the results in the query's datasheet view as shown above. Note the Quantity field name changes to SumOfQuantity.

Product Type	Product Name	SumOfQuantity
Cakes	Black Forest	8
Cakes	Black Walnut	5
Cakes	Buche de Noel (Christmas Cake)- Winter	12
Cakes	Carrot Cake	9
Cakes	Cheesecake	18
Cakes	Coconut	7

Total Query After

In Access, queries can do a lot more than just displaying data. Through the use of assorted Action queries, Access can perform various actions on the database data. Microsoft Access provides 4 different types of Action Queries:

- 
Append – Retrieves data from one or more tables and adds the data to another table.
- 
Update – modifies or updates records. Criteria can be specified to limit which records are updated, but if no criteria is specified, all records will be updated.
- 
Delete – useful for quickly deleting a set of data on a regular basis.
- 
Make-table – retrieves data from one or more tables, and then loads the result set into a new table. That new table can reside in the database currently open, or a new database can be created.

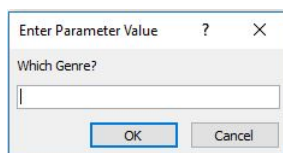
Action queries change data based on some set of criteria. Action queries can delete records, update data, append data from one or more tables to another table, and make a new table. Users must be cautious because once the query is executed, the data is changed and cannot be undone. Experts advise to make a backup of the database before proceeding with significant action queries.

A **parameter query** retrieves information in an interactive manner by prompting the end user to supply criteria before the query is run. If a particular query is used often but modified slightly, a parameter query may be more efficient. When the query is run, the user is first prompted to enter a parameter (or multiple parameters). Once the parameter has been supplied, the query can return the results, based on the parameter provided. The key to creating parameter criteria is to enter the parameter prompt inside square brackets, such as: [What's your favorite color?]. The square brackets will cause a parameter dialog box to open when the user runs the query. The text inside the square brackets will be displayed to the user, and an input field will be displayed for their input. If the user enters purple in the prompt, all records in table that store the data: purple will be returned as the query results.

Field:	Artist	Title	Genre
Table:	Music	Music	Music
Sort:			
Show:	☒	☒	☒
Criteria:			[Which Genre?]
or:			

Parameter query criteria

In the example to the right, parameter criteria is entered in the Genre field's Criteria cell.



Enter Parameter Value ? X

Which Genre?

OK Cancel

Parameter query prompt

When the query is executed, Access will display the following prompt: If the user were to enter the data: Pop in the prompt, the following results would be displayed the query datasheet view. (the data entered into the prompt is not case-sensitive)

Artist	Title	Genre
Prince	The Hits 1	Pop
Prince	The Hits 2	Pop
Curtis Stigers	(Self Titled)	Pop
Abba	Greatest Hits	Pop
Duran Duran	Decade	Pop
Barenaked Ladies	Disc One 1991-2001	Pop
INXS	Kick	Pop
Madonna	The Immaculate Collection	Pop
The Police	Every Breath You Take	Pop

Parameter query results

A **crosstab query** is a special type of query that calculates a sum, average, or other aggregate function, and then groups the results by two sets of values — one down the left side (row headings) of the datasheet and the other across the top (column headings)-much like an Excel spreadsheet. When building a crosstab query in Design view, use the **Total** and **Crosstab** rows in the design grid to specify which field's values will become column headings, which fields' values will become row headings, and which field's values to sum, average, count, or otherwise calculate.

Classic Rock CDs			
*			
Artist			
Title			
Genre			
Field:	Artist	Genre	Title
Table:	Classic Rock CDs	Classic Rock CDs	Classic Rock CDs
Total:	Group By	Group By	Count
Crosstab:	Row Heading	Column Heading	Value
Sort:			
Criteria:			
or:			

Cross Tab query

In the following example, the Music database will calculate a Count on the number of CDs are possessed by Artist – using the Row Headings and by Genre – using the Column Headings. The Title field will serve as Value field. The data source is an existing query named Classic Rock CDs, which provides the fields necessary to create the cross-tab query. After executing the query the datasheet results appear similar to:

Artist	Classic Rock	Country
AC/DC	2	
Aerosmith	3	
Billy Joel	2	
Blondie	1	
Bon Jovi	1	
Boston	1	
Brooks & Dunn		4
Bryan Adams	1	

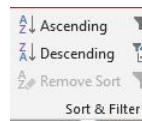
Figure 1.1.9.1: Copy and Paste Caption here. (Copyright; author via source)

A crosstab query does not always populate all of the fields in the result set because the tables used in the query do not always contain values for every possible data point. It is possible to enter an expression in the **Criteria** row to limit the results for the column heading field. However, using a criteria expression with the column heading field does not limit the number of columns returned by the crosstab query. Instead, it limits which columns contain data. For example, suppose you have a column heading field that has three possible values: red, yellow, and blue. If you apply the criterion **= 'blue'** to the column heading field, the crosstab still displays a column for red and a column for yellow, but only the column for blue contains data.

Sorting Queries

Sorting in Access is the process of rearranging records in a specific order. Sorting data can create a significant impact on the usability of Access queries and reports. Sorting can occur before or after executing a query. In the datasheet view, a query can

be sorted after running the query by using the Sort Ascending and Sort Descending buttons in the Sort & Filter group of the Home tab.



Sort buttons

To create more complex sorting criteria, use the Query Design window to create multi-field sorts.

Data Types	Sorting Options
Date/Time	Chronologically sorts in Ascending (oldest to newest) or Descending (newest to oldest) order.
Number, Currency or AutoNumber	Numerically sorts in Ascending (lowest to highest) or Descending (highest to lowest) order.
Short Text or Long Text	Alphabetically sorts in Ascending (A to Z) or Descending (Z to A) order.
Yes/No	Ascending (yes values, then no values) or Descending (no values, then yes values) order.
Memo, OLE or Attachment	Cannot sort by these data types.

Sorting Rules

Multi-field sorting requires the use of primary, secondary and potentially, tertiary sorting fields. Using Access' Query Design View, fields can be arranged in any order to accommodate the desired order. It is important to comply with the sorting rules inherent with the various data types of each field (see table to the left). Access will sort the fields in the Query Design from left to right. However, this can be manipulated by adding duplicate fields to the grid, specifically for sorting purposes. The key to this strategy is to utilize the Show checkbox for these sort-only fields. Clearing the Show checkbox will prevent these fields from displaying the Datasheet results, but their sorting mission will still be accomplished! Using Query Design also affords the ability to sort one field in Ascending order and another in Descending order in the same sort process.

In the example below, the goal is to sort the records primarily by Last Name. However, some records might share the same Last Name. Therefore, if there are duplicates, the First Name field will serve as the secondary sort criteria. Furthermore, it is desired that the results set display the records with the First Name field to the left of the Last Name field. This necessitates the use of a duplicate First Name field that will only be used for sorting purposes. Also notice that the Show checkbox is cleared for this duplicate field, so as not to confuse the layout of the results set. Hiding a field, by clearing the Show checkbox, can also be used for applying Criteria restrictions on a field that is not desired to be displayed in the results set, but is needed to narrow the results. This is popular with the sensitive data, like Salary or Credit Limits.

Field:	Contact First Name	Contact Last Name	Contact First Name	City	State/Province	Telephone Number
Table:	Customers	Customers	Customers	Customers	Customers	Customers
Sort:		Ascending	Descending			
Show:	☒	☒	☐	☒	☒	☒
Criteria:					"ND"	
or:						

Access Sorting

This page titled [1.1.9: All About Queries](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher](#) (Minnesota Libraries Publishing Project) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- [1.1.1: Introduction to Microsoft Excel](#) by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- [7.1: Query Basics](#) by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.
- [7.2: Types of Queries](#) by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

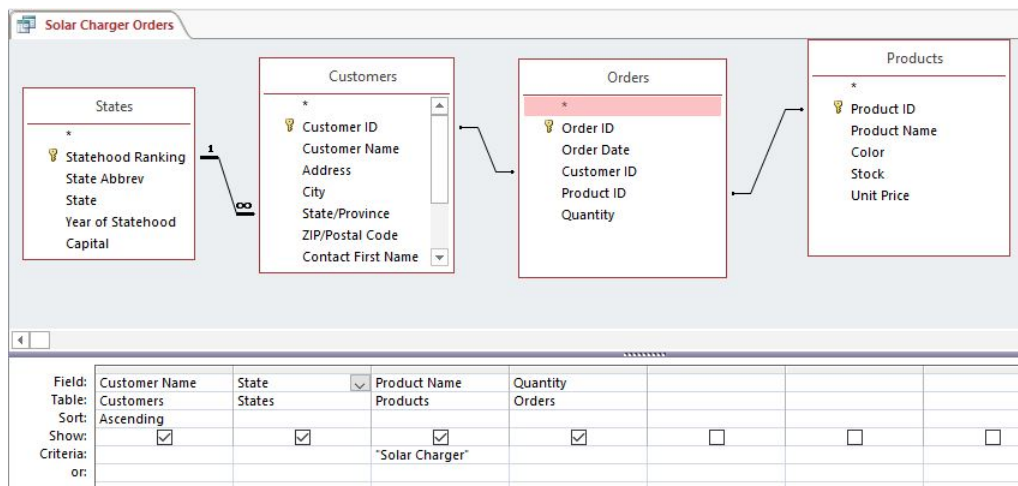
- 7.3: [Sorting Queries](#) by [Marcus Lacher](#) is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

CHAPTER OVERVIEW

2: Microsoft Access - Building Proficiency

Learning Objectives

- Microsoft Access and Data
- Advanced Queries in Microsoft Access
- Creating Forms in Microsoft Access
- Form Basics
- Customizing Forms
- Multiple Table Forms
- Creating Reports in Microsoft Access
- Customizing Reports
- Using Macros



2.1: Microsoft Access - Building Proficiency Sections

- [2.1.1: Microsoft Access and Data](#)
- [2.1.2: Advanced Queries in Microsoft Access](#)
- [2.1.3: Creating Forms in Microsoft Access](#)
- [2.1.4: Form Basics](#)
- [2.1.5: Customizing Forms](#)
- [2.1.6: Multiple Table Forms](#)
- [2.1.7: Creating Reports in Microsoft Access](#)
- [2.1.8: Customizing Reports](#)
- [2.1.9: Using Macros](#)
- [2.1.10: Microsoft Access Terminology](#)

This page titled [2: Microsoft Access - Building Proficiency](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [LueCrecy Ragan - Arkansas State University - Beebe](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

2.1: Microsoft Access - Building Proficiency Sections



Microsoft Access - Building Proficiency

This chapter explains how to use Microsoft Access for more advanced tasks in a simple way. It shows how businesses use Access to manage data, keep it secure, and export it to Excel. It also explains how to create queries to filter and analyze data. The chapter focuses on forms, which make it easier to enter and view data, including forms that connect multiple tables. It explains reports for clear data presentation and introduces macros and VBA to automate tasks and improve efficiency.

This page titled [2.1: Microsoft Access - Building Proficiency Sections](#) is shared under a [CC BY-NC-SA](#) license and was authored, remixed, and/or curated by [LueCrecy Ragan - Arkansas State University](#) - [Beebe](#).

2.1.1: Microsoft Access and Data

[Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Access Scenario

Now it is time to discuss advanced database applications at WorldCorp. As you continue to grow with the company, we will expect you to dive deeper into what is possible with data analysis. Your role will continue to advance as an analyst working with data professionals supporting various teams within WorldCorp.

WorldCorp wants to grow and expand into its independent retail locations to better serve its vast customer base. It still needs to create a database to manage inventory for these new locations. This would involve more complex database design and querying, as well as the use of advanced features in Microsoft Access.



Figure 14.1: As businesses grow, they often need to use more advanced features in Microsoft Access for complex database design and querying to help with tasks like managing inventory. (credit: modification of “Man in Black and Gray Plaid Dress Shirt Holding Black Smartphone” by Tiger Lily/Pexels, CC0)

First, we want to determine the data WorldCorp might need to store. We can start by identifying the key elements to track—potential product names, SKU numbers, locations, quantity on hand, and likely reorder points—and designing a table structure to best serve each type of data. For example, you could create a “Products” table with fields for product name, SKU number, description, and unit cost, as well as a “Locations” table with fields for location name and address.

WorldCorp would need a user interface to enter the data for each product and location into their respective tables. We will walk through some advanced steps in Access queries to retrieve specific sets of data from specific tables—for example, a query to find all products with a quantity on hand below the reorder point, or to find all products located in a specific store. You will also learn how to create reports to help keep track of inventory by summarizing and displaying data in a user-friendly format. For example, you could create a report that shows inventory levels by location, or one that shows sales trends by product category.

Data

Learning Objectives

By the end of this section, you will be able to:

- Integrate data from outside sources
- Export data from Access to other applications
- Build a form to serve as a dashboard that is updated automatically

Now that you are comfortable with building and maintaining databases, it is time to start talking about integration. Suppose you have one database that controls all accounting functions and another that simply focuses on operations. Integration allows you to share information between these two databases to provide comprehensive information about the company.

The simplest way to bring data from one Access database into another is by copying and pasting; however, this method does not support the updating of information, but for better control and flexibility, you may want to use importing and linking instead. When you import data from one database into another, Access creates a copy of the data in the destination database without altering the source. During the import operation, you can choose the objects you want to copy, control how tables and queries are imported, and specify whether relationships between tables should be imported. Relational databases have many layers and details baked into the cake. Making sure the details are captured is important.

You may want to import data to create a table or tables that are similar to tables that already exist in another database. You may want to copy the entire table or you may just need to import the table definitions so you can avoid the need to manually design each table. When you import only the table definition, the fields and field properties are copied to the destination database, but not the data in the table. The result is an empty table, which you can populate with your data.

If your goal is to add records from one database to an existing table in another database, you should consider importing the records to a new table and then creating an append query. You cannot append records to an existing table during an import operation.

Integrating Data from Outside Sources

Data stored outside the current database is called external data. External data may be data that you store in another Access database, or it may be data that you store in a multitude of other file formats. The first step in bringing outside data from Excel or other applications is to prepare the external data. An administrator may need to create an account and provide permissions to ensure that the right people have access to the data and that the data does not end up in the wrong hands. (For example, employees' confidential information must be kept absolutely confidential.) In an external database, the administrator may also want to create specific tables, views, queries, and so on to limit the results to only what is needed.

Access provides a powerful and effective means of presenting data, even data from external sources. When you access external data, you can either import the data into an Access database or you can link to the data from an Access database. Importing the data, though optimal, is not always possible. If you can't import external data, you should link to external files because Access maintains information about these linked files that will optimize performance when manipulating the external files.

Importing Data

The first step is to decide what you want to import and how it will join with your existing data. Normally, data is stored in various formats, files, and locations, which makes it hard to get and use it. If you have data in a spreadsheet, a SharePoint list, or some other format, you can import it into an Access database with just a few steps, making it much more easily available in Access. On the External Data tab, there are a number of tools for importing and linking to data in the Import & Link command group (Figure 14.71).

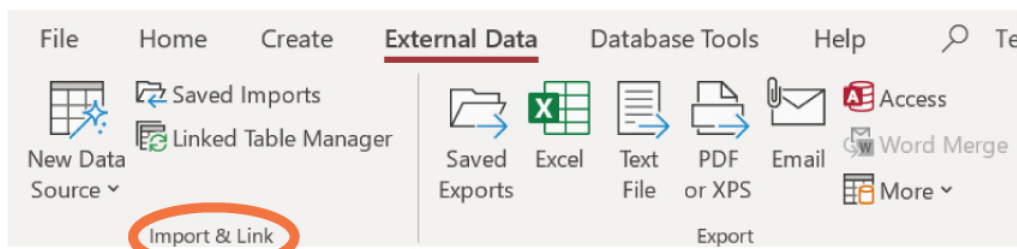


Figure 14.71: Data may often be stored in other programs but can generally be imported into or linked to an Access database. (Used with permission from Microsoft)

Linking to external data is quite different from importing data. Linked data remains in its native format. By establishing a link to the external data, you can build queries, forms, and reports that present the data. Once you have created a link to external data, the link remains permanently established unless you explicitly remove it. The linked table appears in the Navigation Pane just like any other Access table, but with a different icon. In fact, if the data source permits multiuser access, the users of an application can modify the data, just as users of applications written in the data source's native database format are able to do. The main difference between a linked table and a native table is that you cannot modify a linked table's structure from within Access.

There are two circumstances in which you should import (rather than linking to) external data:

1. If you are migrating an existing system into Access
2. If you want to use external data to run a large volume of queries and reports, and you do not intend to update the data.

In either case, you would choose the import option because it gives you the added performance that native Access data provides.

Steps for Importing

Now you are ready to take the steps to import data from another source. Often, data analytics seems to grow in Excel. More and more, we see far too much data being stored in Excel. Luckily, the following steps can help move the data into Access for you.

Open the database into which you want to import objects. You will only be able to import tables and queries from other Access databases; you can't import forms, reports, macros, or modules. The location of the Import Wizard will vary slightly depending on your version of Access. On the External Data tab, in the Import & Link command group, click New Data Source (Figure 14.72). Follow the wizard directions to add the imported records.

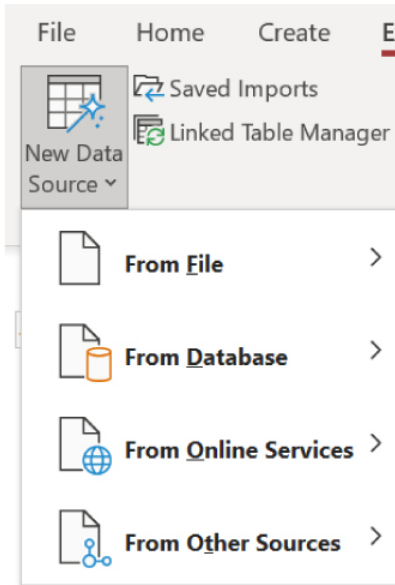


Figure 14.72: Access can accommodate importing data from a variety of different sources and file types. (Used with permission from Microsoft)

As an example, suppose that once a month your local distributor sends you an updated list of wines to sell at your restaurant. The distributor emails you an Excel file that includes both the updated list of names of wines and their updated prices. The wines listed will not change often, but the prices will reflect supply-and-demand changes in the industry. Because the list is extensive, it will be helpful for you to know how to import the list into Access and then how to make changes as needed.

You will need a table that contains both its own product ID and a corresponding Stock Number—the number the distributor uses to identify each type of bottle. Figure 14.73 shows an example of the Excel sheet a wine distributor might send.

	A	B	C	D	E
1	Stock Number	Region	Wine	Vintage	Price
2	5287341	bdxr	Carruades de Lafite, Pauillac	2007	\$1,173
3	4701496	bdxr	Carruades de Lafite, Pauillac	2018	\$1,572
4	9177244	bdxr	Chateau Angelus Premier Grand Cru Classe B, Saint-Emilion Grand Cru	2011	\$1,692
5	2273553	bdxr	Chateau Cheval Blanc Premier Grand Cru Classe A, Saint-Emilion Grand Cru	2018	\$1,130
6	8814798	bdxr	Chateau Figeac Premier Grand Cru Classe B, Saint-Emilion Grand Cru	2016	\$399
7	3712988	bdxr	Chateau Haut-Brion Premier Cru Classe, Pessac-Leognan	2000	\$1,987
8	2851294	bdxr	Chateau L'Eglise-Clinet, Pomerol	2010	\$1,715
9	6230705	bdxr	Chateau L'Evangile, Pomerol	2009	\$22
10	929357	bgnr	Domaine de la Vougeraie, Musigny Grand Cru	2019	\$1,019
11	4014430	bgnr	Domaine Faiveley, Musigny Grand Cru	2013	\$2,165
12	4048255	bgnw	Domaine Fontaine-Gagnard, Criots-Batard-Montrachet Grand Cru	2019	\$1,315
13	9303424	Rhone	Domaine Isabel Ferrando, Chateauneuf-du-Pape, Colombis	2007	\$449
14	3600581	Rhone	E. Guigal, Cote Rotie, La Turque	2016	\$1,279
15	1055140	ger	Egon Muller, Scharzhofberger Riesling Spatlese, Mosel	2020	\$1,424
16	9231681	bgnw	Jean-Claude Ramonet, Chassagne-Montrachet Premier Cru, Morgeot Blanc	2017	\$1,134
17	9088420	bgnr	Joseph Drouhin, Musigny Grand Cru	2018	\$502
18	9210632	bgnr	Laurent Ponsot, Chambertin-Clos de Beze Grand Cru, Cuvee du Frene	2018	\$1,601
19	4151373	bdxr	Liber Pater, Denarius, Graves	2018	\$773
20	3447526	bgnw	Maison Louis Jadot, Chevalier-Montrachet Grand Cru, Les Demoiselles	2019	\$2,040
21	1349380	bgnr	Serafin Pere et Fils, Gevrey-Chambertin Premier Cru, Fonteny	2015	\$537
22					
23					
24					
25					

Figure 14.73: Data is often contained in Excel spreadsheets that might contain formulas or links to other spreadsheets. (Used with permission from Microsoft)

In this example, start by adding a new data source from the saved file on your computer. The Import Wizard dialog box will appear, asking several questions to help you import the Excel data (Figure 14.74). Rather than just importing the Excel sheet, you can go ahead and link the sheet, knowing that the distributor will update the Excel sheet continuously.

Specify how and where you want to store the data in the current database.

We will not import table relationships, calculated columns, validation rules, default values, and columns of certain legacy data types such as OLE Object.

Search for "Import" in Microsoft Access Help for more information.

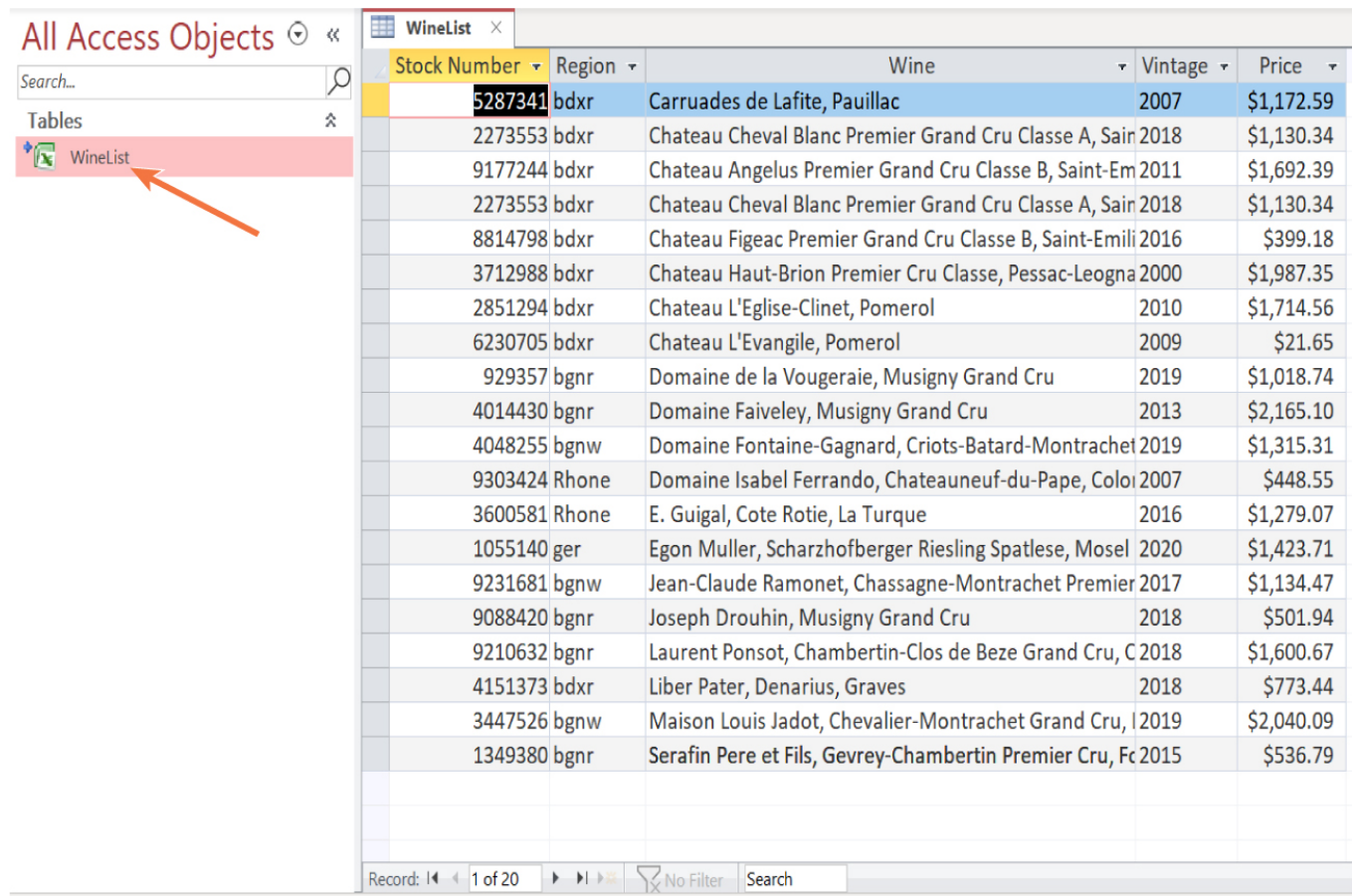
☐ **Import the source data into a new table in the current database.**
If the specified table does not exist, Access will create it. If the specified table already exists, Access might overwrite its contents with the imported data. Changes made to the source data will not be reflected in the database.

☒ **Link to the data source by creating a linked table.**
Access will create a table that will maintain a link to the source data in Excel. Changes made to the source data in Excel will be reflected in the linked table. However, the source data cannot be changed from within Access.

OK Cancel

Figure 14.74: You can import data into the database you have open as a new table currently or you can instead link it to an existing table. (Used with permission from Microsoft)

After the link is made, a permanent object is created in the All Access Objects list. The example shown in Figure 14.75 has the WineList table from Excel linked. It will automatically display any changes that have been made to the Excel file. The Excel table that is now visible in Access is easy to work with and functions much like a table.



Stock Number	Region	Wine	Vintage	Price
5287341	bdxr	Carruades de Lafite, Pauillac	2007	\$1,172.59
2273553	bdxr	Chateau Cheval Blanc Premier Grand Cru Classe A, Sain	2018	\$1,130.34
9177244	bdxr	Chateau Angelus Premier Grand Cru Classe B, Saint-Em	2011	\$1,692.39
2273553	bdxr	Chateau Cheval Blanc Premier Grand Cru Classe A, Sain	2018	\$1,130.34
8814798	bdxr	Chateau Figeac Premier Grand Cru Classe B, Saint-Emili	2016	\$399.18
3712988	bdxr	Chateau Haut-Brion Premier Cru Classe, Pessac-Leogna	2000	\$1,987.35
2851294	bdxr	Chateau L'Eglise-Clinet, Pomerol	2010	\$1,714.56
6230705	bdxr	Chateau L'Evangile, Pomerol	2009	\$21.65
929357	bgnr	Domaine de la Vougeraie, Musigny Grand Cru	2019	\$1,018.74
4014430	bgnr	Domaine Faiveley, Musigny Grand Cru	2013	\$2,165.10
4048255	bgnw	Domaine Fontaine-Gagnard, Criots-Batard-Montrachet	2019	\$1,315.31
9303424	Rhone	Domaine Isabel Ferrando, Chateauneuf-du-Pape, Colo	2007	\$448.55
3600581	Rhone	E. Guigal, Cote Rotie, La Turque	2016	\$1,279.07
1055140	ger	Egon Muller, Scharzhofberger Riesling Spatlese, Mosel	2020	\$1,423.71
9231681	bgnw	Jean-Claude Ramonet, Chassagne-Montrachet Premier	2017	\$1,134.47
9088420	bgnr	Joseph Drouhin, Musigny Grand Cru	2018	\$501.94
9210632	bgnr	Laurent Ponsot, Chambertin-Clos de Beze Grand Cru, C	2018	\$1,600.67
4151373	bdxr	Liber Pater, Denarius, Graves	2018	\$773.44
3447526	bgnw	Maison Louis Jadot, Chevalier-Montrachet Grand Cru, I	2019	\$2,040.09
1349380	bgnr	Serafin Pere et Fils, Gevrey-Chambertin Premier Cru, Fc	2015	\$536.79

Figure 14.75: The imported data file looks very similar in format to the Excel spreadsheet using the column titles. (Used with permission from Microsoft)

Link to Learning

One of the challenges of managing and analyzing data is understanding what to do when data contains errors. Read this [article discussing data cleaning and what to consider when cleaning data](#) to learn more. It will help you understand the importance of cleaning data as you plan your future data projects.

Exporting Data

To export data from Access, first select the table or other database object to export in the Navigation Pane. Click the External Data tab in the ribbon, and then click the button in the Export button group for the file format to which you want to export the object. This will open the Export Wizard for the type of export you want to perform.

The specific steps shown in each wizard will vary slightly, depending on the file type to which you are exporting. In most cases, you must select a name and file location for the exported object. You also often need to set additional parameters for the export routine. You will be prompted to save your export routine, if desired.

The Export Wizard enables you to export data from an Access database to a file format that can be read by Excel. When you export data to Excel, Access creates a copy of the selected data and then stores the copied data in a file that can be opened in Excel. If you copy data from Access to Excel frequently, you can save the details of the export operation for future use and can even schedule the export operation to run automatically at set intervals.

Typically, your department or work group will use both Access and Excel to work with data. You store the data in Access databases, but you use Excel to analyze the data and to distribute the results of your analysis. As another example, you may be a

longtime user of Access, but your manager or another colleague prefers to work with data in Excel.

Building a Dashboard with Navigation Forms

Dashboards provide a visual interface for users to interact with when using forms. The Dashboard provides a listing of buttons that direct the user to specific forms or reports. The first thing you see on opening many Access databases is a dashboard, a form populated with navigation forms that will be accessed through clicking on the specific command button. Using these provides the means for the user to navigate the database, opening forms, reports, and so on. The key for the user is a way to navigate. Traditionally, a dashboard is a graphical interface that allows a quick view of visualized data, as shown in Figure 14.76. In Access, however, we can build a dashboard from a form that will let us easily run queries and pull up reports. This section will demonstrate how to upgrade a Main Menu form that has only buttons and labels into a full dashboard of information for your company to use.

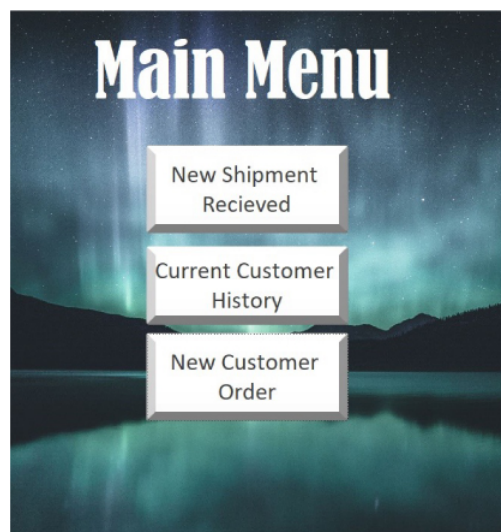


Figure 14.76: A dashboard is a menu of buttons that allows users to access other forms and subforms that you have created. (Used with permission from Microsoft)

Planning Your Dashboard

There are four general subtypes of dashboards:

1. Strategic dashboard: focused on long-term strategies and high-level metrics
2. Operational dashboard: shows shorter time frames and operational processes
3. Analytical dashboard: contains vast amounts of data created by analysts
4. Tactical dashboard: used by middle management to track performance

In the following example, imagine that you are a mid-level sales manager who wants to track a basic sales summary. A tactical dashboard seems like the best fit for your situation. Business intelligence has evolved into smart solutions that provide effective data management—from extracting, monitoring, analyzing, and deriving actionable insights needed to stay competitive in the market, to powerful visualizations created with a dashboard builder, which enable business users to interact with the specific bits and pieces of information they may need at any time. In this simple example, you want to record and track total orders and number of customers. Adding these elements to your Main Menu is a good place to start.

Building Your Dashboard

It is often helpful to first make a listing of the forms that will need to be regularly accessed by users. The forms could vary by department but there might be a group of forms that are routinely used company-wide. At WorldCorp, key performance indicators (KPIs) are critical to monitor for the success of their strategies. A dashboard that contains navigation options, as well as KPIs, can help manage the business. Figure 14.77 shows an example of this.

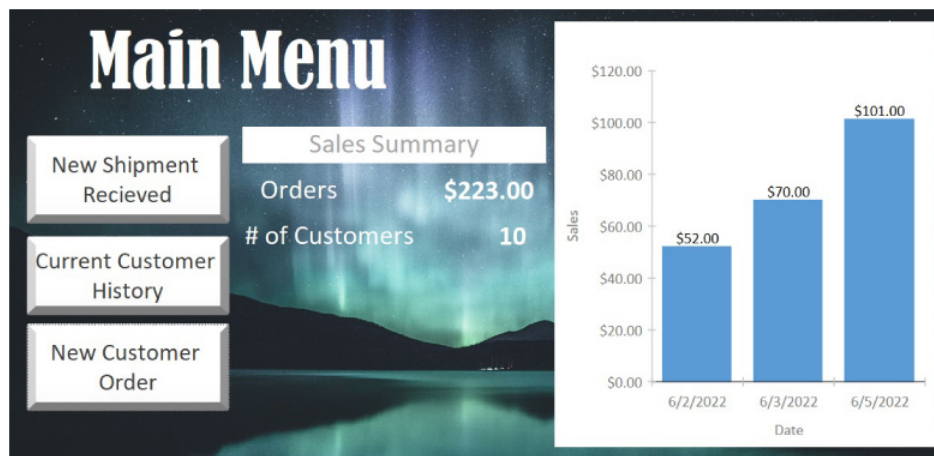


Figure 14.77: Dashboards can include not only navigation buttons but also summary information for tracking purposes. (Used with permission from Microsoft)

DSum and DCount expressions can add both a sales total and a customer count to your dashboard. These have been added to text boxes that were unbound to the form. Each text box was then formatted to fit the theme of the form. Under form design, Insert Modern Form was selected. This is a simple yet powerful way to add context to a sale. Additionally, your team can review the form and see the results as they are updated after each launch of the form. Note that there is an option to Refresh All on the Home tab, which will update the text box and charts.

Real-World Application - Relational Databases in the Cultural World

Many industries use relational databases. They are often hard at work in the background compiling data, gathering customer information, optimizing inventory, or organizing content for decision making. Database usage is more evident in industries such as banking, airlines and transportation, manufacturing, and retail.

In recent years, the usage of relational databases in unexpected industries has surged. One example is in libraries and historical preservation. Libraries and museums have utilized new technologies to make their collections of archives and artifacts available digitally for a much larger audience to enjoy, appreciate, and use for their own research. Wilmington College, a four-year liberal arts college in Ohio that was founded by the Religious Society of Friends (Quakers), has done just that. Through the generosity of a significant peace activist and donor to the institution, the college has extensive collection of artifacts related to Hiroshima and the impacts of nuclear weapons held at its Peace Resource Center (PRC).

The PRC archives is the most extensive collection of interdisciplinary materials related to the legacy of nuclear war in the United States. Each year, the PRC BRMA receives between 400-450 visitors, including researchers, students, and members of the public. The archives holds documents; creative responses to the atomic bombings, such as poetry, plays, and artistic works; historic poster collections; historic photos; slides; scrapbooks; photo albums; 16-mm films; reel-to-reel audio; cassette tapes; and artifacts among others.

—the Barbara Reynolds Memorial Archivese

Through the use of digital archiving and cataloging in an extensive database, this collection is available online to the broader public. The searchable, relational database is linked to several research libraries and contains digital images of the archives and artifacts held in the college's collection. Many other such databases of items of historical significance exist across the world such as through the Smithsonian and the United States National Archives. Historians and archivists have taken advantage of these technological tools to make history more accessible without compromising the integrity of the artifacts.

This page titled [2.1.1: Microsoft Access and Data](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.0: Chapter Scenario** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.6: Data Analysis and Integration** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.2: Advanced Queries in Microsoft Access

[Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Create a query with multiple criteria
- Create queries that use operators and expressions
- Design queries that select, aggregate, and run actions

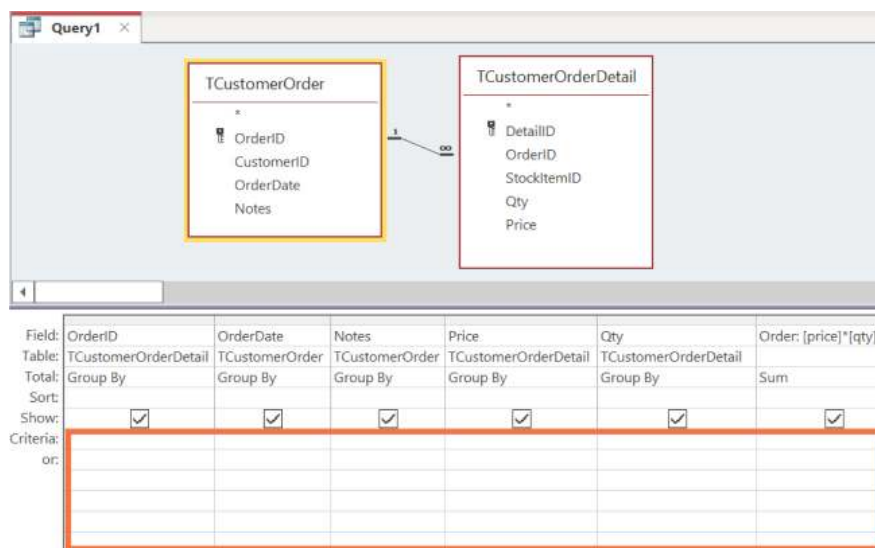
As your database continues to evolve, the incorporation of advanced queries aids in the management and manipulation of the exact data of interest. For example, at WorldCorp, you might need to collect all types of information on customers (e.g., email, phone, address, demographics, shopping history). This information often needs to be stored across multiple tables for both speed and efficiency. As an example, a new customer's basic information will be stored in one table. We assume this information doesn't change often—only when the customer moves to a different location or makes a major life change. In addition to contact and location, basic information could include demographic information, which may never change in our records.

In a separate table, we would store a record of each time the customer makes a transaction. Transactions may happen often and are usually unique, as each one has a particular time/date, exact location of purchase, and possible coupons or rewards points added. Additionally, a customer's purchase may be for a single item, several items, or perhaps 100 or more different items (think of a trip to the grocery store) on the same transaction.

By using an advanced query, you can isolate both customer information and purchasing information at the same time from both tables. For example, suppose you want to know how many customers shop for a particular service or item at locations at least twenty miles from their homes. Determining this would require enforcing multiple criteria across multiple tables and using advanced features in query design to find the exact data requested. The design tools available in Access allow for this level of advanced database design and function.

Queries with Multiple Criteria

A query criterion is a kind of formula used in Access to filter the records based on values in fields of interest to determine whether to include the record in your query results. Criteria may take the form of a number, full or partial text, or any combination of data points within a table, such as "Red" or ">5". You can include criteria for each field, as shown in Figure 14.2. Only items that match all the criteria you enter will appear in the query results.



Field	Table	Total	Sort	Show	Criteria
OrderID	TCustomerOrder	Group By		☒	
OrderDate	TCustomerOrder	Group By		☒	
Notes	TCustomerOrder	Group By		☒	
Price	TCustomerOrderDetail	Group By		☒	
Qty	TCustomerOrderDetail	Group By		☒	
Order: [price]*[qty]		Sum		☒	

Figure 14.2: Access data can be filtered and sorted using criterion such as customer or order date. (Used with permission from Microsoft)

Types of Criteria

Listed are the different types of criteria that can be used in an Access query:

Simple criteria: This type of criteria involves specifying an exact value or expression to filter the results. For example, when building out the inventory database for our potential freestanding retail locations, you could use simple criteria to specify which exact type of inventory you want to isolate.

Comparison operator: Comparison operators are used to compare two values or expressions. Examples include =, <>, <, >, <=, and >=. The < operator can be helpful to find out what inventory has less than a specified quantity on hand.

Wildcard: Wildcards are used to represent unknown characters in a search string. The two most common wildcards in Access are the asterisk (*) and the question mark (?). For example, a wildcard can be used to find all customers whose last name starts with a particular letter. Table 14.1 shows the most common wildcard symbols and a description of how they can be used when you have incomplete information or unknown characters.

Table 14.1: Wildcard Characters Wildcard characters are useful when filtering where data could lack uniformity or you have incomplete information.

Symbol	Description	Example
*	Identifies zero or more characters. Often used as the first or last character in the character string.	"ba*" finds ba, bat, ball, and balloon
?	Finds any single alphabetic character.	"w?ll" finds will and well
[]	Matches any character found within the brackets.	"T[ae]m" finds Tam and Tem, but not Tim
!	Identifies all characters not in the brackets.	"B[!ai]d" finds bed and bold, but not bad or bid
-	Ties any one of a range of characters. Users must specify the range in ascending order (A to Z, not Z to A).	"c[a-c]t" finds cat, cbt, and cct
#	Finds any single numeric character.	"5#4" finds 504, 514, and 524

Range criteria: Range criteria are used to specify a range of values for a field. As an example, a vender that we buy from to resell in our new retail locations may want to know all of the orders we have sold of theirs within a specific date range that have been taken out of our inventory. Rather than waiting for our next order, our vender could receive critical information on how much we may be looking to order in the near future.

Null criteria: Null criteria are used to find records that have a null value in a particular field. For example, you could use null criteria to find any inventory labels that do not include product descriptions.

Logical operator: Logical operators are used to combine multiple criteria in a query. The two most common logical operators in Access are AND and OR. For example, you could use the AND operator to find all orders with a total greater than \$100 and placed by a specific customer. Table 14.2 summarizes common logical operators.

Table 14.2: Logical Operators Logical operators are used often with databases to allow you to filter information with multiple criteria such as several locations, specific customers, or noncontiguous date ranges.

Logical Operator	Description	Example
AND	Returns TRUE if both operands are true, and FALSE otherwise. It is represented by the symbol &.	[quantity] > 100 AND [location] = Philadelphia
OR	Returns FALSE if at least one of the operands is true, and FALSE otherwise. It is represented by the symbol .	[quantity] > 100 OR [location] = Philadelphia

Logical Operator	Description	Example
EQV	Returns TRUE if both operands are either true or false, and FALSE otherwise. It is represented by the symbol <=>.	[quantity] <=> 25 OR [location] = Philadelphia
NOT	Returns TRUE if both operands are false, and FALSE otherwise. It is represented by the symbol ~ .	NOT ([quantity] > 10 OR [location] = Philadelphia)
XOR	Returns TRUE if exactly one of the operands is true, and FALSE otherwise. It is represented by the symbol ^.	(([quantity] > 10) ^ ([quantity] > 100))
IMP	Returns FALSE if the first operand is true and the second operand is false, and TRUE otherwise. It is represented by the symbol ->.	[location] = Philadelphia -> [quantity] > 100

By using these different types of criteria in an Access query, you can create powerful and flexible queries that can help you analyze and manipulate your data effectively.

Figure 14.3 displays a few sample criteria, shown in the highlighted box, and explains how they work with operators.

Field:	OrderID	OrderDate	Notes	Price	Qty	Order: [price]*[qty]
Table:	TCustomerOrderDetail	TCustomerOrder	TCustomerOrder	TCustomerOrderDetail	TCustomerOrderDetail	
Total:	Group By	Group By	Group By	Group By	Group By	Sum
Sort:						
Show:	☒	☒	☒	☒	☒	☒
Criteria:		Date()-1				
or:						

Figure 14.3: When delimiters are used to filter a database you will notice the criteria listed under the field column. (Used with permission from Microsoft)

At WorldCorp, you may need to isolate orders that took place during a certain time frame. For example, how did our marketing campaign do in sales over a particular time period for our retail locations? How many orders did we do over the past week? How many did we do over the past month? You might need this information to help schedule orders for our retail locations. This example returns items with yesterday's date (Date()-1). Thus, if the current date is 5/12/2024, you will see only items for the previous day, 5/11/2024.

In addition to the Date() function, there is also a Now() function to consider in Access. Both Date() and Now() are used to retrieve the current date and time, but they have different purposes and applications. The Date() function returns only the current date, without any time component. It is useful for calculating dates and time intervals, such as the number of days between two dates, or for filtering data based on a specific date. On the other hand, the Now() function returns both the current date and time, including the hours, minutes, and seconds. It is useful for tracking events that occur at a specific date and time, such as recording the time that a delivery showed up in inventory at a retail location.

Some reasons why you might choose to use the Now() function instead of the Date() function in Access can include the need to record time stamps. If you need to keep track of when events occur, such as the time of day when items are most likely to sell, the Now() function can be used to record the time stamp accurately. In addition, if you want to filter records based on a specific time range, such as all transactions that occurred within the last hour, then you will make use of the Now() function. In designing tables, note that it is important to use the correct format for each field used in the table you are pulling from in the query so that the unique expressions can be applied to that field.

In the example shown in Figure 14.4, the field labeled Order has a criterion of less than 50, so you will see only amounts that are less than \$50 from what our customers spent at WorldCorp. The amount found in this field is calculated by multiplying two fields (price and qty) and grouping the results by sum for a total of all unique Order ID numbers. Greater-than (>), less-than (<), equal-to (=), greater-than-equal-to (>=), less-than-equal-to (<=), and other mathematical expressions can be used to query all fields that are formatted as numbers, including currency, integers, autonumbers, and calculated fields.

Field:	OrderID	OrderDate	Notes	Price	Qty	Order: [price]*[qty]
Table:	TCustomerOrderDetail	TCustomerOrder	TCustomerOrder	TCustomerOrderDetail	TCustomerOrderDetail	
Total:	Group By	Group By	Group By	Group By	Group By	Sum
Sort:						
Show:	☒	☒	☒	☒	☒	☒
Criteria:						<50
or:						

Figure 14.4: A logical operator is used for order price to only display orders that are less than \$50. (Used with permission from Microsoft)

In the example shown in Figure 14.5, the Notes field in a query is based on a table that stores notes from sales orders that are formatted as long-text. A note can contain any amount of text, ranging from no words to a few words or even multiple paragraphs, depending on the order and the customer's requirements. This criterion isolates all notes that contain the string Express, signifying the customer's urgent need for the product. The key word "Express" can be extracted from all of the text logged in the Notes field, regardless of the order or length of the note.

Field:	OrderID	OrderDate	Notes	Price	Qty	Order: [price]*[qty]
Table:	TCustomerOrderDetail	TCustomerOrder	TCustomerOrder	TCustomerOrderDetail	TCustomerOrderDetail	
Total:	Group By	Group By	Group By	Group By	Group By	Sum
Sort:						
Show:	☒	☒	☒	☒	☒	☒
Criteria:			Like **Express**			
or:						

Figure 14.5: Notice the asterisk following the word "Express" as a wildcard to filter the data where the word in any form appears in the notes section. (Used with permission from Microsoft)

Depending on how you format the data, your next task is to select the appropriate expression to extract the data. Yes/No, OLE objects, hyperlinks, and attachments all have unique expressions to help optimize Access's ability to match your desired outcome. Again, formatting the data within tables is key to developing successful criteria expressions.

Building a Query with Criteria

When you create or modify a query in Access, you will be identifying/creating fields of data to isolate. These fields will represent the data you want to include, omit data that is not of interest to you, sort the data, and even combine data from multiple tables, as Figure 14.6 shows.

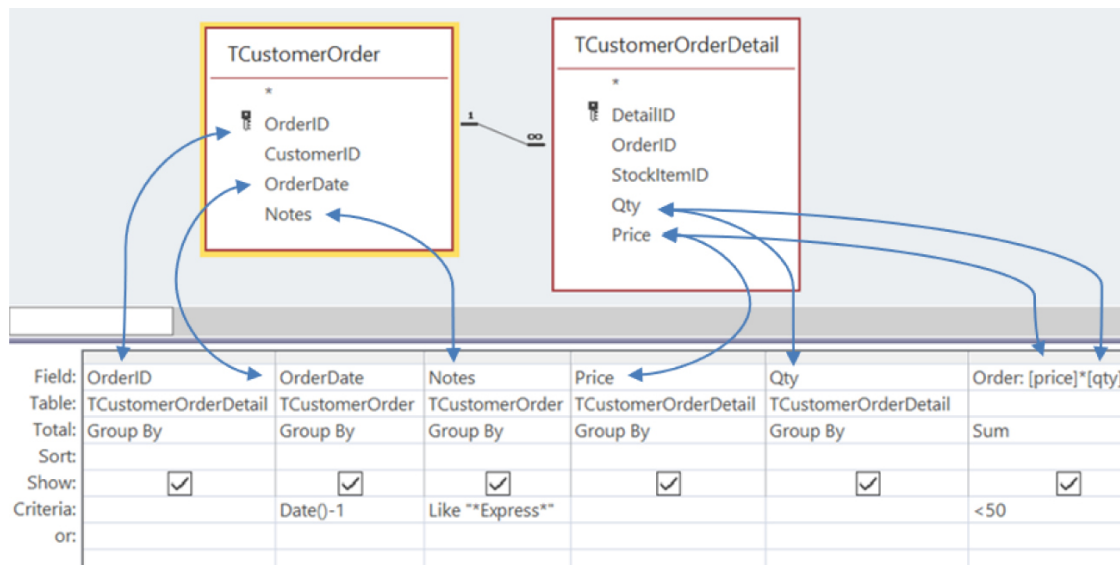


Figure 14.6: Several layers of filtering can be used in one query to build a customized list to meet specific needs. (Used with permission from Microsoft)

The example shown in Figure 14.6 of a query at WorldCorp combines two tables and isolates five fields from the related tables. The tables are related by the OrderID field. When a new order comes in, a unique OrderID is created in the TCustomerOrder table. Then, the order details are created in the TCustomerOrder table. The two tables are related by OrderID. This means that every order can have multiple sales items on the order. In short, every order is uniquely related to a customer. However, every order can have multiple sales items on the order. Those items are stored as a list in a separate table. Every customer, every order, and every list of order details need their own tables.

This example's five fields to review are: OrderID, Order Date, Notes, Price, and Qty. Because running this query would yield every order, it may not be useful unless you create criteria to isolate a particular question.

Within Query Design tab, choose the column of the field to which you want to apply a criterion. Click in the box below the field name in the Criteria row for that field. Type in your first expression or right-click in the Criteria box and choose "Build" to utilize the Criteria Wizard. Repeat these steps for each field for which you want criteria that help answer the question at hand. For instance, if you are concerned about offering free next-day shipping on all orders because of the rising costs of shipping, you might want to look at low-dollar purchases to see how often customers are asking for express shipping. One option is to review only those orders from yesterday (criterion #1), that were less than \$50 (criterion #2), and also had the string "Express" in the notes (criterion #3) on the receipt. The following criteria would be typed out or built through the wizard for each field highlighted, in order, as Figure 14.7 shows.

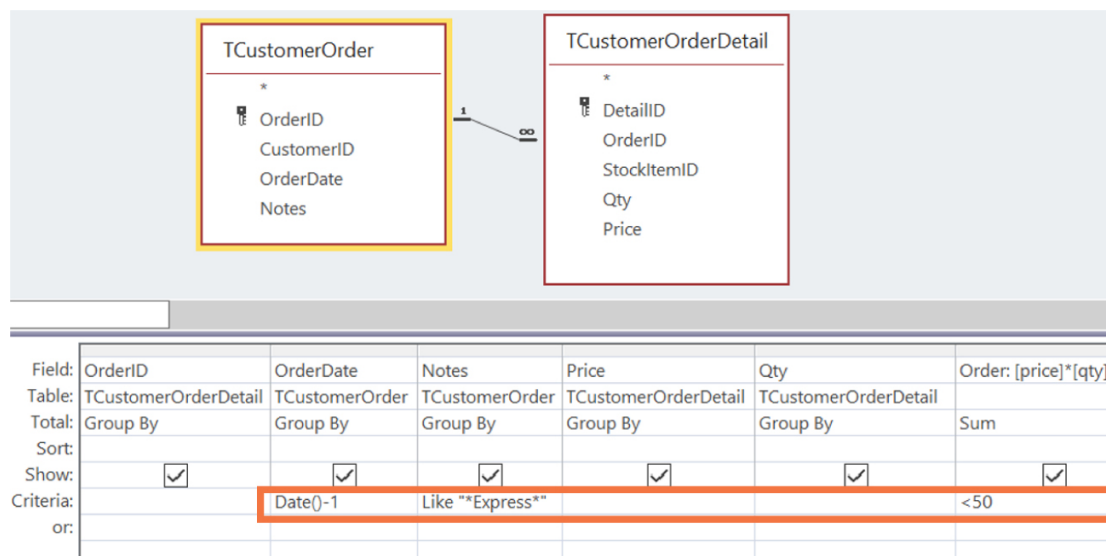


Figure 14.7: The Criteria Wizard can assist you in building a more complex query, or you can also type criteria directly into the query if you are familiar with the appropriate syntax needed. (Used with permission from Microsoft)

Link to Learning

Have you ever wondered how books made into movies do in theaters? And if it is a series of books such as *Harry Potter* or *The Hunger Games*, which book/movie is more popular? Databases can be used to find out these answers. The power of a relational database lies in its ability to efficiently manage and organize large amounts of data in a structured and flexible way. To find answers—for a person, an organization, or society as a whole—data needs to be processed in a way that helps transform it into useful information. To learn more, watch this [video discussing how to build a query with basic criteria using data points](#) using the *Harry Potter* movies for reference.

Using Operators and Expressions

As you develop an inventory system for WorldCorp’s future retail locations, inventory becomes tied with POS (point of sale) systems. The actual sales amount for each inventory item has not been calculated yet. Taxes, discounts, and price corrections all have to be considered for a total. Expressions become an important element to know and understand.

You might want to use expressions to create a formula to calculate a total order amount. The order amount is calculated by multiplying the price-per-item in stock to the quantity ordered of each item. As discussed in the previous example, or any number of other requests, if the number can be calculated, there may not be a need to store it in a table. In the previous example, the field “Order” has no associated table syntax is important to consider and be aware of when writing expressions in Access because it defines the rules and structure for how the expression is written. The set of rules by which the words and symbols in an expression are correctly combined is called syntax. Access uses a specific syntax to understand and interpret expressions, and any deviation from that syntax can result in errors or unexpected results. Proper syntax ensures that the expression is written in a way that Access can understand and process it correctly. It helps to avoid errors such as syntax errors, calculation errors, or incorrect results. Additionally, consistent use of syntax can make the expression more readable and understandable to other users or developers who may need to review or modify the expressions you create in the future.

Overall, understanding and using proper syntax is crucial for writing accurate and efficient expressions in Access (see Figure 14.8). Many of the expressions that are used in tables and forms are also applicable to writing/building expressions as criteria in queries. An expression is a combination of operators, identifiers, functions, and constants that are used to perform a calculation or to produce a result. An expression can be used in a variety of contexts, such as in queries, forms, reports, or macros.

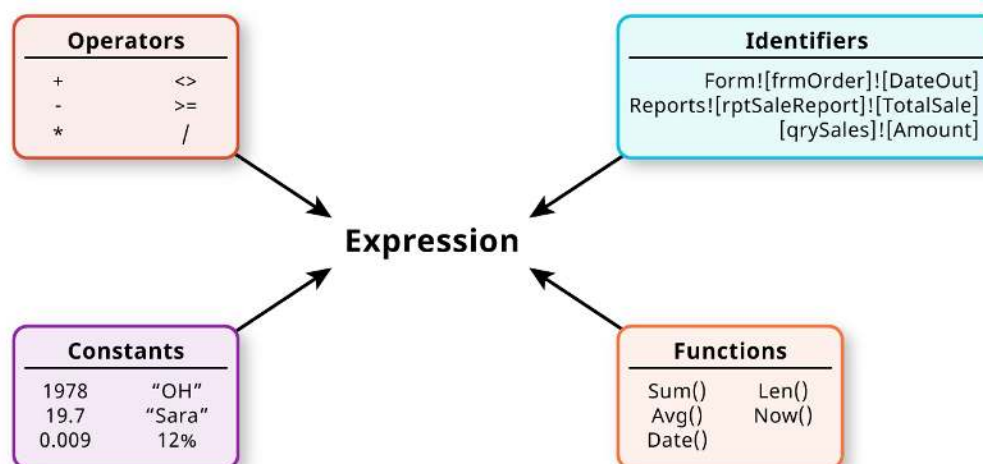


Figure 14.8: Through expressions, you can restrict queries based on text, numbers, and also perform functions. (Used with permission from Microsoft)

To be a valid expression in Access, an expression must contain the following elements:

- **Values:** Every expression must contain constants that include one or more values, such as a number, string, or date. Values may be constants, which are entered directly into the expression. Values may also be retrieved from fields or controls in a table or form.
- **Operators:** Operators are required symbols or keywords used to perform mathematical and/or logical operations on values.
- **Functions:** Functions are predefined calculations or operations that can be used in an expression, but are not required. Access includes a wide range of built-in functions to perform such tasks as calculating totals, formatting values, and manipulating text.
- **Parentheses:** Parentheses are used to group parts of an expression and control the order in which calculations are performed. Following the rules of mathematics, expressions within parentheses are evaluated first.
- **Identifiers:** Identifiers are names used to refer to tables, fields, and other objects in an Access database. Identifiers must follow specific naming conventions, such as not containing spaces or special characters, and must be enclosed in square brackets to work within expressions.
- **Commas:** Commas are used to separate multiple arguments in a function or expression.

In addition to containing these elements, keep in mind that correct syntax is established and maintained.

Understanding Operators and Expressions

The two most common logical operators used in building an expression for criteria in a query are AND and OR. These are assumed depending on where you place the expression in query design. Any expression placed in line with the Criteria line is defined as an AND operator. Our last example used all AND operators for the three expressions. See Figure 14.9 as an example.

Field:	Customer ID	Document	DateOut	Amount	Notes
Table:	tblCustomers	tblOutbound	tblOutbound	tblCustomerOrderDetail	tblOutbound
Sort:					
Show:	☒	☒	☒	☒	☒
Criteria:			Date()-1	<50	Like "Express"
or:					

Figure 14.9: By default, when criteria are added to more than one field, those criteria will be used simultaneously in the query using the "AND" operator. (Used with permission from Microsoft)

Placing all three criteria on the same line means that the query will display only those items that are from yesterday *and* are less than \$50 *and* contain the string Express. If you want the query to look for any of the three criteria, that is yesterday, *or* less than \$50, *or* containing the string Express each needs to be on a separate line. Date()-1 would be in the criteria row, <50 in the or row, and then place Like Express in the row below the or row. Rather than a dataset needing to match all three criteria or to match either of the two requirements, it can now match any of the criteria.

Any expression placed below the Criteria line is then defined as an OR logical operator. If we change our last example by rearranging the placement of the expressions, it operates very differently, as Figure 14.10 shows.

Field:	Customer ID	Document	DateOut	Amount	Notes
Table:	tblCustomers	tblOutbound	tblOutbound	tblCustomerOrderDetail	tblOutbound
Sort:					
Show:	☒	☒	☒	☒	☒
Criteria:			Date()-1		
or:				<50	Like "**Express**"

Figure 14.10: To query using the “OR” operator, the selected criteria should be moved one line below to the “or:” line. (Used with permission from Microsoft)

Placing the second and third expressions on the line below the first criterion means that the query will now yield those items that are from yesterday “or” are less than \$50 “and” contain the string “Express.” Rather than a dataset needing to match all three criteria, it now needs to match either of the two requirements.

Common Operators

In both Excel and Access, the most common operators are arithmetic, comparison, and logical. Each has its own function and returns different results.

- An arithmetic operator is used to calculate a value from two or more numbers or to change the sign of a number from positive to negative or vice versa (Table 14.3).

Table 14.3: Arithmetic Operators A mathematical operator could be used to combine data such as getting the total sales amount for two or more company purchases.

Operator	Description
-	Subtraction (2-1=1)
*	Multiplication (2*3=6)
/	Division (12/4=3)
\	Integer division (6\4=1)
+	Addition (15+4=19)

- A comparison operator is used to compare values and return a result that is true, false, or null (Table 14.4).

Table 14.4: Comparison Operators To compare one customer’s sales to another, you can use comparison operators such as greater than or less than.

Operator	Description
<	Less than
<=	Less than or equal to
<>	Not equal to
=	Equal to
>=	Greater than or equal to
>	Greater than
Between “Value1” And “Value2”	Between two values, inclusive (for example, “Between 1 And 3” would return 1, 2, 3)

- A logical operator is used to combine two Boolean values and return a true, false, or null result. There are three main logical operators that are commonly used in Access:

- AND operator: This is used to combine two or more expressions and returns TRUE only if all the expressions are true. For example, the expression ‘size > 8 AND Color = 'Blue'’ will return TRUE only if the size of a shirt is greater than 8 and the color is blue.
- OR operator: This is used to combine two or more expressions and returns TRUE if any one of the expressions is true. For example, the expression “Size > 8 OR Color = 'Red'” will return TRUE if either the size is greater than 8 or the color of the shirt in inventory is red.
- NOT operator: This is used to reverse the logical value of an expression. It returns TRUE if the expression is false and FALSE if the expression is true. For example, the expression “NOT (size > 8)” will return TRUE if the size of the shirt is less than or equal to 8.

These logical operators can be used in combination with comparison operators such as =, <, >, <=, >=, and <> to create complex expressions that evaluate to true or false (Table 14.5).

Table 14.5: Logical Operators Queries that filter using several layers might involve logical operators to extract information from a large database.

Operator	Description
AND	Both elements of an expression must be true.
NOT	The expression must evaluate as false.
OR	At least one element of an expression must be true.
XOR	Exactly one element of an expression must be true, not both.

The next example, shown in Figure 14.11, is a query designed to help create a report—in this case, you need to generate an invoice after a customer places a new order with WorldCorp. Invoices for customer sales include items such as the customer information, the date, the products ordered, price for each product, and total for the entire invoice. Refer to the chapter on Advanced Document Preparation for examples of invoices. For the report to work properly, two different criteria must be used with the logical operator AND. The first expression is [Forms]![frmOutBound2]![Document], which uses identifiers to isolate which document number the report will generate. In essence, this expression uses identifiers to query only the exact document number that is in use on the frmOutBound2 form, which is used to place all new orders from customers. By identifying the form number, the invoice will generate the exact order and only the called-upon order.

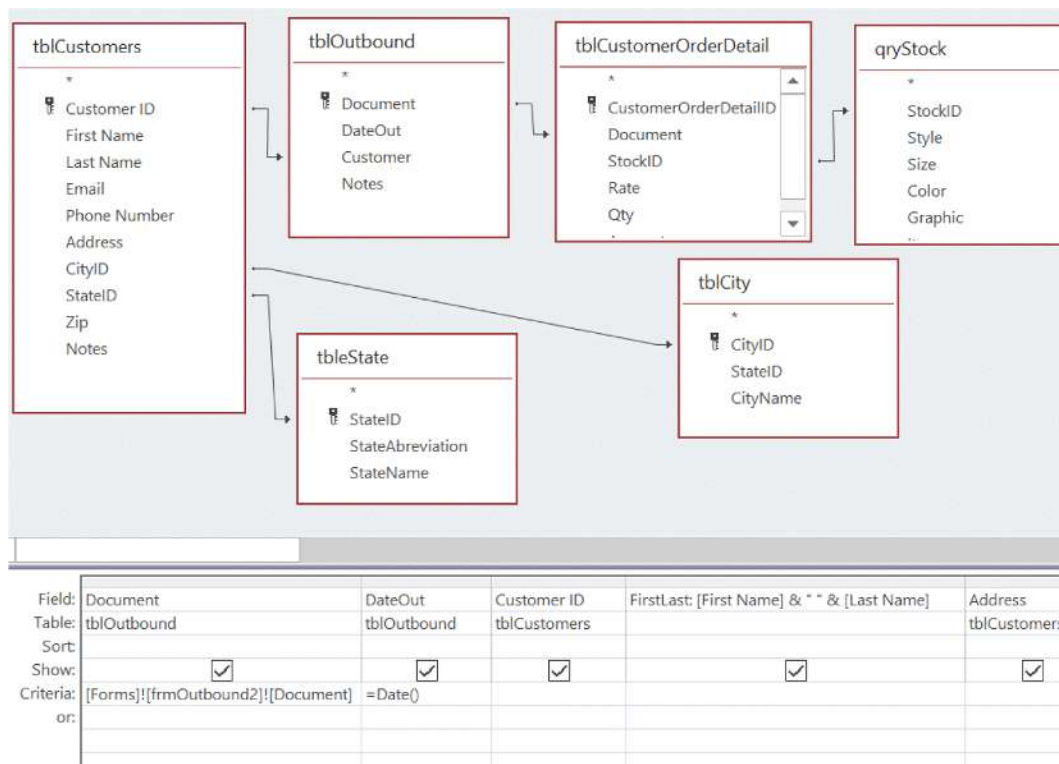


Figure 14.11: Access not only lists the fields in text, but also provides a visual representation of how elements are connected in the database. (Used with permission from Microsoft)

The second criterion is a function expression that isolates the order date to today, and today only, ensuring that the query only creates new invoices that represent new orders from today (see Figure 14.12).

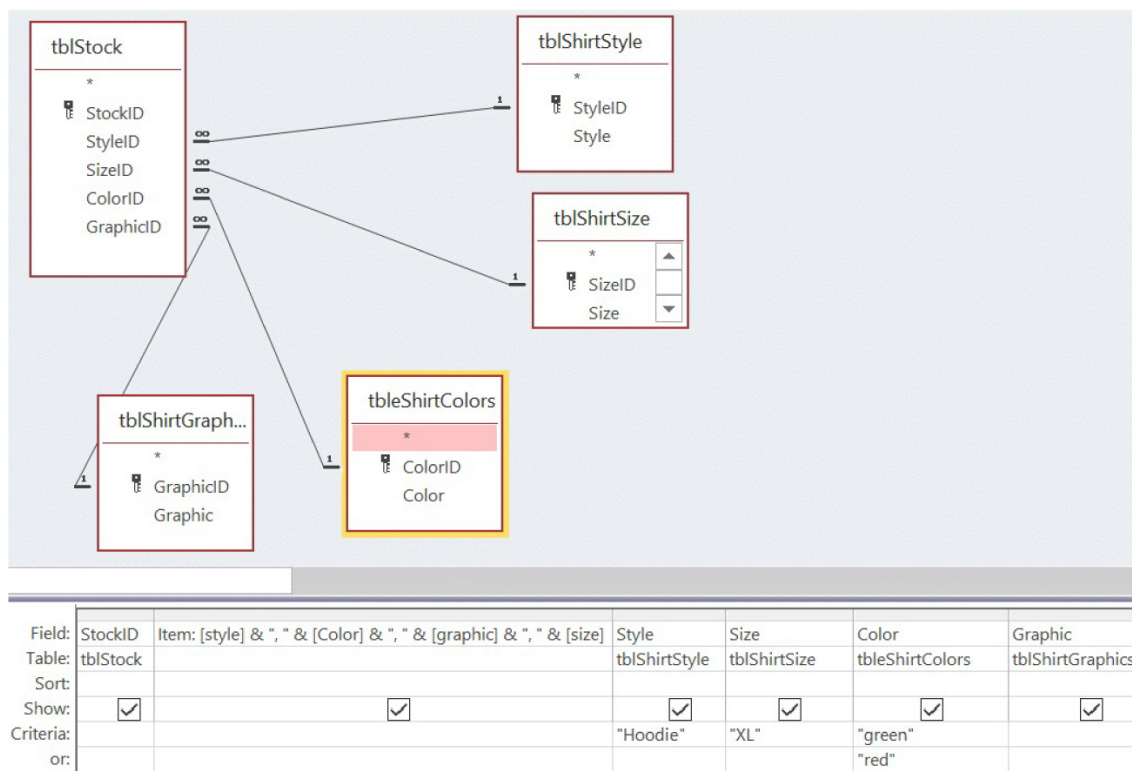


Figure 14.12: Notice how several fields are used to perform the query using logical operators. (Used with permission from Microsoft)

This example shows a query designed to help show which items a T-shirt design company has in stock. Each stock item has four fields that collectively describe the piece of clothing: style, size, primary color, and graphic design on the item. In this example, a customer wants to know what graphic designs are available on a hoodie that is size XL and has a base color of green or red. These criteria combine both constants and logical operators to help isolate the data requested by the customer.

Guidelines for Using Criteria

There are best practices to follow when using criteria:

- Keep it simple. Do not try to do too much with any one query to get an answer. A query can do a great deal, one strategy is to copy previously designed queries and rename before making small changes to each copied query for a very specific purpose.
- To stay organized, be consistent in how your queries are named. When you first learn the syntax used in creating criteria, it is tempting to try to use all of the syntax in a single query. When creating a query, however, it is more important to keep the main purpose of the query in mind. If you have several questions, don't be afraid to create multiple queries—a separate one for each question or task.
- Be sure to create tables carefully, using the correct format for each field within each table. Using good naming practices is important: A key reason that queries won't run is likely to be the formatting of the fields rather than incorrect syntax.
- Develop a list of expressions that you know and can use for future projects. Often, these can be repeated in various queries for various uses.

Designing Queries to Run Actions

Now that you know how to build a query, let's look at the different types of queries in Access. There are five types of queries, each with a unique function and use: (1) select queries, (2) action queries, (3) parameter queries, (4) crosstab queries, and (5) SQL (structured query language) queries.

Select query is the most common type of query and the easiest to use. They retrieve data from one or more tables and display the result in a datasheet. Select queries are also used to group records and calculate averages, sums, counts, and other types of totals. As an example, a select query might be used to group customers by state to see which states have the highest demand.

Action query specifies a particular action, such as creating a new table, deleting rows, updating records, or creating new records. Action queries are very popular in data management because they allow for many records to be changed at one time. For example, suppose you need to update your records to reflect a new state tax rate that affects numerous types of transactions. Creating an action query can help update records across various tables.

There are four types of action queries:

1. Append query: Adds records from one or more tables to the end of one or more tables.
2. Update query: Makes global changes to a group of records in one or more tables. For example, you can use an update query to increase the prices on your menu by 25 percent for products within certain categories.
3. Delete query: Deletes a group of records from one or more tables. For example, you can use a delete query to remove products that have been discontinued or for which there are no orders.
4. Make-table query: Creates a new table and populates it with data from one or more existing tables. When you create a make-table query, you specify the fields and criteria that you want to include in the new table. Access then creates a new table and copies the selected data from the existing table(s) into the new table. The new table will have the same field names and data types as the original table(s). These queries can be useful in a variety of situations, such as creating an archive or backup of data. You can use a make-table query to create a backup copy of important data before making changes to it. In addition, simplifying complex data can be very helpful. As an example, if you have a table with many fields or complex relationships, you can use a make-table query to create a simplified version of the data that is easier to work with on a different project. Also, aggregating data can be accomplished. You can use the query to aggregate data from multiple tables into a single table, which can be useful for reporting or analysis purposes.

Parameter query is a type of query that prompts the user for input values when it is run. The input values are used to filter the data that is returned by the query, based on the specified criteria. Parameter queries are useful when you want to create a query that can be easily customized to return different results based on specific criteria. For example, this query can be used in building a form that asks users which employee is logging into the Main Menu screen to start a transaction at WorldCorp; you type [Which User?] on the Criteria line. When the query runs, Access will prompt the WorldCorp employee for the answer to your question.

Crosstab query calculates and restructures data for easier analysis when looking for trends or patterns. Much like the PivotTable option in Excel, Crosstab queries calculate a sum, average, count, or other type of total for data that is grouped according to two different types of information. For example, total sales for WorldCorp might be grouped in two ways: by salesperson, listed down the left side of the datasheet, and also by month and year, listed across the top of the datasheet, which now shows total sales for each salesperson by month and year.

SQL query is created by using an SQL statement. When you create a query in Query Design View, Access builds the equivalent SQL statements for you behind the scenes. You can view or edit the SQL statement in SQL View. Upon viewing any of the created statements by Access, you can edit or add to the statement. This creates additional options for you that are often more efficient than using the Query Design View options.

Building a Query that Aggregates

Aggregate query lets you carry out calculations on record groups rather than performing individual operations. An aggregate query considers the total, subset, or gross amount of records.

Aggregation is a powerful tool because of the nature of relational data. By looking across the connected tables and through the built-in relationships, these can be accounted for when performing complex data searches and totals among a vast area of applications.

Often, the first step in building an aggregate query is to turn on the Totals tool that is found within the Query Design tab, as seen in Figure 14.13.



Figure 14.13: The Totals tool is found in the Show/Hide command group when you are designing a query. (Used with permission from Microsoft)

After you select Total from the menu, a new row, labeled Total, appears just below the field names, with an option area in the Access query, as seen in Figure 14.14.

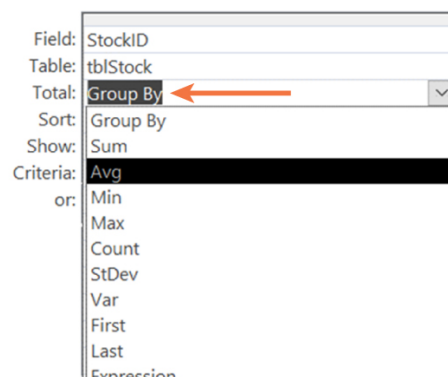


Figure 14.14: Although the word “Total” is used, you can choose to get values other than just the sum of the numbers such as the average or the maximum number in the field. (Used with permission from Microsoft)

The default option is Group By, which groups all identical data items in the field. For example, if you wanted to know which stock item in your company’s inventory sold the most units, you could use the StockID in two ways:

1. Create a field for StockID in the query, and list the Totals as Group By. This will provide a list of each type of item you have sold.
2. Create another field, also with the StockID, but this time, choose the Sum function of the Total to provide a complete sales count for each unique item. Both fields together provide a report on which items have sold the most and the least.

Examples of Aggregate Queries

Scenario #1: You have an online business that has customers from around the country, and you want to find out which states have the fewest customers. You decide to create an aggregate query to summarize the list of states and count the number of instances in

which each state appears among your current customers. Selecting the Totals option creates the following query from the customer table, as Figure 14.15 shows: Group By StateName and Count StateName.

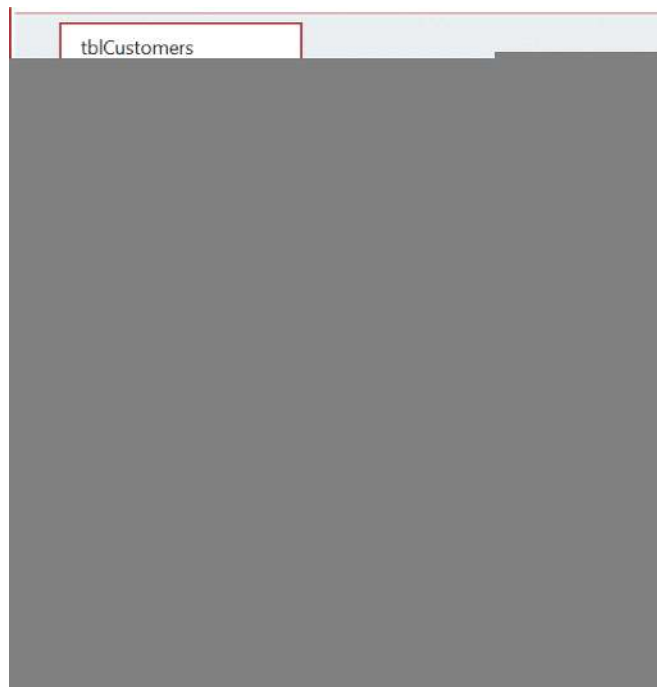


Figure 14.15: This query will give a summary of the total number of customers in each state using the “Count” operator. (Used with permission from Microsoft)

Scenario #2: You receive two free tickets to the Super Bowl. Unfortunately, you cannot attend this year’s game, so you want to give the tickets away as a thank-you gift to the customer with whom you have done business the longest. To determine the recipient, you decide to do an aggregate query on customer orders, using date ranges to identify the longest-standing customer. The following aggregate query (shown in Figure 14.16) is then created by grouping the customer ID, using DateOut as the minimum total (first order) and then using DateOut as a maximum total to establish the last sale the customer made (last order).

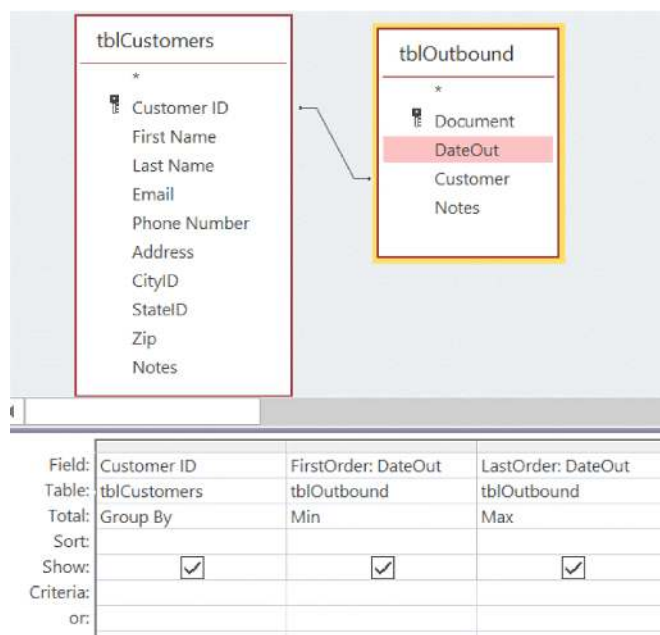


Figure 14.16: More complex queries can be constructed to compare dates or other fields using the Total tool. (Used with permission from Microsoft)

In this scenario, it would be best to build an additional query to calculate the length of time between FirstOrder and LastOrder, as determined in the original aggregate query. The query has to solve both totals before it can calculate the difference in time between

the two fields using the Min and Max functions.

This page titled [2.1.2: Advanced Queries in Microsoft Access](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.1: Advanced Queries in Microsoft Access** by [OpenStax](#) is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.3: Creating Forms in Microsoft Access

 [Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Understand why forms are useful
- Create and edit a form using the Form Wizard
- Customize a form by adding images and text

Relational databases are built on a structure of tables; each table is made up of columns, which hold fields, and rows, which hold the information for each field for a specific record. This is an effective structure for the organization, but as the database becomes more complex and we use joins in queries to create relationships between data across tables, we need an easier and more manageable way to work with the database. Using forms to interact with our database tables has two benefits:

- Forms provide an easy way to interact with the database.
- Forms protect the data and underlying table structure from accidental changes.

Most websites are based on databases and use forms to connect to the database. In fact, online shopping sites use a combination of forms, preset queries, and reports to create a positive user experience. Forms are the pages themselves, which allow us to search and retrieve information (e.g., a product listing) by running preset queries and returning detail in either a form or report format. When we look at a website, we do not see the underlying data tables; we see forms and the results of queries based on forms or formatted as results.

Using Forms in a Database

Forms are objects that are positioned between the data and the individual user. When individuals work with a database using forms, they do not see the table structure with thousands or hundreds of thousands of records. Instead, they see only a specific view designed to make the data accessible and useful for them. Forms are typically bound to the database, which means that the fields within the forms are the same fields as in the database, and any changes in either the form or the table are reflected in the other.

You can also create an unbound form containing buttons and commands that allow an individual to access the database but that does not display any information. An unbound form is likely to be used as an interface to the database and can create a “home page” that makes it possible for someone unfamiliar with the database to easily access and use what they need.

If you were going to build a database that was simple and used only by you, you might not need a form. Even so, a form would make it easier for you to navigate, interact with, and complete tasks. You could build a form that would let you automate tasks that you perform often, like updating inventory. For a single person accessing a database, forms make work more efficient, although they may not be critical to success.

When a database needs to be used by many people, including some who are relatively unfamiliar with the database or with database technology, forms are a critical tool. They provide both an interface to the database and a way to navigate and access data that is pleasant to use, makes sense, is designed for the specific use case, and enables anyone to use the database while ensuring data integrity and security.

The sales and marketing group at WorldCorp has asked you to set up a database that the team can use to look at sales figures for different time periods based on products. The team is not familiar with Access or with databases in general and does not want to spend time learning the technology. Instead, they want an easy-to-use interface that will allow them to search and retrieve summary data. You already have a database with all this information that is ready for use, but it was not designed for the sales and marketing team. You decide to create a form that will allow the team to add a date range and a product type, then return a summarized report showing the necessary sales.

You build a form with this function and then add text that explains the purpose of the form and how to use it. Once the form is finished, you create a dashboard for the team that allows the team to choose this specific form by clicking a labeled button. After you have built and tested the forms, you alert them to the new function. This gives them a powerful new way of looking at sales

data and retrieving the information they need, when they need it, in a format that works for them—all without needing to modify the database or its objects.

Spotlight on Ethics - Addressing Data Security Risks

Employees who work with data need to be aware of data security and able to manage security issues. Nevertheless, many employees may not be aware of the risks that their behavior poses to personal and business data. Even as we work to increase technical security, employees' irresponsible or unethical behaviors may put their own, their coworkers, and even the organization's data at risk. According to *Forbes*, risks to consider include the following:

- Social engineering vulnerabilities: The risk of employees being tricked into revealing credentials or installing malware through tactics like phishing attacks.
- Unmanaged IoT devices: The challenges posed by the increasing number of unmanaged Internet of Things (IoT) devices, which lack proper policies and controls.
- Lack of awareness: The risk associated with employees' inability to detect frauds, scams, and phishing emails, emphasizing the need for strong security awareness training.
- Data loss: The importance of adopting a data-centric approach and implementing data loss prevention measures to safeguard data and ensure compliance.
- Personal data accessibility: The need to ensure that customer data is only accessed by authorized individuals through proper security protocols.
- Managing the increasingly complex digital business environment: The difficulty in consistently following security rules in a growing and complex digital landscape.
- Insider threats: The significance of paying attention to insider threats and ensuring that employees and other individuals with access to corporate data do not misuse their privileges.
- Insecure applications: The risk associated with applications storing and transmitting sensitive data, especially through insecure application programming interfaces and third-party channels.
- Untrained end users: The vulnerabilities arising from employees who fall prey to phishing emails, download malware, or mishandle confidential data, emphasizing the importance of ongoing training.
- Consumer trust in data repositories: The impact of data breaches on consumer trust and the need for comprehensive asset audits and robust segmentation to mitigate such threats.
- Third-party risk: The importance of evaluating and securing third-party relationships to prevent data from being exposed to a wider network.
- Overestimating the ability of network defenses to ensure data security: The need to protect data within applications through encryption, authentication, and secure key management, rather than relying solely on network defenses.
- Misconfigured cloud servers: The risks associated with misconfigured servers in cloud environments and the need for automated security solutions to identify and remediate misconfigurations.
- Lack of data access oversight: The risk posed by granting employees and contractors excessive data access privileges, necessitating continuous data classification and auditing efforts.

Addressing these risks is crucial for businesses to protect their data, minimize potential financial losses, and maintain customer trust and confidence.

Types of Forms

As mentioned, you might decide to create a form as a user interface or a form that accesses and displays information. A form that acts as an interface is likely to be unbound (meaning it is not connected to any specific table or database query) and to include a combination of text, images, and command buttons that access certain objects in the database. These forms can also include search fields if that function is valuable to your audience.

In a bound form, the fields are linked directly to objects in the database. We use bound forms when we want to access data in tables or through queries. Remember, any modification to a field value in a form also updates the underlying data in the table.

Here are some common forms that you can create in Access:

- Simple form: a form that displays and allows editing of a single record at a time.
- Split form: a form that has two functions; for example, it might display a single record as well as a view of the datasheet in which the record is held. A split form is useful when you need to see the larger set of data but work with only one record at a

time.

- Multiple-item form: a form that displays information about more than one record or item at a time. It appears like a datasheet but allows you to add commands, format text, and add other controls.
- Navigation form: a form that allows you to add formatting, text, and commands to act as an interface, or navigation area, for your database. (These are typically unbounded forms.)

Tips for Effective Forms

Creating useful and effective forms requires a clear understanding of the data you have, the audience that will use the form, and the use case for the data. Good forms are more than just boxes on a screen; they demonstrate an awareness of how people will interact with the data and use it on a daily basis. A good form should anticipate user expectations and adapt to meet them. For example, using white spaces to separate information into useful blocks can make it easier for the reader to quickly navigate the form. In addition, you can use the form to control what users can enter into fields. Customers should not be able to enter in numbers as their name for example. To ensure data integrity, you should consider restrictions on what can be entered into forms. If you are using free-form fields, those filling out the form could really put in anything they want, resulting in unusable data for your organization.

Here are some tips for creating effective forms:

- Label your fields (to the left or above the point where data will be entered or viewed).
- Do not use spacing in field names.
- Stack your fields on top of one another to increase readability.
- Mark required fields with an asterisk (*).
- When there are more than five or six choices, use a drop-down menu.
- Break forms into steps if you have many fields to fill in.
- Adjust field size based on the length of the answer.
- Keep related areas on the form together (e.g., all name fields should be together).
- Offer ways for people to get help (including instructions, email addresses, and phone numbers).
- Keep the form as brief as possible; make sure the size fits the use of the form.
- Use white space (blank space) on your form to help your users understand and work with the form comfortably.
- Test your form with different people in the target workgroup and on different technology when possible.
- Do test prints with PDF files to see if records will be visually attractive if saved.

Creating and Editing Forms with the Form Wizard

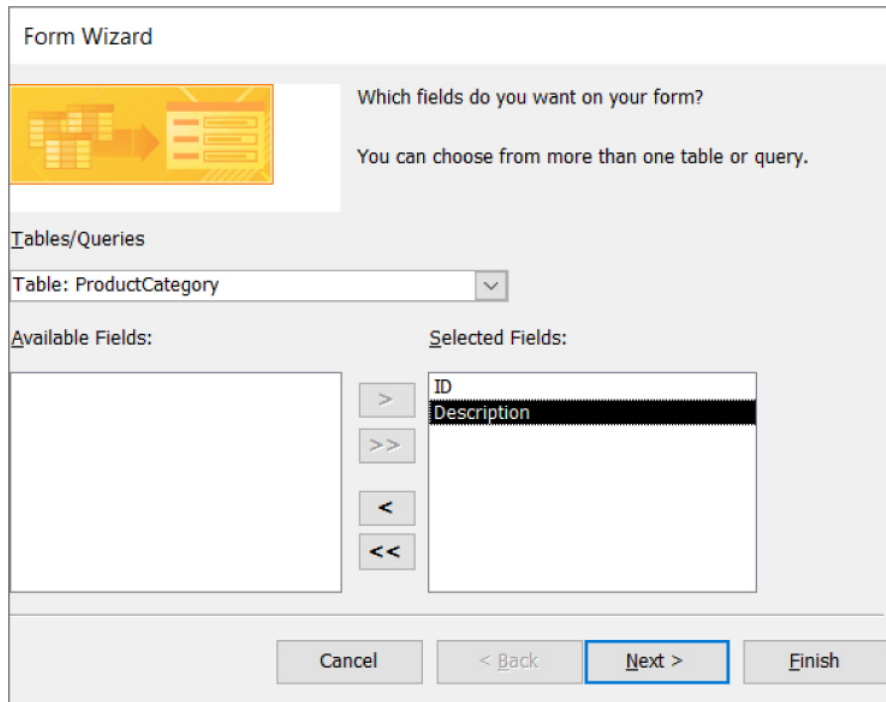
You have several options for creating forms in Access. You can use the Form Wizard, which walks you through the steps to set up a simple form. Alternatively, you can use a preset form, or you can work entirely in Design View, where you have complete control over your layout and contents. Each method has its benefits, and many people use a variety of techniques (e.g., starting with the Wizard or another type of form, then customizing your form in Layout or Design View).

The Form Wizard gives you a quick way to create a form that you can later modify as needed. Like other Microsoft Office Wizards, the Form Wizard asks you a series of questions and uses your responses to format the desired object.

Creating a Form

To start the Wizard, click on the Create tab. Then, in the Forms group, click on Form Wizard. Once the Wizard opens, walk through each screen, making choices at each step to design and complete your form.

Step 1: Choose the table(s) and fields you want to appear on the form. Move fields to the form by clicking on the >> button. Note: To use fields from more than one table, switch tables while on this screen and move the appropriate fields to the right using the >> button (see Figure 13.48).



Form Wizard

Which fields do you want on your form?
You can choose from more than one table or query.

Tables/Queries
Table: ProductCategory

Available Fields:

Selected Fields:
ID
Description

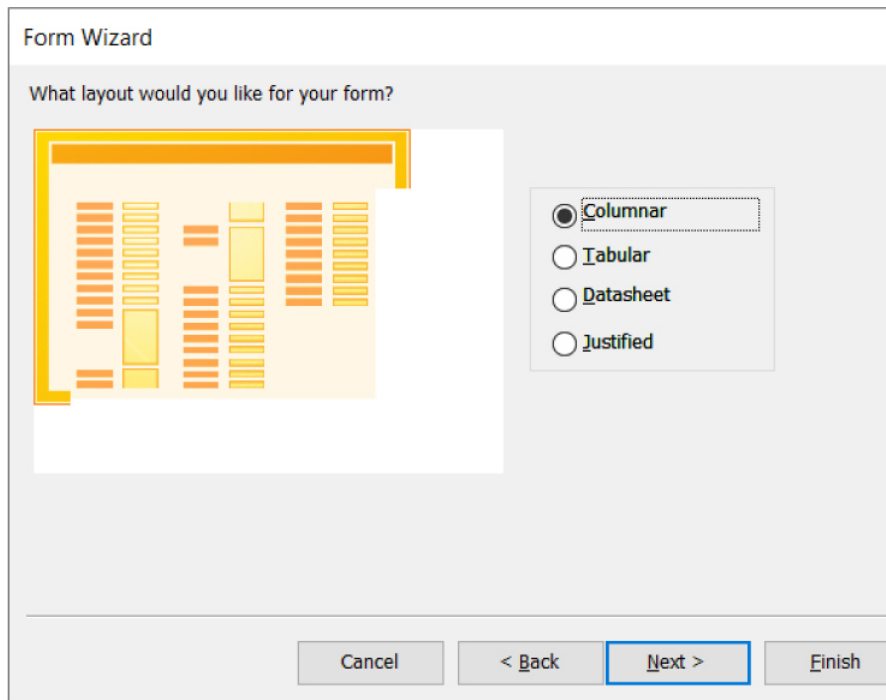
Cancel < Back Next > Finish

Figure 13.48: Using the Form Wizard, you can select fields to include in your form. (Used with permission from Microsoft)

Step 2: Choose the layout you want for the form and click Next:

- Columnar layout shows a single record set up on the screen.
- Tabular layout creates a listing of all records in columns, similar to a datasheet (but formatted differently).
- Datasheet layout gives you a list of records as they would appear in a datasheet.
- Justified layout creates a record-by-record form, arranged across the screen.

Once the form is built, you can modify it to ensure it looks the way you feel is best for your data and audience (see Figure 13.49).



Form Wizard

What layout would you like for your form?

Columnar
Tabular
Datasheet
Justified

Cancel < Back Next > Finish

Figure 13.49: You can modify the layout of a form using the Form Wizard. Some data will lend itself to one format over another. (Used with permission from Microsoft)

Step 3: Enter a unique name for the form and choose whether you want to see the form in design or view mode. Click Finish to create the form.

Forms can also be built from any table or query or a combination of both. One way to streamline setting up a form is to set up a specific query with the fields you want to be displayed on the form. Once the query is ready, you can do the following:

1. Click on the table or query from which you want to build the form.
2. Click on the Create menu, then click the Form command. Access will create a form and open it in Layout View for review and modification.

Using the More Forms Command

If a simple form does not provide what you need for your database, consider using the More Forms command on the Create menu. This command offers four form types, built directly on the table or query you click on in the navigation pane.

To use the More Forms command:

1. Click on the table or query on which you want to base your form in the All Access Objects navigation area.
2. Click on the Create menu and then the More Forms command under the Forms section.
3. Choose the type of form you want to work with. Access will immediately create the form, which you can then modify to suit your needs (see Figure 13.50).

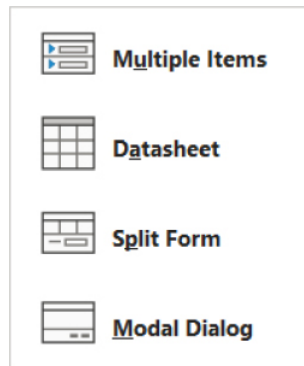


Figure 13.50: You can select the type of form you wish to create using the More Forms command. (Used with permission from Microsoft)

Reviewing a Form

Regardless of the method you use, once your form is created you can view it either in view mode (which shows how your audience will view the form) or in Layout View or Design View, which allows you to modify the form (see Figure 13.51). To change your view, click on the Home tab and then on the View command. Under the View command, choose one of the following:

- Form View: to see how the audience will view the form
- Layout View: to modify the layout and formatting of the form
- Design View: to modify any and all attributes of the form

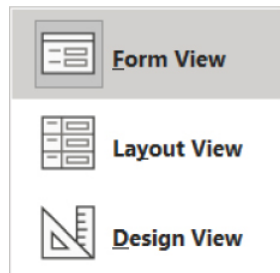


Figure 13.51: As with viewing reports, there are three different options for viewing a form. (Used with permission from Microsoft)

The forms that Access builds can be used as is, but you will often want to add to a form or slightly adjust its formatting. You can do this in two views: in Layout View, which allows you to change the overall layout and formatting, and in Design View, which allows you to change any attributes in the form.

When you edit a form in either Layout View or Design View, you will have the ability to work with fields and controls. Fields are columns within the database where data is stored. You can add and remove fields from your form to ensure that you display the data you need to your audience when they work with the form. Fields in the form reference field values in the database and display information from the database. When you are in Layout View, you can see this as you work; in Design View, you will see the design elements but not values.

You can also add controls to a form. Controls are items that are part of your form and can enhance its usability. Bound controls are connected (or bound) to a specific field, table, or query in your database. Unbound controls are not tied to the database but exist only in your form, where they run a specific command or action or display specific information.

Some common controls in Access include text boxes (see Figure 13.52), labels, buttons, and links. Each of these controls adds functionality to your form. You can add a command to open a form in Design View by clicking on the specific command in the Controls group on the Form Design tab in the ribbon.

Depending on the control you add, you may need to work with the Control Wizard, complete a screen, adjust properties, or complete the control on screen. When you begin working with forms in Access, you may work with the Control Wizard and then preview the form in the View menu to see how the control works in practice.

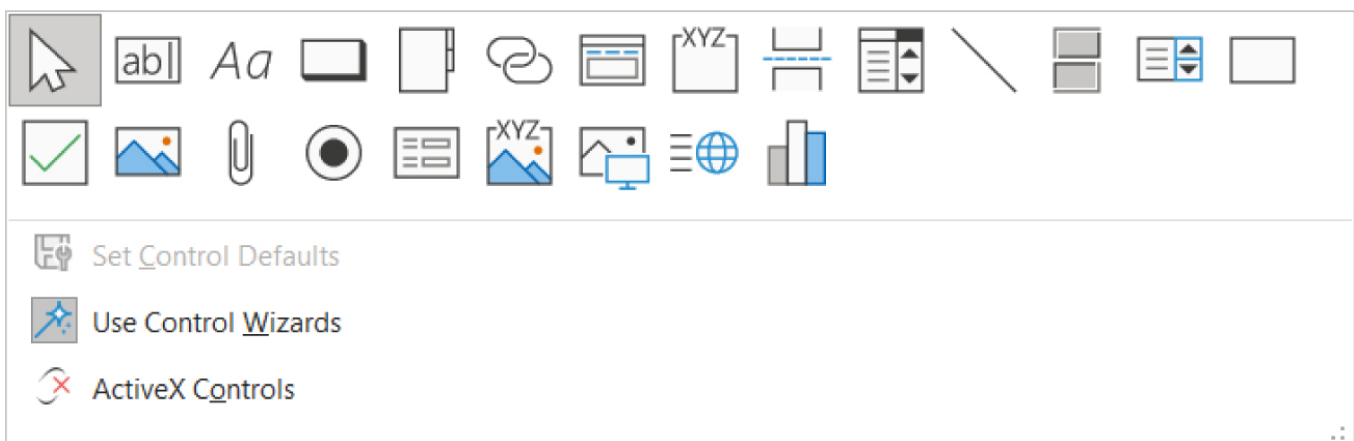


Figure 13.52: Adding controls to a form can be easily done with the Control Wizard. (Used with permission from Microsoft)

Note that in Access, a text box is typically linked to a field in a database. To add static text to a form, you should instead use a label. Labels do not change on the form based on the data within the database, unlike text boxes.

Customizing Forms

In both the Layout View and the Design View, you can customize elements of the forms to meet your needed. This goes beyond simply adding, reordering, or removing fields from your form. You can change the size of the field and the caption that the users see for that field. In the Design View, you can change the size of the fields by simply clicking on the field box and resizing the box. To change the caption for the field, double-click on the text in the field, and the Property Sheet screen will appear on the right side of the screen. Conversely, you can simply click on the Property Sheet tool in the Tools command group on the Form Design tab to change the captions. Through the Property Sheet tool, you can also adjust items such as the font, color, background color, and border of the fields in the form. You can adjust these settings for each field independently. This means that one field could have different settings than another field. More is covered about the Property Sheet options in the following section.

You might want to add header or footer information to your form. Typical information contained in a form header would be specific company details such as the company logo, address, and contact information. WorldCorp has standard header information that should be included on all files as was covered in previous chapters. This information is customized on the form through the Form Design tab and the Header/Footer command group. You can have text, images, and the date and time added to the header. This information can also be added in the same manner to the footer. To adjust the text, simply click on the existing text in the header or footer and replace with the text you want to include. Note that Access will give a header to the form based on the name of the table used to create the form. You will likely want to replace this header with something more descriptive. For example, if you are creating a form based on a table called “Sales Data,” the form header will be “Sales Data” as well. You could replace that with something like “Sales Data for Region 3, 2023.”

In summary, customizing forms in Access is similar to customizing reports, which we covered earlier in this chapter. You can generally customize just about every aspect of the form from the font, to colors, to adding images, and other related aspects.

Layout View

The Layout View screen gives you a good look at your form, with real data showing on the screen. This enables you to modify the screen's visual appearance, including the size and placement of controls. Not all tasks are available in Layout View; depending on your goal, you may find that you need to change to Design View.

To modify a form in Layout View, start by opening the form in Layout View. Then, proceed as follows:

1. To add a field, click on the Format tab in the Controls group; then click Add existing fields and drag the field to the form.
2. To move a control, click on any instance of the control and drag it to a new location. (Note: You will have limited options for moving a control.)
3. To resize a control, click on the control, select its border, and click and drag until it is sized properly.
4. To add an image to the page, click on the Insert Image command on the Controls group and choose the file. Once the image is on the form, you can move and resize it.

These are only a few of the options available to you in Layout View; to see more options for customizing your form, hover over the command on the ribbon and read the description. Remember, what you see on screen in Layout View reflects the final form, making it easier to set up a form that meets your needs and has the appearance you feel works best for your data and your audience.

Design View

If you find that you need to make changes to a complex form, you will likely want to switch from Layout View to Design View. The Design View screen gives you complete control over the appearance of the form but is more complex to use. You may want to use Design View for tasks such as moving elements and controls, adding images, adding simple controls, and changing form properties.

In Design View, you will see tabs that control how you lay out the form. You may also see floating menus and boxes that appear depending on the items you click on:

- The Design tab contains commands that control the overall design scheme (colors and fonts), controls you can add to the form, and commands that allow you to insert items and change properties of items on the form.
- The Arrange tab contains commands that allow you to line up items; insert, move, and arrange items; and manage the layout of items on the page. (Note: This tab may not be available unless you select more than one item, such as an image or a control.)
- The Format tab contains commands that let you format the text on the form and add background images and formatting.

To move items manually, click and drag them or click on multiple items and use the commands on the Arrange tab on the ribbon. To move a single item (control, image, text), click and drag the item. To move more than one item, hold down Control, click on each item, and then move all of them at once.

Adding an Image

It can be beneficial to add an image, such as a logo, to a form for customization. You may also create a text explanation or block of text in a graphics program and then insert it into the database as an image.

To add an image in Design View, click on the Form Design tab and choose the image that you would like to add. You can add any image to the page by clicking on Insert Image and locating the file (be aware that if you insert an image in the detail section of the form, it will repeat in each row). To add a logo, click on Logo in the Header/Footer group and the image will appear at the top of your form, in the header.

In addition to images, you can also add a title or date and time in the header/footer section of the form. These items will be displayed for your users but will not be stored as data in the database.

Changing Form Properties

Properties describe how a form control or field acts in a form. Form controls determine how the form works. There is a long list of properties that you can control; initially, you will likely only change basic properties such as the width of a screen or the addition of scroll bars (see Figure 13.53).

To change a form property:

1. In the open form, click on the command Property Sheet (Form Design, Tools group). The property sheet will open.
2. In the property sheet, verify that Form shows as the selection type.
3. Modify the properties you would like to change, and click on the category of property you would like to work with (e.g., you might want to format items on the form).

Here are a few examples:

- To add a caption to the form, click on the Format tab within the property sheet and add caption text.
- To change the default view, click on the default view and switch the type of form.
- To control the width of the form, set the width in the width row.

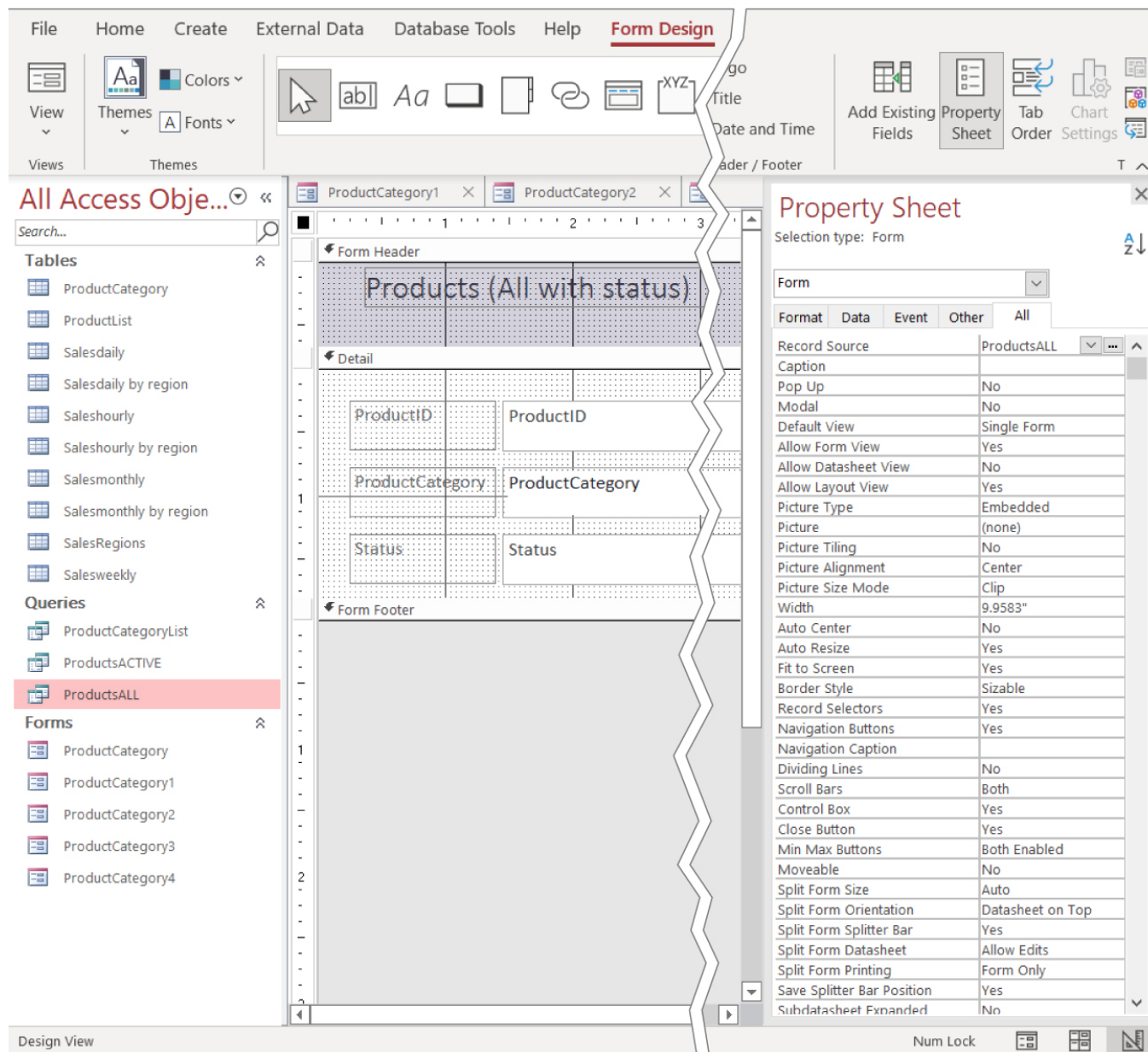


Figure 13.53: Each form contains a number of properties that can be edited and customized depending on the purpose of the form.
(Used with permission from Microsoft)

When you create a form or modify an existing form, Access will prompt you to name and save it. You can save manually by right-clicking on the name of the tab in the work area and choosing the Save command. To close a form that is open in the work area, click on the X next to the tab name.

This page titled [2.1.3: Creating Forms in Microsoft Access](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

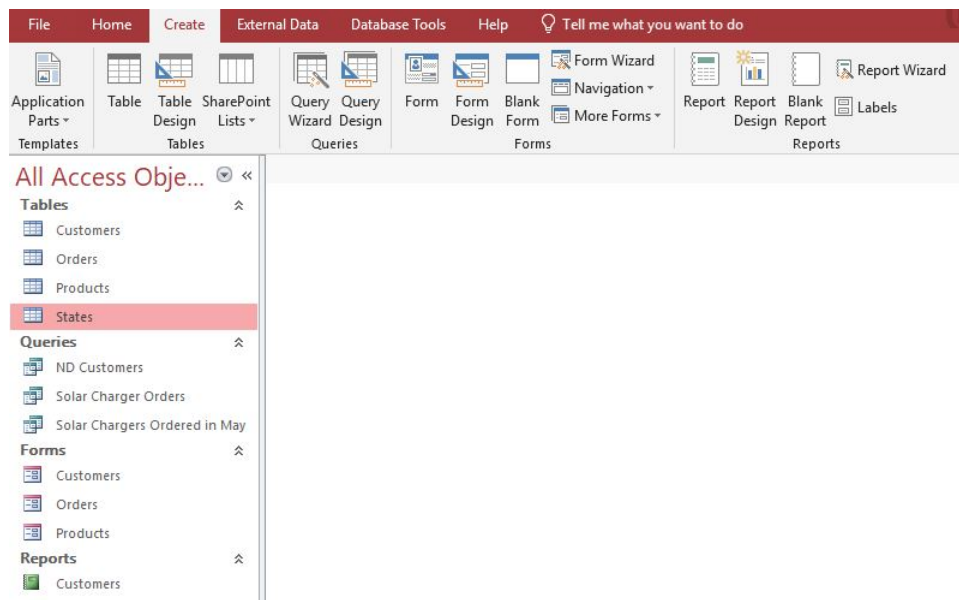
- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.6: Creating Forms in Microsoft Access** by [OpenStax](#) is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.4: Form Basics

[Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

A form is a database object that offers an alternative interface to a table beyond the Datasheet view of the table. Forms can improve data entry proficiency by rearranging the order and tab sequence of fields within a table. Forms can also be designed to hide certain fields from appearing. For example, a credit limit or credit card number field might only be accessible from the table Datasheet view to limit the access to sensitive data from users who do not have rights to view or edit tables. In Access, users have the option of creating a blank form from scratch, using the Form Wizard, or having Access automatically create a form from a table or query.

A basic form can be easily created by using the Form tool. Simply select the table (or query) in the Navigation Pane that the form should be based off of, and then click Create > Form (from the Forms group). All the fields from the underlying data source are placed on the form. If Access finds a single table that has a one-to-many relationship with the table or query that you used to create the form, Access adds a datasheet to the form that is based on the related table or query.

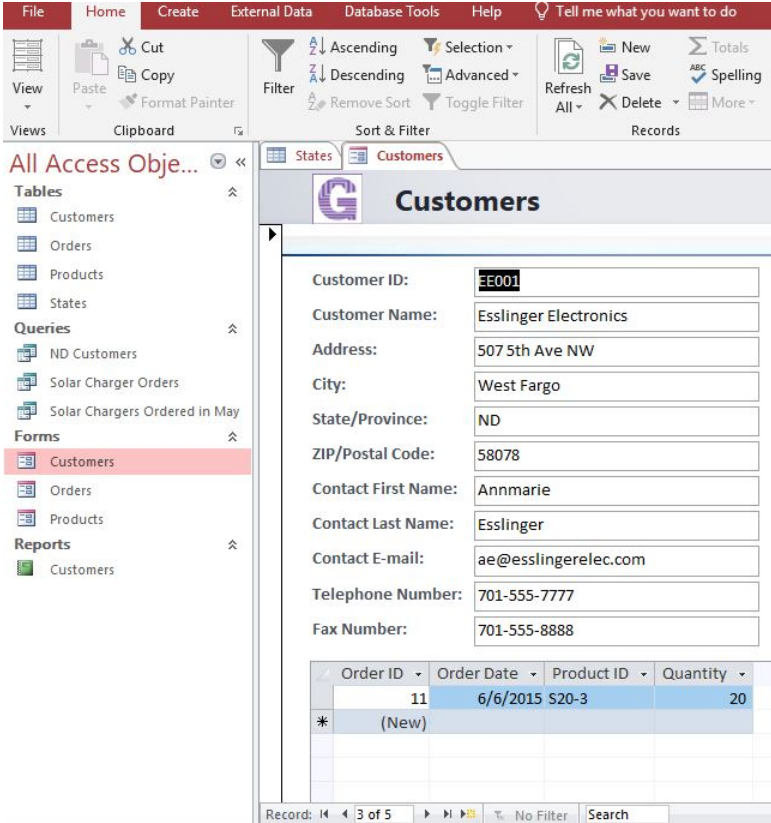


Access Create tab

A split form can be created by clicking More Forms from Create tab, and then choosing Split Form. A split form displays two views of the data simultaneously — a Form view and a Datasheet view. Working with split forms provides the benefits of both kinds of forms in a single form. For example, a user can use the datasheet portion of the form to quickly locate a record, and then use the form portion to view or edit the record.

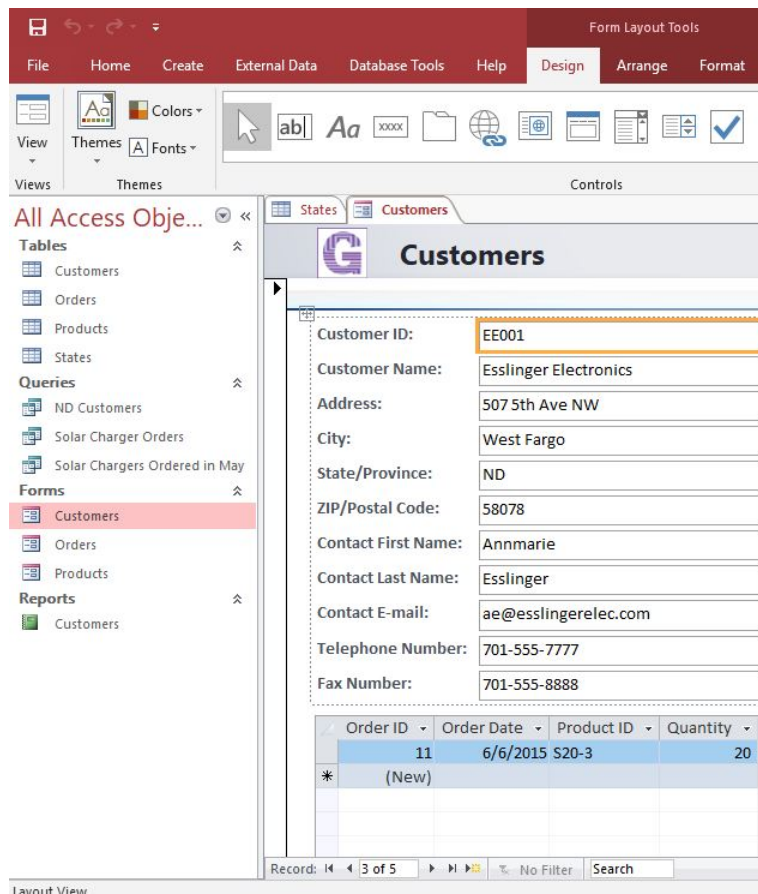
To be more selective about what fields appear on your form, use the Form Wizard. This multi-step process allows the user to define how the data is grouped and sorted, and in some instances, fields from multiple tables/queries can be added to the same form. Once a form is created, use the different views to modify the forms design or populate the forms with data. Forms have three views:

- **Form view** – shows the data in the form. This is the view to use to enter or change data. Record controls at the bottom of the form are used to display specific records and add a new record (the right arrow with the yellow star). The form's design properties cannot be changed in this view.



Form view

- **Layout view** – shows the form and the data in a visual layout. This is the view that opens when a form is initially created. Some of the form design properties, such as field fonts and column widths, can be changed in this view. However, the data cannot be changed in this view. Notice the Form Layout Tools contextual tabs!



Form Layout Tools

File Home Create External Data Database Tools Help Design Arrange Format

Views Themes

All Access Objects

Tables

- Customers
- Orders
- Products
- States

Queries

- ND Customers
- Solar Charger Orders
- Solar Chargers Ordered in May

Forms

- Customers
- Orders
- Products

Reports

- Customers

Customers

Customer ID: EE001

Customer Name: Esslinger Electronics

Address: 507 5th Ave NW

City: West Fargo

State/Province: ND

ZIP/Postal Code: 58078

Contact First Name: Annmarie

Contact Last Name: Esslinger

Contact E-mail: ae@esslingerelec.com

Telephone Number: 701-555-7777

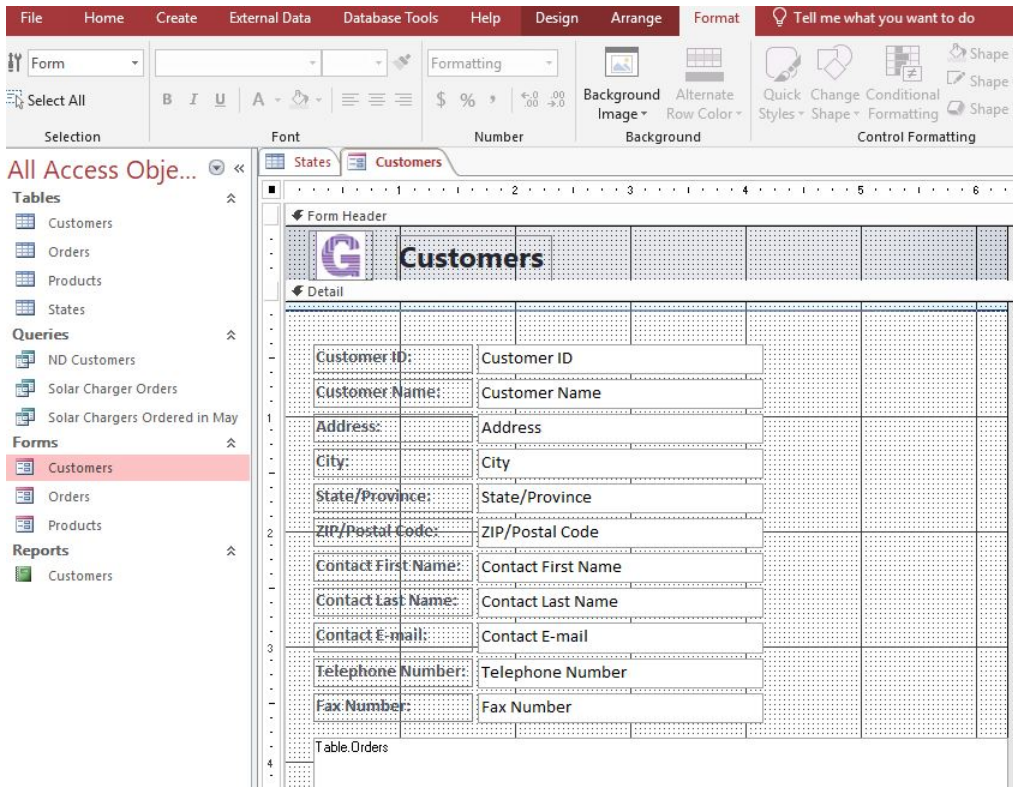
Fax Number: 701-555-8888

Order ID	Order Date	Product ID	Quantity
11	6/6/2015	S20-3	20
(New)			

Record: 14 3 of 5 No Filter Search

Form Layout view

- **Design view** – shows the form design, but not the data. Any aspect of the form's design can be changed in this view, but the data cannot be changed. The Design, Arrange and Format contextual tabs appear for this view too. The *Design* tab allows the user to modify the color, fonts or theme of the form. A logo, title and/or date/time field can also be added to the form. Form controls (buttons, text boxes, combo boxes, etc) can also be added. The *Arrange* tab is used for modifying the order and position of the fields on the form. The *Format* tab allows the user to modify the style of the form including the fonts and a background image can be added.



Form design view

A user can switch between views as often as desired while customizing the form. Switching views can be done by using the View button on the Ribbon, or the three icons at the bottom right of the Access window.

Note

If a field in a table has been modified after the form is created, the existing form does not automatically update. The form can be modified, but sometimes it is easier to delete the existing form and recreate it based off of the new table settings.

This page titled [2.1.4: Form Basics](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [Marcus Lacher \(Minnesota Libraries Publishing Project\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **6.8: Creating Forms** by Marcus Lacher is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

2.1.5: Customizing Forms

 [Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Use the tools to build a custom form
- Modify and add controls to a form
- Customize the appearance of a form by including images and modifying fonts

WorldCorp has manufacturing locations in the United States, China, and Germany, with headquarters in key metropolitan areas in those same countries. Smaller functional units are located in offices around the world, and there is an established international remote workforce. As you continue to develop an inventory database for potential retail locations its customers experience, forms will help users add data records to fields within tables. Those working in the new retail locations become the users of forms. As they receive shipments of inventory or create sales transactions for customers, their input is what truly creates the data. Above all, a form should be functional for them. Beyond functionality, a form that is customized in a way that makes it easier and faster to use will create a more enjoyable user experience. Consider our diverse workforce around the world. Knowing how to customize a form for cultural business practices is crucial in supporting the user. This will be no different as you develop and design an inventory database for a potential booming retail arm of WorldCorp.

Typically, one of the first forms all users experience is the Main Menu at WorldCorp, which helps them navigate through all the other forms available to them in the database. Rather than scrolling through the entire list, the user can go to the Main Menu, a starting point for all navigation.

Link to Learning

A quick way to prioritize a Main Menu for any graphical user interface is to manage its settings. A Main Menu can be created either all at once quickly or over time, and the process can be creative and fun. Be sure to set the Display Form option to Main Menu so that when the database is first opened, the Main Menu form is opened up and ready for the user.

This [video on creating a menu GUI in Access 2013](#) discusses the basic steps in creating a Main Menu and, most important, walks through the display options.

Building a Custom Form

In general, using the Form Wizard will be the quickest and most efficient way to create a custom form. We covered how to do that in Understanding and Using Databases. Here, however, we will walk through the steps of building a form from scratch, in Design View, to demonstrate all of the possibilities. Start by selecting the Query Design option under the Create tab in Access, and you will create a blank form in which you can start working. Figure 14.35 shows the Detail section that appears when you create a blank form.

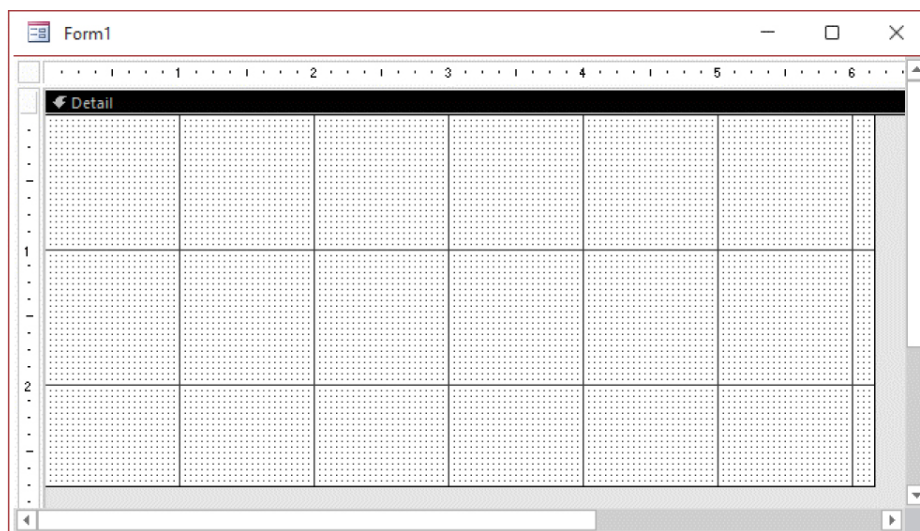


Figure 14.35: From the Create tab, choose Form Design to create the form from scratch. (Used with permission from Microsoft)

The next step is to save the form (frmMainMenu) and start planning how you want the Main Menu to look and operate.

Form Views and Tools

Right-click on the top of the form to see the list of options, as Figure 14.36 shows. Here, we will review the four ways to view a form: Form View, Layout View, Design View, and Datasheet View.

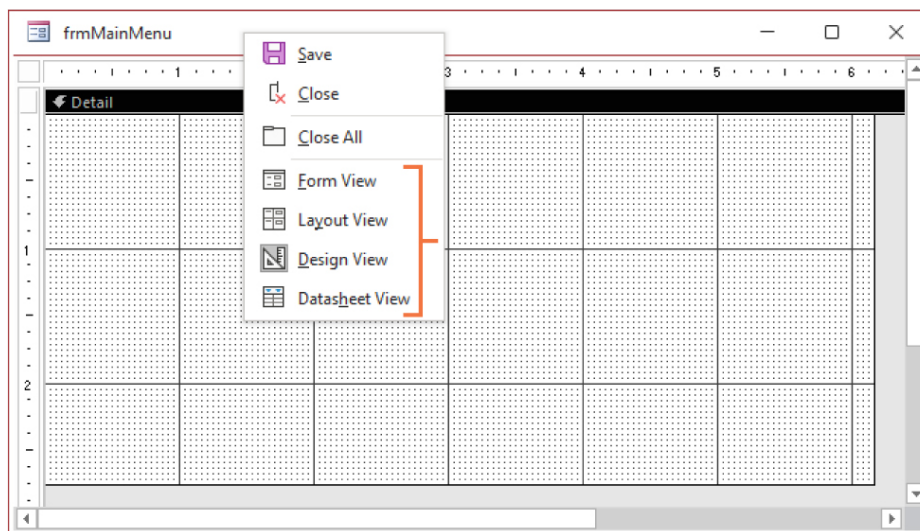


Figure 14.36: To begin a new form, you must first save it. (Used with permission from Microsoft)

Form View

Form View shows us what the form will look like for the user. Data can be changed in Form View, but formatting cannot be changed. Whenever a form is launched, it will first be displayed in Form View. It is recommended to continually check the Form View while creating a form to make sure that the final version of the form displays all of the desired options.

Layout View

Layout View is a feature that allows us to design and customize the layout of forms and reports visually, by dragging and dropping fields and other objects onto the design surface. It is helpful for inexperienced users because it provides a more intuitive and user-friendly way to create forms and reports than the traditional Design View. In Layout View, you can see how your form or report will look as you design it, and you can easily move fields and other objects around to create the desired layout.

For example, let's say you want to create a form to enter data about damaged items that need to be removed from inventory. In Layout View, you can drag and drop fields such as Item Name, UPC Code, Quantity, Description of Damage, and Date onto the form, and then arrange them in a way that makes sense for your users. You can also add other objects such as labels, images, and

buttons to make the form more user-friendly. Overall, Layout View is a helpful tool for creating forms and reports in Access, particularly for the least experienced users, who may not be familiar with the intricacies of Design View.

Design View

Design View is a feature that allows us to create and modify the structure of tables, queries, forms, and reports in a more detailed and precise way. Similar to Layout View, in Design View for forms and reports, you can add and arrange fields, labels, buttons, and other objects, and set their properties to control their behavior and appearance. However, Design View provides a more granular level of control over the structure and properties of your database objects, allowing you to create more complex and customized solutions. It also allows you to view and modify the underlying SQL code that Access generates for your objects, which can be helpful for troubleshooting and advanced customization.

Design view is a powerful tool for creating and customizing the structure and functionality of tables, queries, forms, and reports in Access, and is helpful for naive users who want to create more complex and customized solutions.

Datasheet View

Datasheet View is similar to what you might see in a spreadsheet program. The data is in a table grid like Excel. This allows you to interact with the information in a form similar to how you would work in Excel. You can add or delete records (rows) and add or delete fields (columns) easily in this view. You can also use some of the common keyboard shortcuts for copy and paste.

Having this view option can depend on which version of Access you are using. As of 2021, Datasheet View was removed from forms in Access. So, if you are using a new version of the app, you will likely not have the Datasheet View option.

Using the Form Tool

A list of form tools is available within the Form Design tab on the right side of the ribbon. To add existing fields to a form, click the Add Existing Fields command (see Figure 14.37). The Field List pane will appear, and you can select a field or fields to add to your form.

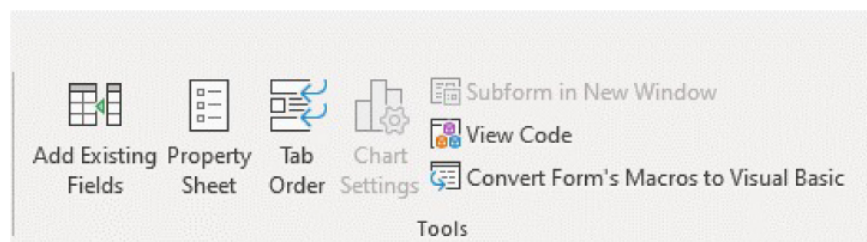


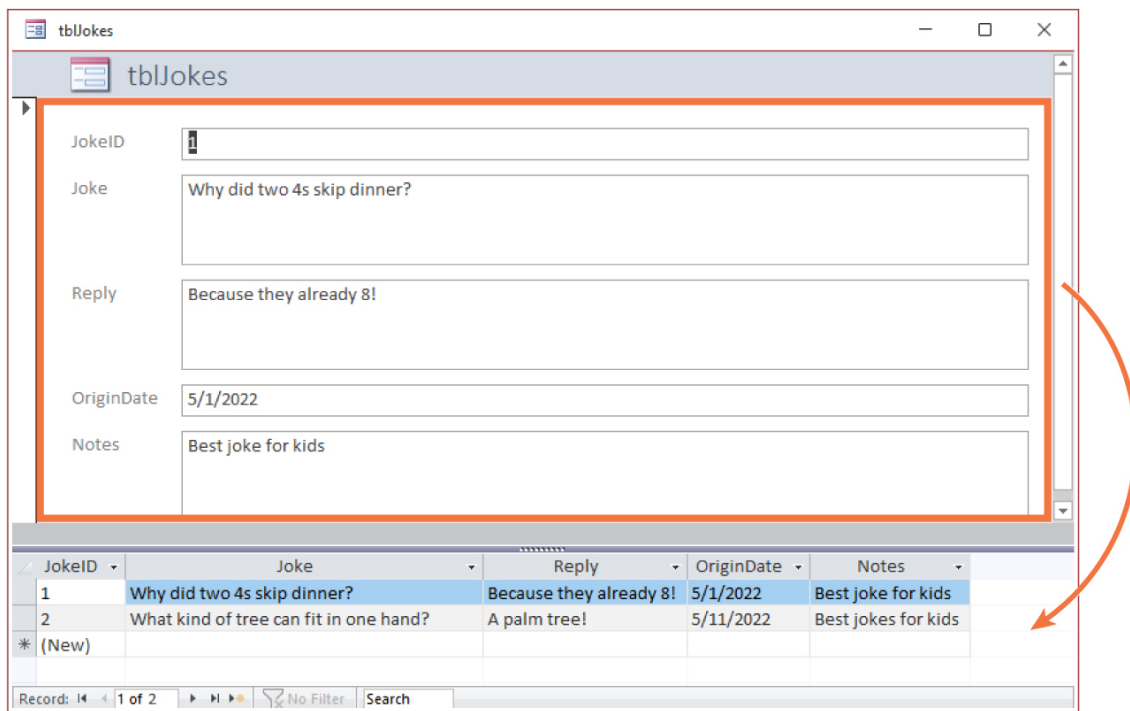
Figure 14.37: You can begin to add elements to your form such as fields using the Tools command group. (Used with permission from Microsoft)

An option set in Design View on the property sheet is called a form property. These options control how the form looks, works, and interacts with the rest of the database. Use the tabs to find the specific property you want to set, and use the help available to you in the bottom-left corner of the Access window to learn the function of each property setting. In many cases, the default setting may be adequate for getting started.

You can use tab order to help control which section of the form the cursor moves to next. This feature provides ease-of-use for gathering data in a user-friendly way. When adjusting tab order, always consider the layout of the form: What is the first thing the user will do, what is next, and so on? Also consider omitting any fields the user will simply tab through—for example, autonumber fields. These types of fields can be omitted altogether for a better user interface experience.

Using the Split Form Tool

After building a table, you can create a split form. Select the table from the All Access Objects list and then go to the Create tab. Under More Forms, select Split Form. In this example, Figure 14.38, you can collect a list of your favorite jokes. The split form shows the datasheet (a running list of jokes) in the lower section, outlined in yellow, while allowing access to the single-form information outlined at the top.



JokeID	Joke	Reply	OriginDate	Notes
1	Why did two 4s skip dinner?	Because they already 8!	5/1/2022	Best joke for kids
2	What kind of tree can fit in one hand?	A palm tree!	5/11/2022	Best jokes for kids
* (New)				

Figure 14.38: Split form view allows you to the information as it would look in the form as well as the tabular view showing each record in the table. (Used with permission from Microsoft)

Using a split form may be helpful when you are scrolling through a datasheet to make changes, because it enables you to highlight a row of data in the single-form view. As you have seen, a common theme in form design is making the process easy and intuitive for the user.

Adding Controls to a Form

A form control is the part of a form that is used to enter, edit, or display data. Remember that controls can be bound, unbound, or calculated. In Understanding and Using Databases, we covered how to use the Control Wizard to walk you through adding controls to your form. Here, we will go into more detail about the different types of controls you can add, and how to add them without the wizard.

As an example of adding controls, start by creating a Main Menu for a multiuser-friendly database, using the example of a carpeting business, as Figure 14.39 shows.

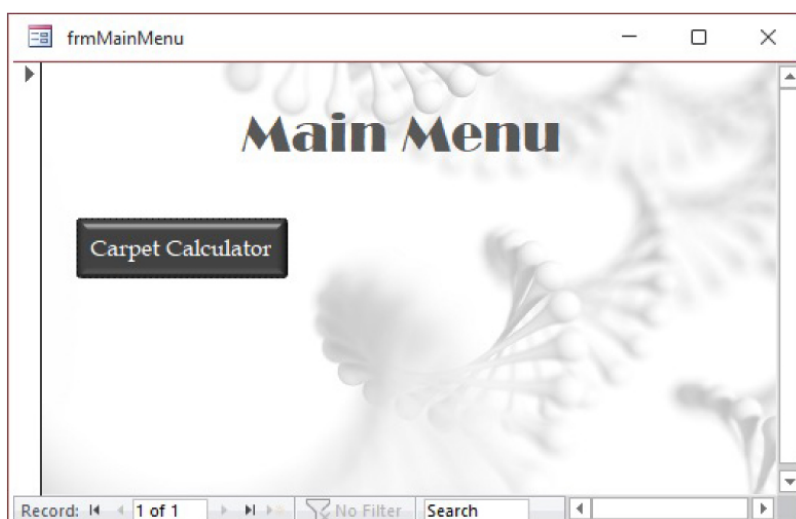


Figure 14.39: Controls are used to customize the form by inserting images, text boxes for titles, and other related elements. (Used with permission from Microsoft)

When helping a customer, an agent of the business would begin by opening the Main Menu form. Notice that a form control, Carpet Calculator, has been added to the Main Menu as the first control. To respond to a customer's call, the agent will most likely need a form to calculate a potential quote for a project. The form control that was added is a button, one of many options available in the Controls section of Form Design.

Common Controls

Controls can be placed on forms to add to their functionality. This section walks you through the procedure for adding several common controls.

The command control is used to start an action or a set of actions. For example, the button in Figure 14.39 is designed to open the Carpet Calculator form. The Command Button Wizard shown in Figure 14.40 appears automatically when you add a button to a form in Design View. When you select Form Operations within the Categories field, followed by Open Form within the Actions list, the wizard records a macro or event procedure and attaches it to the command button's On Click property.

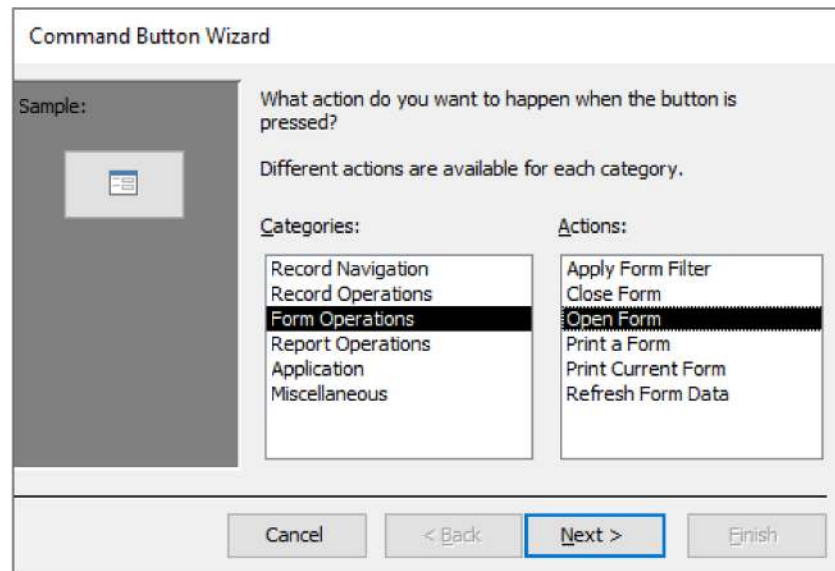


Figure 14.40: You can further customize the form by adding a command button to perform a specific task such as closing or printing the form. (Used with permission from Microsoft)

Text Boxes

Text boxes are great items to elevate the use and function of any form. They can be bound to a field or unbound. You can use an unbound text box to display the results of a calculation or to accept input that you don't want to store directly in a table, such as the current date and time—information that is not connected to a field in a table or query.

It is easiest to add an unbound text box in Design View. On the Design tab, in the Controls group, click Text Box. Position the pointer where you want the text box to be placed on the form or report, and then click to insert the text box (outlined in yellow in Figure 14.41). In this example, the expression =Now() was added to the unbound text box within the property sheet (circled). (Note that Access also places a label to the left of the text box. In this example, the label was deleted, as the current date and time do not require a label.)

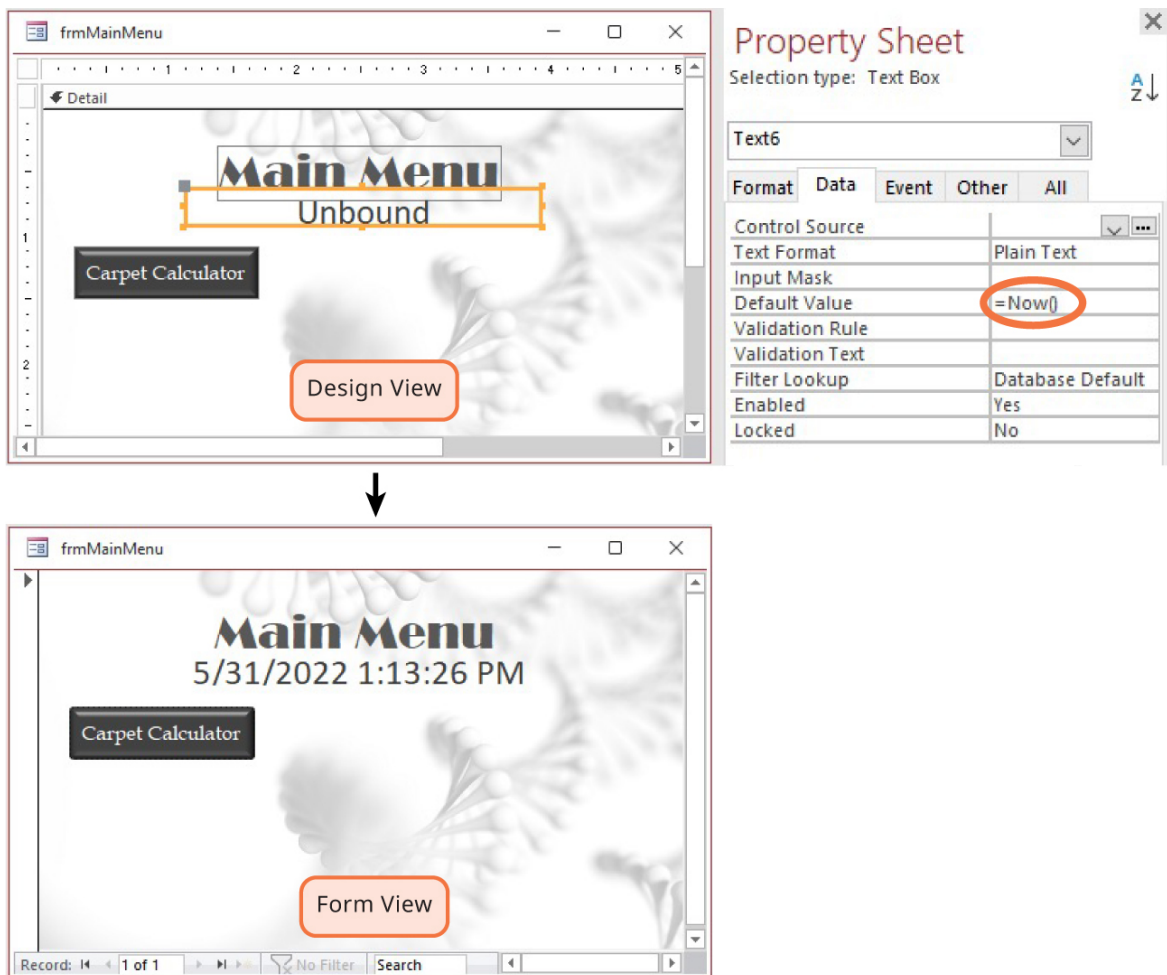


Figure 14.41: Elements such as the date and time can be added to text boxes when inserted into the form. (Used with permission from Microsoft)

Combo Boxes

A combo box is an object or control that you can also place on a form. It displays a list of values from which a user can quickly select. The values displayed within a combo box can be populated by a table/query, value list, or field list. Combo boxes are commonly used in forms to provide users with a list of choices to select from, while also allowing them to enter a new value if none of the predefined options meet their needs.

Combo boxes are helpful for numerous reasons. As an example, they can simplify data entry by providing users with a list of predefined options to choose from, which can help reduce errors and ensure data consistency. In addition, combo boxes can be set up to validate the data entered by users, such as by checking if the input matches a specific format or if it exists in a related table. This helps ensure the data entered into the database is accurate and complete. In consideration of your users in retail, combo boxes can increase efficiency by allowing retail associates to quickly select from a list of options, rather than having to manually enter the information every time they handle any of the inventory.

In the example shown in Figure 14.42, a New Customer form is designed with two combo boxes: one for State and the other for City. These two combo boxes were designed differently. The State combo box was built from a table of state names and abbreviations, while the City combo box was built from a query of city and state names. The query is set to display only the conditions selected by the State combo box.

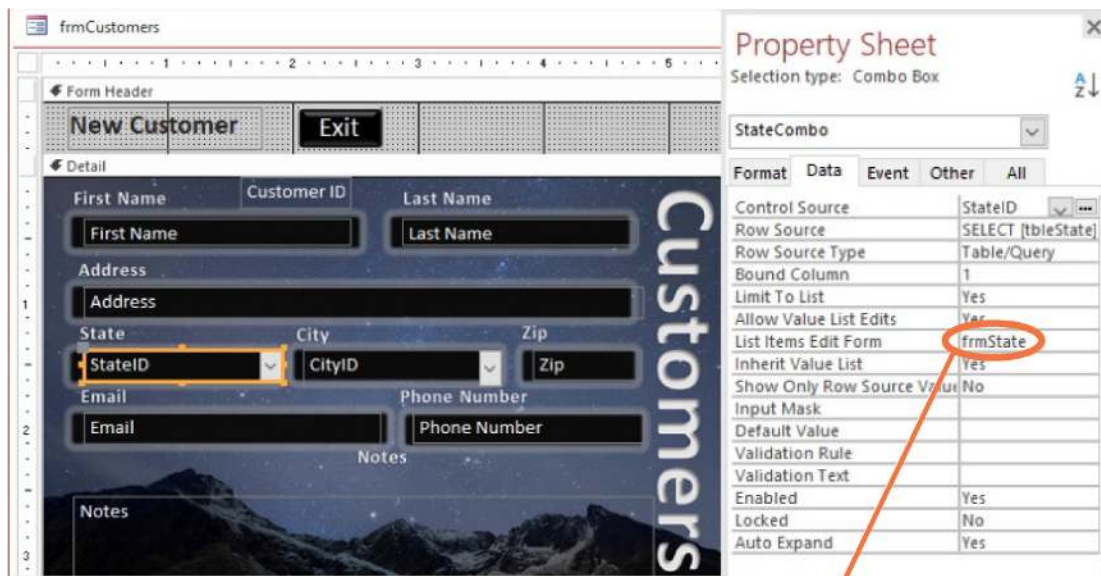
(a)

(b)

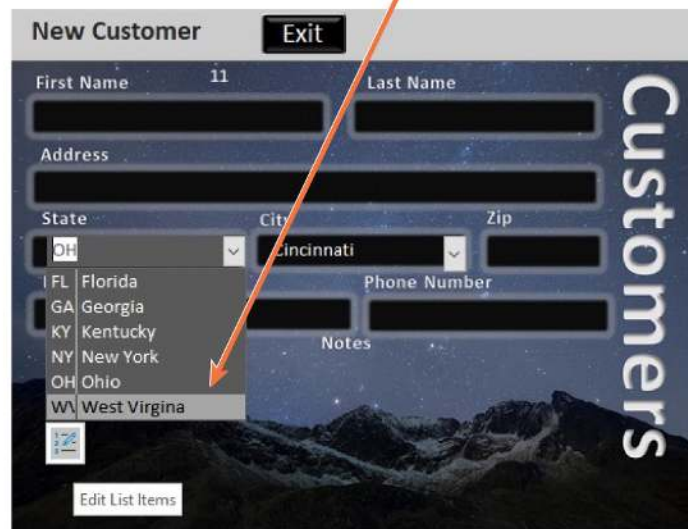
Figure 14.42: Drop-down menus can be helpful to reduce the amount of information the user needs to type manually into the form. These menus can be formatted and built in different ways, as evidenced by the State combo box in (a) and the City combo box in (b). (Used with permission from Microsoft)

An additional option that can be added to any combo box is the ability to update the list provided in the combo box. In the example, only a couple of states and cities are listed, so you may want to include a shortcut for adding a new state or city to the table. Combo boxes maintain spelling integrity.

In this example (see Figure 14.43, frmState (circled) is a form designed to update the tblState table. This form was then added to the List Items Edit Form criteria on the associated property. Users can select List Items Edit Form option, located in the property sheet, which launches the frmState form to update the table.



(a)



(b)

Figure 14.43: A table in the database (a) can be used to populate the options in the combo box so that if edited, (b) the form is automatically updated as well. (Used with permission from Microsoft)

Changing the Appearance of a Form

This section discusses changes that make a form easier to use and more appealing overall, and that will allow for personalization of the project, including company branding.

Before you start to manipulate the appearance of a form, consider creating a template for both primary default views: Single Form View and Continuous Form View. It is important to have a starting point to build from if you anticipate creating numerous forms for the project, all of which will require formatting changes. In this section, you will learn how to add headers and footers, add a title, modify text, control colors, and, finally, add logos and custom imagery. This will be the starting point for you to explore your creativity.

Adding and Modifying Headers and Footers

Adding Header and Footer sections to a form in Access takes just a few simple steps. Start by displaying the form in Design View (see Figure 14.44). Right-click on the grid area on the design surface and select Form Header/Footer from the shortcut menu.

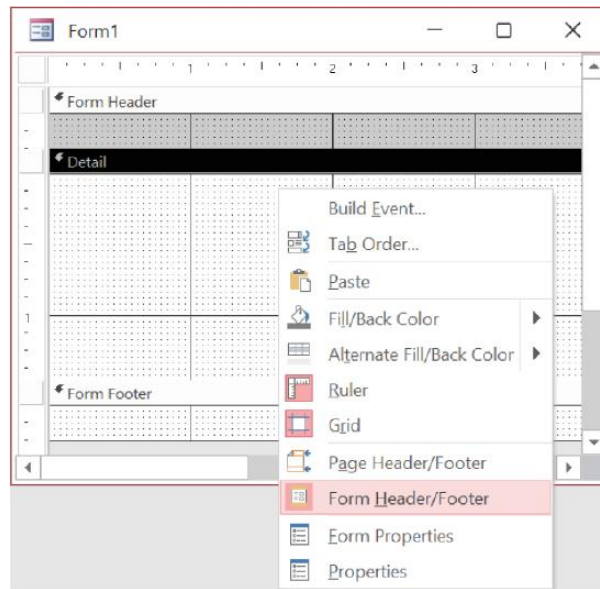


Figure 14.44: You can include company information or logos to each page of the form by adding a header. (Used with permission from Microsoft)

Adding a Title to the Form

A clear and descriptive title can help users quickly identify the purpose of the form and understand how it relates to the overall database. This may be the very first thing they see when the form opens. A well-crafted title can communicate important information about the form, such as the type of data it contains, the purpose of the form, or the specific function it serves within the database. This can help users understand how the form fits into the larger context of the database and improve their overall understanding of the system. In addition, consistent use of titles can help maintain a cohesive and organized database structure, making it easier for users to locate specific forms and navigate the database overall. In consideration that you're a WorldCorp employee, adding a title to a form can add a level of professionalism to the database and make it look more polished and organized. This can improve the overall perception of the database.

Often, the original form design is built around the title. When you envision a WorldCorp user interacting with a form you designed, consider the title placement, size, and font selections in correlation with the company branding.

As shown in Figure 14.45, you can add a title from Design View, on the Form Design Tools tab in the Header/Footer command group. Select Title (circled) to add a text box to the Header section of the form (Figure 14.46).

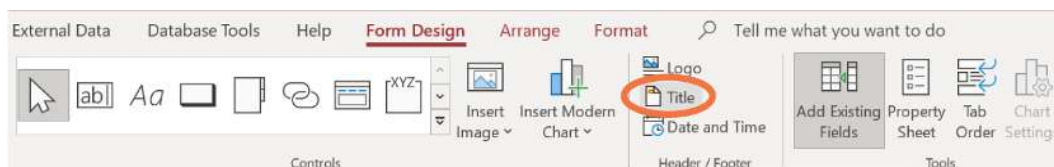


Figure 14.45: Including the date and time in the header could be helpful especially when sorting or tracking information from completed forms. (Used with permission from Microsoft)

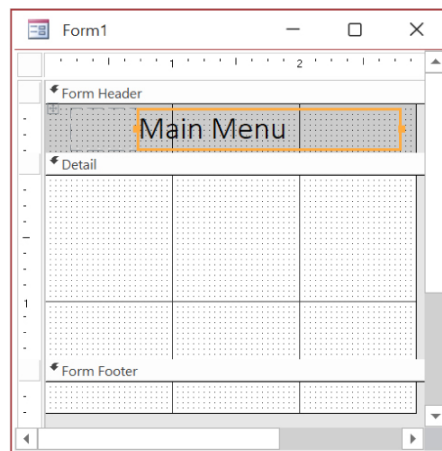


Figure 14.46: The title of the form should be descriptive of the data you will collect in the form. (Used with permission from Microsoft)

Adding and Formatting Text

In today's modern world with information at our fingertips, marketing toolkits (MTs) are often available and are expected to be used when available. Numerous company websites now have highly detailed information on their branding. As an example, [Tennessee Tech](#) is one of many institutes that make branding information available to both employees (faculty and staff) and customers (students). This enables their current workforce and stakeholders to post on social media, post flyers, or add branding information to newsletters as a way of promoting the company culture. MTs provide users with the various elements the company requires—all promotion, training, and public relation materials. From the color scheme to the font to the logo placement, expectations in design are being elevated every year.

In the current example, make sure to select the text box that you want to format. If you want to apply the same formatting to multiple text boxes, click and hold Control on the keyboard to select multiple text boxes before you apply the desired formatting (see Figure 14.47).

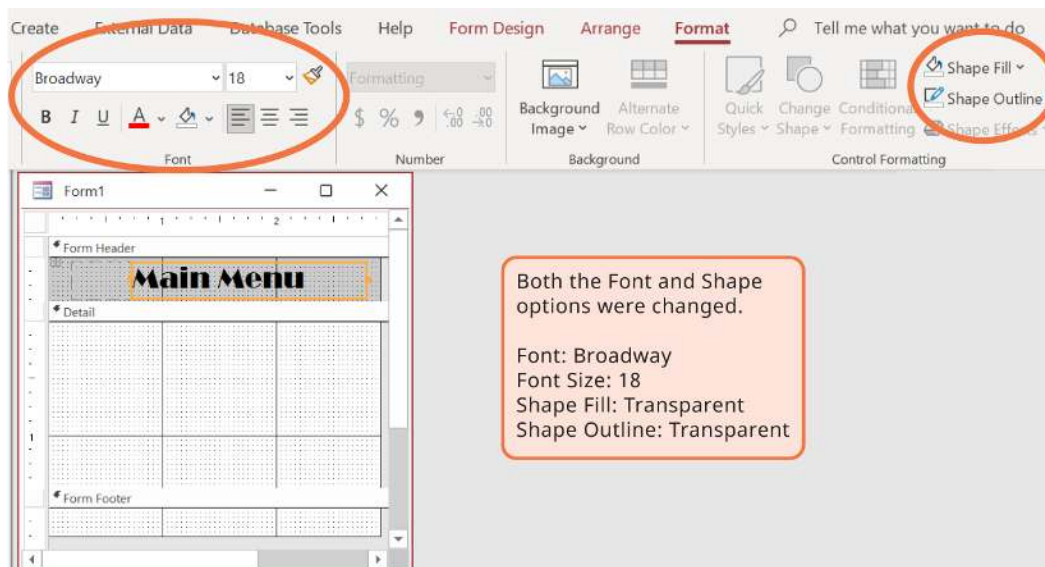


Figure 14.47: The tools to customize the header are similar to the text formatting tools in other Microsoft apps. (Used with permission from Microsoft)

When you are working in Design View, remember to return to Form View to preview your results. You may need to repeat this several times to achieve your desired formatting.

Adding Logos and Images

Logos and images are important elements to add to forms for several reasons. They can be used to reinforce the WorldCorp branding and create a consistent visual identity across all aspects of our databases, including forms. They can enhance the visual

appeal of a form, leading to a more engaging and attractive experience for the user. They can also be helpful in communicating information to users—for example, highlighting important points, guiding users through a process, or providing visual cues for navigation. Consider the rapid rise of self-ordering screens at your local fast-food restaurant. Images help a diverse group of users identify what they would like to order from the menu.

In addition, logos and images can evoke emotional responses in users, creating a more positive overall experience. For example, using images of happy customers can help create a more positive association with the database and improve user satisfaction while helping a customer in a retail location. Adding logos and images can improve the overall user experience and encourage users to use the database more frequently.

To insert a logo from Design View, select the text box that was created when inserting the title. Then, select Logo (Figure 14.48). The Windows folder will open, allowing you to select a picture file to be inserted as a logo (Figure 14.49).

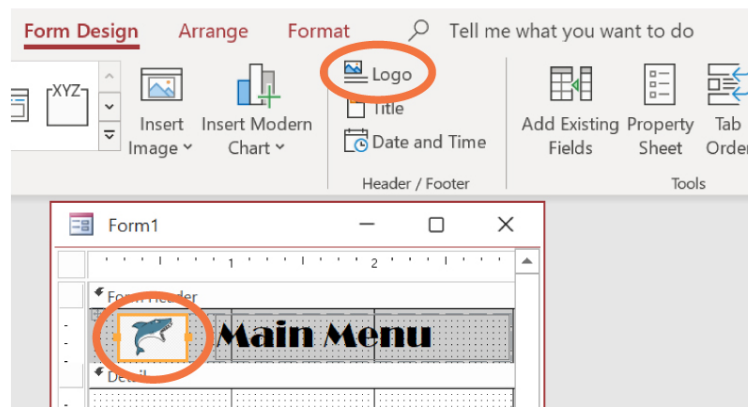


Figure 14.48: Insert your company logo in the header to customize the form. (Used with permission from Microsoft)

Figure 14.49: After you have added elements such as the company logo, you can preview what users will see using the Form View from Form Design tab. (Used with permission from Microsoft)

Formatting a form can be an enjoyable and creative process. It is satisfying to invest the time needed to craft the most visually pleasing form. Once you are satisfied with a design, consider creating a template of it that you can copy and paste as a starting point for additional forms. That way, you can avoid duplicating the effort that you have put into perfecting your form design.

This page titled [2.1.5: Customizing Forms](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.3: Customizing Forms** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.6: Multiple Table Forms

[Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Create a form that uses data from two different tables
- Create a form that contains subforms

Much like any global manufacturing corporation, we have an extensive list of assets that include company vehicles, equipment, and facilities to operate and manage. At WorldCorp, when you work with relational data, it is often convenient to be able to view multiple tables on the same form. For example, suppose you are reviewing a fleet of vehicles for one of our manufacturing facilities in Germany. Initially, you built a form that lists all the vehicles in use and enables you to add and subtract vehicles. You also built a form to create service records for each vehicle that WorldCorp owns.

When a new vehicle is added to the fleet, it receives an initial oil change and fresh fluids, and these service records are logged. You want to create a form that will make it easy to add a new vehicle and simultaneously add the initial maintenance and repair notes for that vehicle. To do this, you need to add a vehicle to a list of fleet vehicles on one table and, at the same time, add information about the maintenance/repair that has been done to the vehicle on another table. To accomplish this goal, you may want to combine data from two tables in one form or create subforms. This section presents several methods to help you create advanced forms.

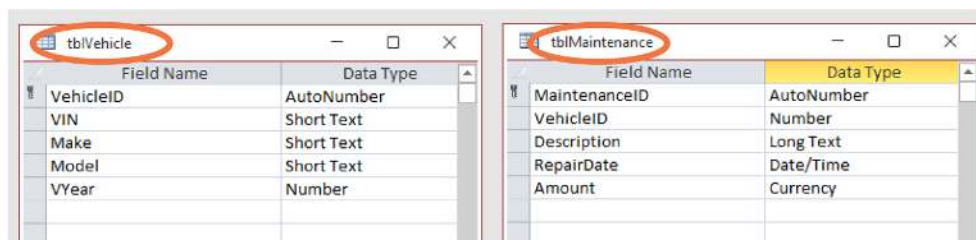
Adding Data from Two Tables in a Single Form

If you want to add data across multiple tables in one form, this could help with efficiency when managing a fleet of vehicles at WorldCorp. Primarily, it can speed data entry while maintaining the organization of tables, fields, and records.

To effectively create forms that feed into multiple tables, you should use a query as a critical building block. Recall from the chapter on Understanding and Using Databases that a query is an Access object used to view, analyze, or modify data. The query design determines which fields and records you see and their sort order. Using the example of our manufacturing facility fleet vehicle program, you can review how the query is designed, how to adjust it for user experience, and how to use the Form Wizard to build a form that can add data to two tables.

Using Queries for a Form

First, a query needs to be designed to build a relational database for the fleet vehicle program. In this example, you have two tables of information, as seen in Figure 14.17: tblVehicle and tblMaintenance.



Field Name	Data Type
VehicleID	AutoNumber
VIN	Short Text
Make	Short Text
Model	Short Text
VYear	Number

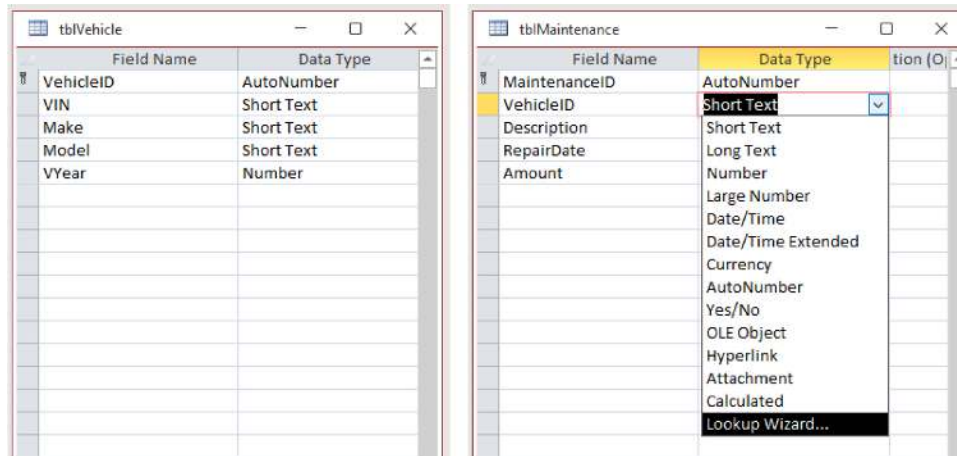
Field Name	Data Type
MaintenanceID	AutoNumber
VehicleID	Number
Description	Long Text
RepairDate	Date/Time
Amount	Currency

Figure 14.17: Forms are used in a variety of ways in organizations and using Access collects the data automatically into a table. (Used with permission from Microsoft)

The tblVehicle table represents each unique car/truck the company owns. The tblMaintenance table represents every maintenance record (oil changes, new tires, chipped glass repair, inspections, etc.) on all vehicles. In this section, you will see how to design a form that can add information to both tables simultaneously.

Remember that the ultimate goal of the new form is to add a new vehicle to the fleet *and* to include that vehicle's initial service information (oil change, safety inspection, cleaning, company decals, etc.) so that WorldCorp does not lose track or misuse valued assets. The new form will serve as a quick form for all new vehicles. For the query to be effective, a clear relationship needs to be established between the two tables.

In this example, VehicleID is the primary key of tblVehicle and is listed as a required field of tblMaintenance. Recall from the chapter Understanding and Using Databases that a primary key is a field or set of fields with values that are unique throughout a table. Values of the primary key can be used to refer to entire records because each record has a different value for the key, as shown in Figure 14.18.

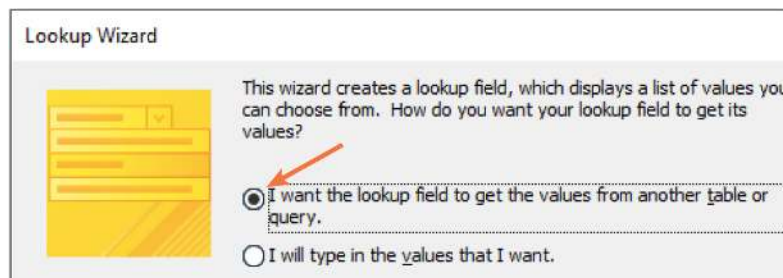


Field Name	Data Type
VehicleID	AutoNumber
VIN	Short Text
Make	Short Text
Model	Short Text
VYear	Number

Field Name	Data Type
MaintenanceID	AutoNumber
VehicleID	Short Text
Description	Short Text
RepairDate	Long Text
Amount	Number

Figure 14.18: Similar to defining the number type in Excel, when you create fields you can determine the type of information in that field. (Used with permission from Microsoft)

Access provides different data types for formatting field names. Rather than setting the VehicleID field in the tblMaintenance to short text, autonumber, or number, you can use the Lookup Wizard to build a relationship between tables. The example, shown in a series of screenshots (Figure 14.19, Figure 14.20, Figure 14.21, Figure 14.22, Figure 14.23, and Figure 14.24) will walk through the use of the Lookup Wizard steps that are used to create a relationship.



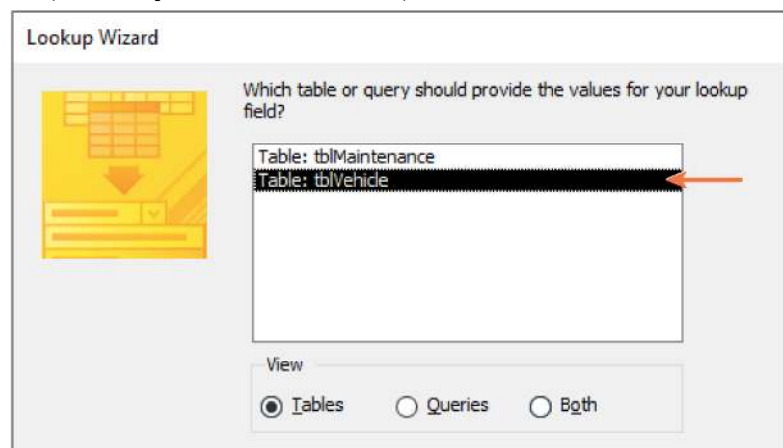
Lookup Wizard

This wizard creates a lookup field, which displays a list of values you can choose from. How do you want your lookup field to get its values?

☒ I want the lookup field to get the values from another table or query.

☐ I will type in the values that I want.

Figure 14.19: The Lookup Wizard can be used to create fields in a table in a step by step fashion either from an existing table or you can create a new table. (Used with permission from Microsoft)



Lookup Wizard

Which table or query should provide the values for your lookup field?

Table: tblMaintenance

Table: tblVehicle

View

☒ Tables ☐ Queries ☐ Both

Figure 14.20: If you choose to pull fields from an existing table in the database, you will need to identify the specific table. (Used with permission from Microsoft)

Lookup Wizard

Which fields of tblVehicle contain the values you want included in your lookup field? The fields you select become columns in your lookup field.

Available Fields:

Selected Fields:

VehicleID
VIN
Make
Model
VYear

Figure 14.21: Only fields contained in the table you select will be listed. (Used with permission from Microsoft)

Lookup Wizard

What sort order do you want for the items in your list box?

You can sort records by up to four fields, in either ascending or descending order.

1 Make Ascending

2 Model Ascending

3 VYear Ascending

4 Ascending

Figure 14.22: Sort the information using a logical field such as the specific car details. (Used with permission from Microsoft)

Lookup Wizard

How wide would you like the columns in your lookup field?

To adjust the width of a column, drag its right edge to the width you want, or double-click the right edge of the column heading to get the best fit.

☒ Hide key column (recommended)

VIN	Make	Model	VYear

Figure 14.23: Recall the primary key is the field that ties the tables together. (Used with permission from Microsoft)

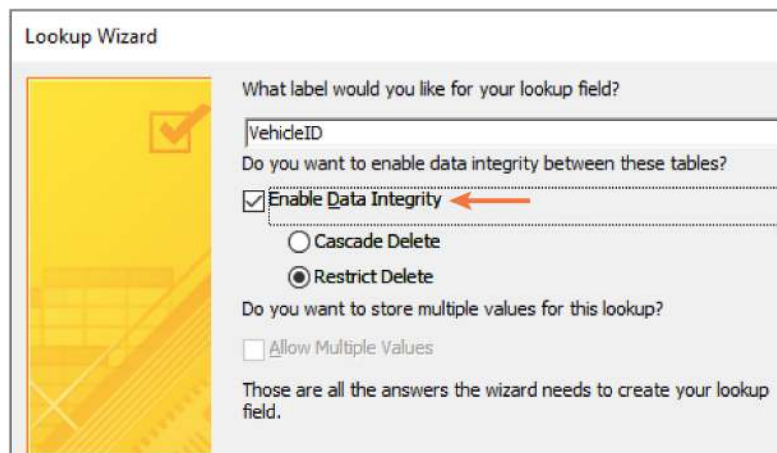
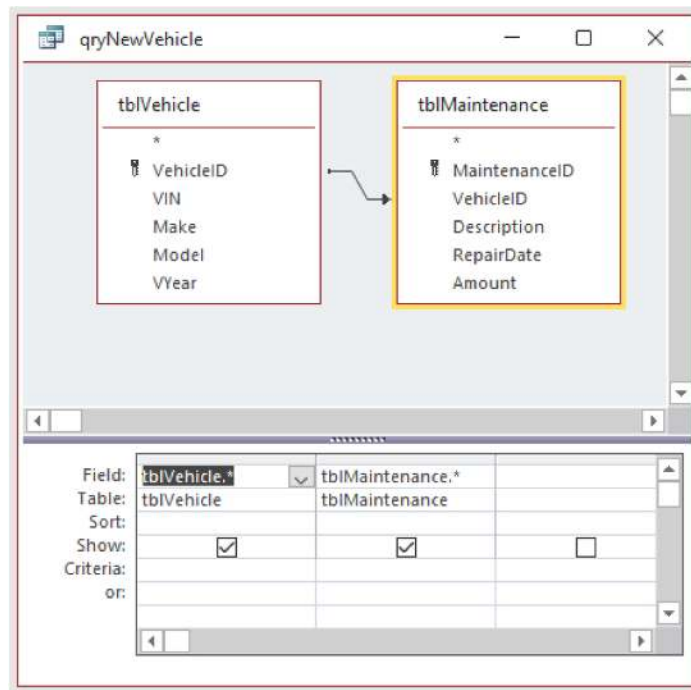
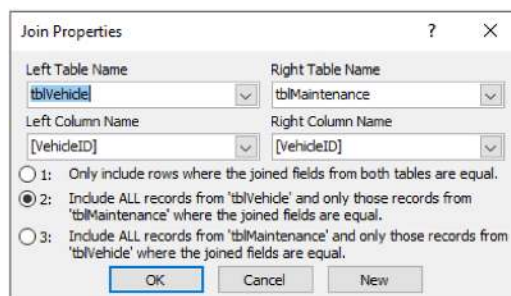


Figure 14.24: Enabling data integrity is a good idea if you are collaborating with others in the same database file. (Used with permission from Microsoft)

Now that you have created a relationship between the tables, build a query that joins the fields from the two tables together. To start, go to Query Design and bring in all the fields from both tables. In this example, the query will be saved as qryNewVehicles, as it will be the driver for the New Vehicle form to use in the vehicle fleet program (see Figure 14.25).



(a)



(b)

Figure 14.25: (a) In the query view you can see the primary keys identified in each table. (b) Define the relationship between the two tables and which records are created in the new table created from the form. (Used with permission from Microsoft)

Select qryNewVehicle from the list of All Access Objects to create the form “fromNewVehicle” (see Figure 14.26). Access will create the form automatically. This form allows the user to enter both the new vehicle that is part of the fleet and the initial check-in service performed on the vehicle.

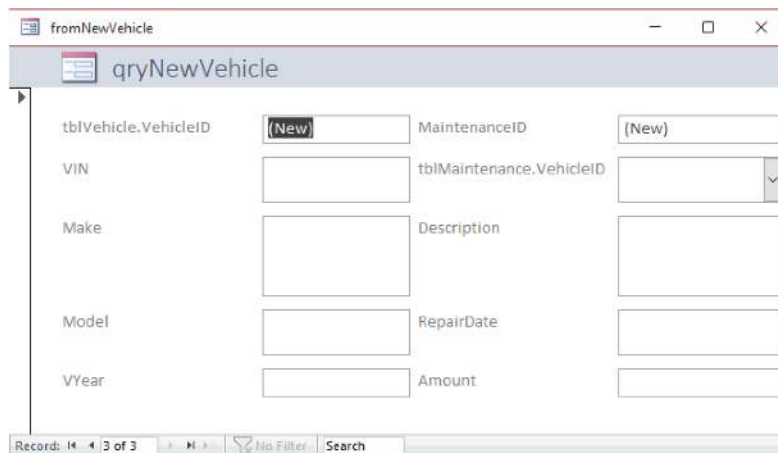


Figure 14.26: Because relationships were created previously between the two tables using the Wizard, when data is entered into the form, both tables are populated with the information. (Used with permission from Microsoft)

Using Subforms

A subform is a form that is inserted in another form (known as the main form). A form/subform combination may be referred to as a hierarchical form, a master/detail form, or a parent/child form. Subforms are generally used whenever you want to display data from multiple tables where there is a one-to-many relationship.

In a one-to-one relationship, each record in one table has no more than one related record in another table. This type of relationship is not common, as most information related in this way would be in one table, so there would be no need for a second table.

By contrast, a one-to-many relationship is the most common type of relationship. In a one-to-many relationship, a record in one table can have many matching records in another table, and the opposite is true as well.

For example, suppose you have two tables in your database, one called Employees and another called Employee Details. The Employees table contains basic information about each employee, such as their name, ID number, and job title. The Employee Details table contains additional details about each employee, such as their address, phone number, and emergency contact information. Because each employee has only one set of details, there is a one-to-one relationship between the Employees and Employee Details tables. To view and edit this information on the same form, you can use a subform.

To do this, you could create a main form based on the Employees table, with fields for Employee ID, First Name, Last Name, and other relevant information. Then, you could add a subform based on the Employee Details table, with fields for Address, Phone Number, Emergency Contact, and other relevant information. By embedding the Employee Details subform within the Employees main form, you can easily view and edit data for both the employee and their details on the same screen. You can navigate between records in the Employees table and see the corresponding details in the subform, or you can add a new employee and their details at the same time.

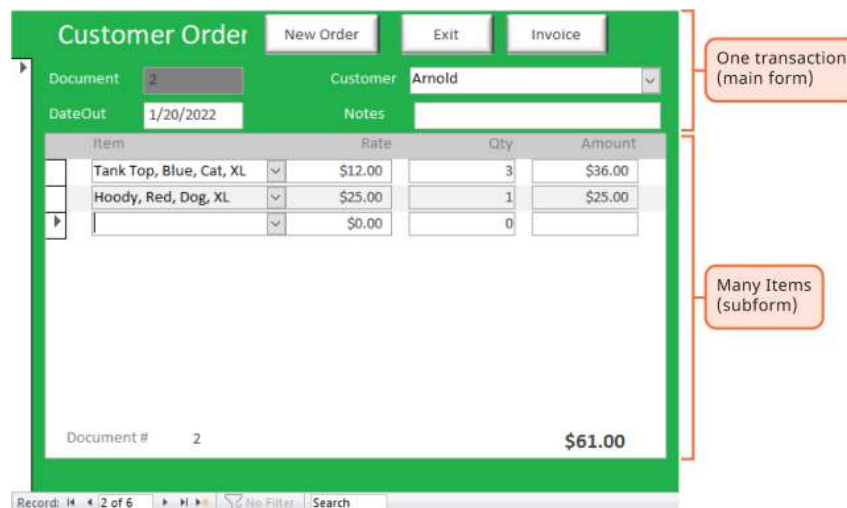
A many-to-many relationship means that for each record in one table, there can be many records in another table; likewise, for each record in the second table, there can be many in the first. Many-to-many relationships cannot be directly represented in relational database programs; they have to be built by using two or more one-to-many relationships. For example, for any business, a many-to-many relationship exists between customers and products. Customers can purchase various products and various amounts of a product, and products can be purchased by many different customers. A third table is required to create the relationship between customers (many) and products (many).

Understanding How Subforms and Relationships Work

Using subforms is an effective way to create one-to-many relationships and offers unlimited options for business use. In this type of association between two tables, the primary key value of each record in the primary table corresponds to the value in the matching field or fields of many records in the related table.

For example, if you want to display an order with all of its details, you will use a subform. The order information would display on the “main” form (one detail), and the order details would display on the subform (many details). The data in the CustomerOrder table in Figure 14.27 is the “one” side of the relationship in this example. It creates a Document number for the transaction, identifies the Customer, provides room for Notes (delivery instructions, special handling, etc.), and date-stamps the transaction. In

contrast, the data in the OrderDetails table is the “many” side of the relationship, showing an unlimited list of items purchased in a transaction.



Item	Rate	Qty	Amount
Tank Top, Blue, Cat, XL	\$12.00	3	\$36.00
Hoody, Red, Dog, XL	\$25.00	1	\$25.00
	\$0.00	0	

Figure 14.27: The main form provides the lookup values to determine what to populate in the subform, which lists the items ordered. (Used with permission from Microsoft)

The main form and subform are linked so that the subform displays only records that are related to the current record in the main form. For example, when the main form displays a Customer Order by Arnold on 1/20/22, as shown, the subform then displays items that were included on Arnold’s #2 order only. If the form and subform were unlinked, the subform would display all the items that Arnold ever ordered, not just those ordered on 1/20/22, on Document #2.

Adding a Subform through the Subform Wizard

There are two common ways to add a subform to a main form. The quickest way is to simply open a form in Design View and then drag and drop a form (from your list of completed forms) that you want to be the “many” as your subform. However, using the Form Wizard can ensure that the correct relationships are established when creating the advanced form. The following example will walk you through the use of the Form/Subform Wizard to track your time away from the WorldCorp office on the golf course.

Start by designing a query that helps ensure that relationships between the tables are created. To help you think through the query design, you might consider sketching it out on paper or in another app with boxes and arrows to show how fields are connected. Then you can take this plan and use it as your work through the queries when designing a form. Using the Form/Subform Wizard can also make designing the queries and the form easier. In this example, shown in Figure 14.28, you will design a golfing relational database that keeps track of individual scores and rounds. It may sound like a complicated task to create for a weekend activity, but with the wizard, it can take just a few minutes to have a great application in place for your out-of-office recreation.

First, create four tables to help keep track of (1) the different courses (tblCourseID), (2) the different players (tblPlayer), (3) each round played (tblRound), and (4) the score shot on each hole played per round (tblScore). Then, create a query that will bring together all four tables with all the fields from each table using the skills covered earlier in the chapter. The goal is to make tblPlayer the main form; with tblRound and tblScore listed as subforms.

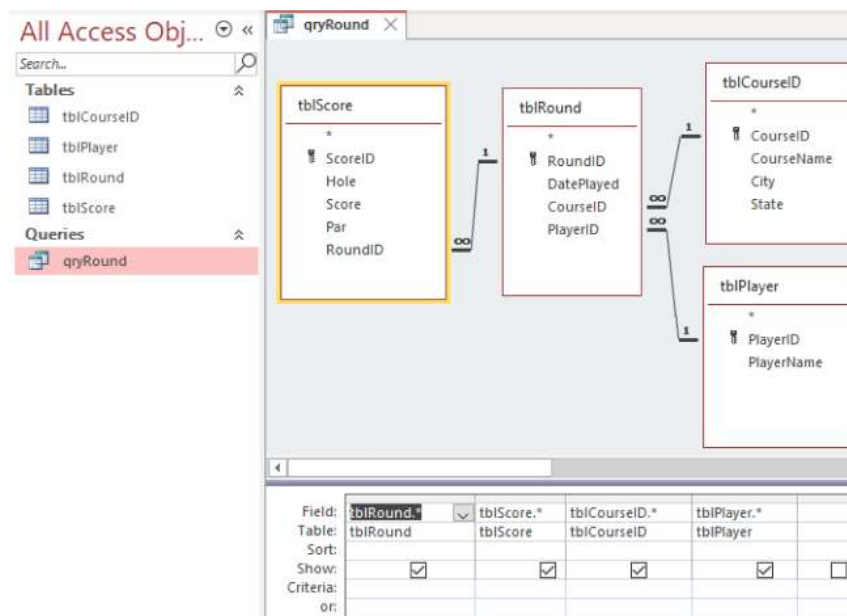


Figure 14.28: Four tables are joined in this query to create the main form and two subforms. (Used with permission from Microsoft)

Now that you have created a query, save and close it. The next step, shown in Figure 14.29, is to select the Form Wizard from the Create tab in Access.

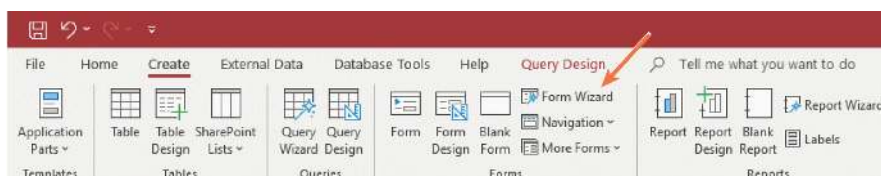


Figure 14.29: The query established the relationships between the tables and the Form Wizard found on the Create tab is used to create the form from the query. (Used with permission from Microsoft)

The following sequence of figures walks through the different options in the wizard. Once you have used the wizard a few times, you will find it easier to plan future forms and subforms for data management (see Figure 14.30, Figure 14.31, Figure 14.32, and Figure 14.33). Remember that we also used the Form Wizard in Understanding and Using Databases.

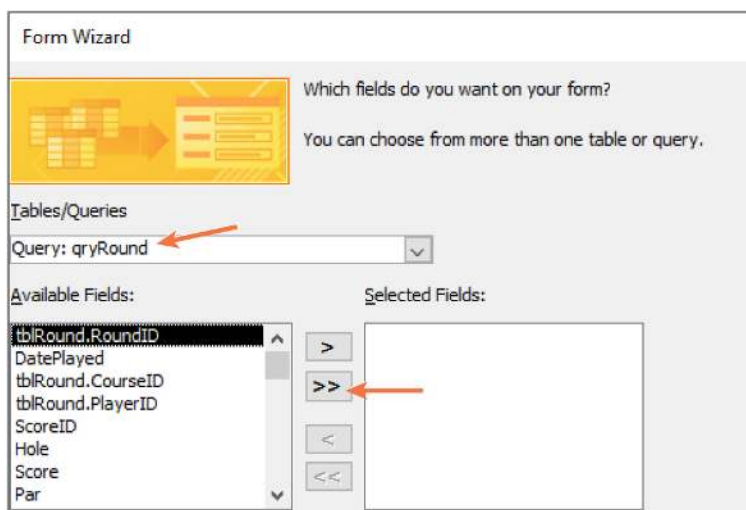


Figure 14.30: The Form Wizard begins with determining the fields that will be used in the form and from which tables those fields come from. (Used with permission from Microsoft)

Figure 14.31: Rather than creating a form with a subform, you can instead create two separate forms and link them together. (Used with permission from Microsoft)

Figure 14.32: Each subform can be displayed differently on the main form. (Used with permission from Microsoft)

Figure 14.33: Choose Modify to preview the form before using it to collect data. (Used with permission from Microsoft)

After you select Finish in the wizard, the following form is created Figure 14.34. This database, built with four tables, one query, and three forms, will provide a way to keep track of weekend golf performance for years to come.

Figure 14.34: This form was designed as a training aid to demonstrate how the Form Wizard can be used to create subforms. The subform to collect the scores is displayed in a tabular form. (Used with permission from Microsoft)

To review, the Form Wizard created three forms, as shown in the All Access Objects section: Player, Round, and Score. When you open the Player form, it will display all three forms. In this example, Larry is the selected player on the main form—PlayerID #1. The subforms will display information related to only Larry. He is currently playing his first round at Twin Oaks on 5/24/22. We know this because the information displayed within the Round subform is clearly highlighted in blue. Round is set to be a subform of Player. Additionally, Larry has just entered his score on hole #3 of his first round. His score was a 3, and he also marked that the hole was listed as a par 3; Larry shot par!

Score is a subform of Round, so only scores for the selected round will be stored. As Larry continues his round, he can simply add additional rows of data from each hole he plays on the Score subform. These scores will both be saved under RoundID #1 and kept as Larry's unique scores. If Larry is playing against another player, you would navigate to the correct player on the main form and enter their round/score as well.

Larry can use this simple yet powerful form to keep track of unlimited rounds of golf. In the office, he might apply these steps to developing a way to keep track of a variety of business tasks or projects. For example, each of his employees could have an assigned project with numerous unique details concerning each project. The employee would be the main form, while the project names and project details could be the subforms.

Adjusting a Subform in Design View

Subforms are difficult to change and update within Design View of the main form that they are sitting on. A much easier way to alter a subform is to open that form separately (first making sure the main form has been closed). Using the previous golf example, if you want to add details to the Round subform (for example, add a place for notes on weather or course conditions), it is best to open up the Round form by itself and make the changes. Once the changes have been made, you can save and close the form, then reopen the main form (in this example, Player) to see whether you need to adjust the size. After making any changes, always make sure the subform displays the new information within the main form.

This page titled [2.1.6: Multiple Table Forms](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

- **14.2: Multiple Table Forms** by OpenStax is licensed CC BY 4.0. Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.7: Creating Reports in Microsoft Access

 [Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Understand why reports are useful
- Create and edit a report using the Report Wizard
- Customize a report by adding images and text

Databases are useful only if they enable us to transform data into information that will be ready for use when it is needed. Although queries are helpful in extracting the data that we need and even transforming it into actionable information, most decision-makers will need more refinement, formatting, and preparation before the data is ready for use.

Recall that a report is a formatted display of information designed for a specific use case or project. Reports do more than summarize information; they also create the foundation for making sound decisions. When we combine queries with formatting, we create reports that we can use in our work.

Using Reports in a Database

Reports are a cornerstone for business decisions. We can generate reports within a database for a specific request, and we can also set up reports to provide summaries and information that we will need on an ongoing basis. Access enables us to set up reports (e.g., a financial, sales, or summary report) that we can then access as necessary for decision-making. Companies routinely run reports on key performance indicators (KPIs) that show an organization's effectiveness, efficiency, quality, economics, project performance, resource use, and personnel and workforce performance. Companies also routinely look at revenue, profit, and customer and employee satisfaction. Because tables filled with data do not adequately describe KPIs, businesses rely on reports as part of their evidence-based decision-making.

Tables are extremely efficient ways of storing data, and using a relational database with connections across tables is an effective way of building information storage. Unfortunately, however, tables do not display information in ways that people easily understand. We can manage this by creating reports—formatted, attractive displays of information. Reports use combinations of labels, values, and even graphics to present the results of a query in a way that supports action. Once designed, a report can easily be reused or copied and revised. Like other objects such as queries and forms, reports will show the most up-to-date information each time the database changes. Among the many reasons to generate reports are the need to create summary reports, to save snapshots of data, and to provide detail about specific customers or products.

Consider this example. Anna, an analyst in the sales and marketing group at WorldCorp, has been asked to get data ready for a meeting with sales representatives. The meeting will focus on products that are on hold, and the representatives will need to identify wholesale companies that may be interested in these products. Anna begins by determining what the sales representatives will need for the meeting and the level of detail they will want to have available for decision-making. She then looks at the available data and decides how to create the report. Anna's final report includes two sets of information—the list of on-hold products and the list of wholesalers who have shown an interest in those products. Therefore, she creates two versions of the report:

- List of on-hold products with a list of interested wholesalers for each product
- List of wholesalers with a list of on-hold products that may be of interest to each wholesaler

Anna sends the final report to the person in charge of the meeting for review and comment.

Tips for Effective Reports

The way you design a report will depend on the data it will include, the audience that will use the report for decision-making, and the way the data will be used. Reports can be as brief as a few lines of summary text, or they can be highly detailed spreadsheets or within datasheets with many rows of records. The key to building a useful report is communication—your communication with the audience that needs the report, so you understand them, and the communication of the information to the audience so they can use it.

Effective reports should utilize the headers and footers of a report and its pages to aid the reader when navigating the report. Headers and footers should contain useful information such as the report title, author, and page numbers. This makes it easier for a report reader to quickly find the information they are looking for across multiple reports.

Link to Learning

Data is only good to the extent that it is both accurate and actionable. Raw data without any use is just clutter. With this in mind, watch this [video on how to make data actionable](#) and consider how reports can summarize large amounts of data and make that data actionable.

The goal of a report is to eliminate as many individual lines of information as possible and to simplify information through summary wherever you can. In general, consider the following when preparing effective reports:

- Minimize line-by-line data
- Use labels to explain the data that is shown on the report
- Use formatting to designate titles and other labels
- Incorporate white space to ensure readability
- Test with possible audience members to be sure the report makes sense to the final audience
- Focus on answering specific business questions generated by the person or people who request the report

Organizing a Report

A report can be flat, with one record per line, or you can use grouping to further organize the data. When designing a report, think about organizing your data as follows:

- Sorting to order your data by an important field (e.g., Name), when you are discussing customers or sales region
- Grouping records based on a field, such as active versus inactive products, regions that met specific targets versus those that are lagging in meeting goals, or geographical area
- Totaling groups of data, such as sales figures, to show both the detailed figures (e.g., an individual salesperson's sales figures) and the figures for the overall region

Access will allow you to quickly add and preview sorting, groups, and totals to ensure the report communicates what you need it to for the audience you are working with.

Creating and Editing Reports with the Report Wizard

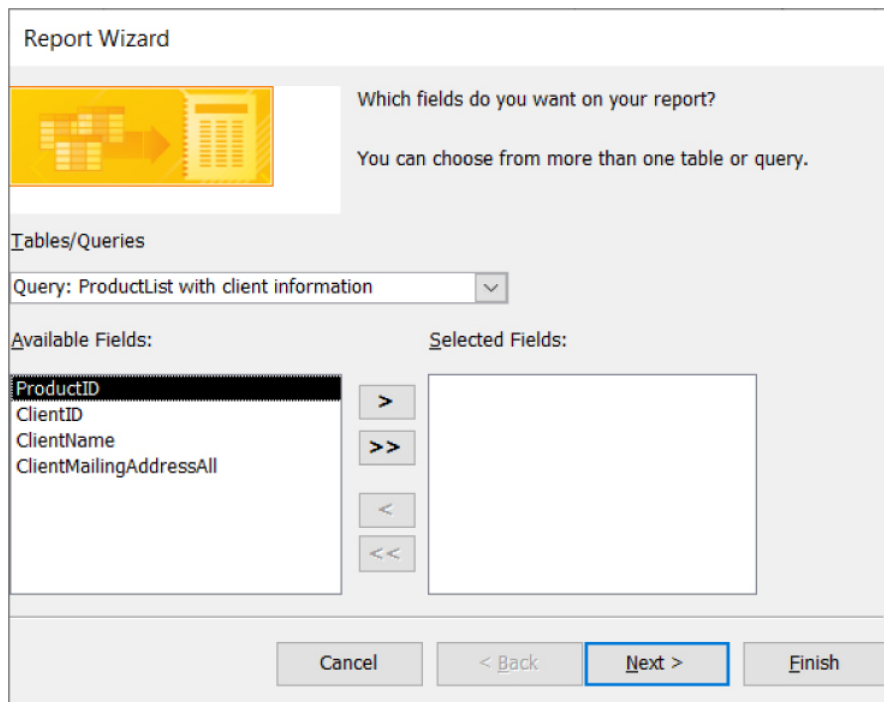
WorldCorp is interested in creating a summary of their recent sales data for their latest product line. The report should include useful information such as which products have the highest total sales, which products have the highest net profit, and which customers are purchasing the most products. WorldCorp could create this report from a blank canvas, but it would be much easier to generate a report using the Report Wizard. This is a small program that will walk you through the steps of setting up a report, which you can then modify to meet your needs.

Using the Report Wizard is usually an efficient and easy way to start a report. Most reports will need some form of customization, but the Wizard will provide a good starting point and will save steps by adding fields that are bound to the database.

Creating a Report

To start a report with the Report Wizard, click on the Create menu in the ribbon and then choose the Report Wizard command from the reports section. The Wizard will walk you through each question before creating your report.

Step 1: In step 1 (Figure 13.40), you can set the table(s) and/or queries you want to work on within the report. This gives you flexibility, as you can use data from more than one table or query in your work. (Hint: Consider setting up a query for your report before starting the report. This will enable you to check the data before setting it up for display.) Once you have chosen and added tables, queries, and fields to the report (use the >> button to move fields to the selection box), click Next to move to the next screen in the Wizard.



Report Wizard

Which fields do you want on your report?

You can choose from more than one table or query.

Tables/Queries

Query: ProductList with client information

Available Fields:

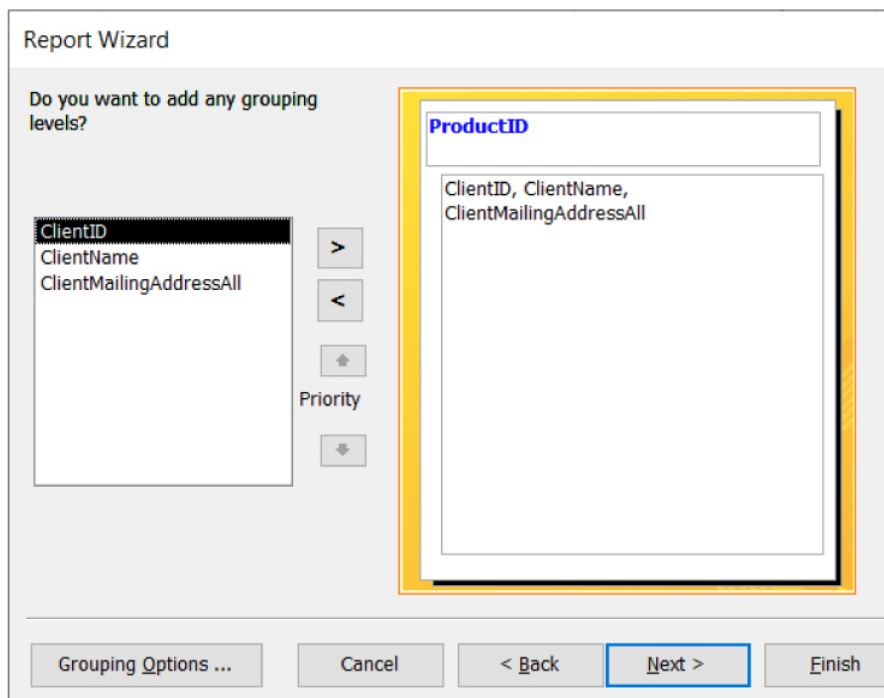
- ProductID
- ClientID
- ClientName
- ClientMailingAddressAll

Selected Fields:

Navigation: Cancel, < Back, Next >, Finish

Figure 13.40: The Report Wizard allows you to select fields to include in the report. (Used with permission from Microsoft)

Step 2: In step 2 (Figure 13.41), you can choose grouping for your report. (Skip this step if grouping does not make sense for your report.) A group is a heading that organizes data with something in common. For example, you might group data based on the region a customer is assigned to. When used as a group, all customers belonging to a specific region will appear together. Once you set up grouping, you can click Next to continue.



Report Wizard

Do you want to add any grouping levels?

Available Fields:

- ClientID
- ClientName
- ClientMailingAddressAll

Grouping Options:

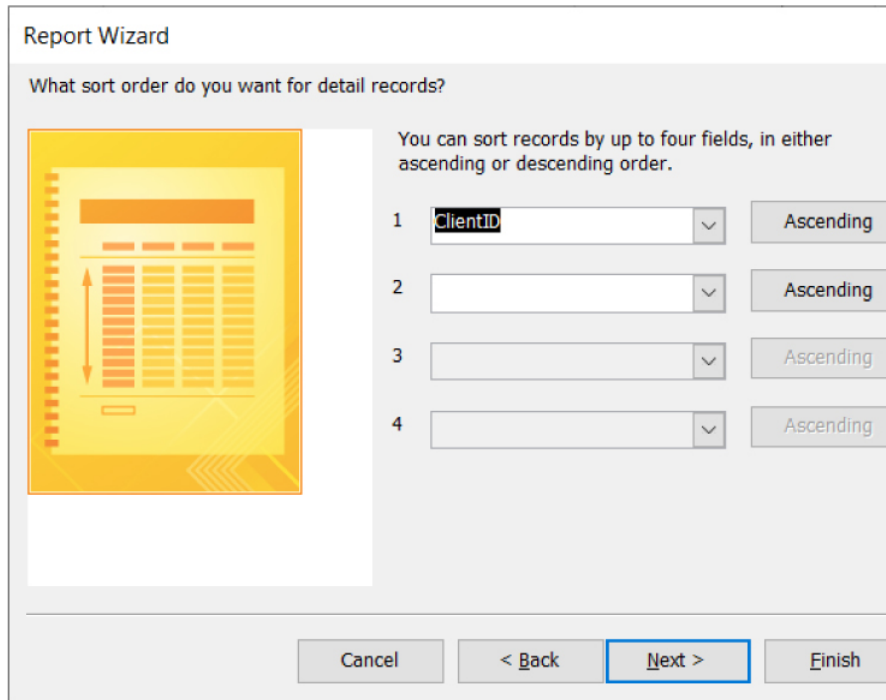
- ProductID
- ClientID, ClientName, ClientMailingAddressAll

Navigation: Grouping Options ..., Cancel, < Back, Next >, Finish

Figure 13.41: In this example, grouping by product ID has been used, creating a report that first lists each product, with the client info of every client who has purchased that product listed under the ProductID. (Used with permission from Microsoft)

Step 3: Step 3 of the Report Wizard asks how you want to sort your records, as Figure 13.42 shows. Access allows you to sort by up to four fields in the designated order. For example, to create a report that fits the needs of your audience, you may want to sort a

set of records by sales region in alphabetical order and then sort further by product ID. Once you set up sorting, you can move to the next step in the Wizard.



Report Wizard

What sort order do you want for detail records?

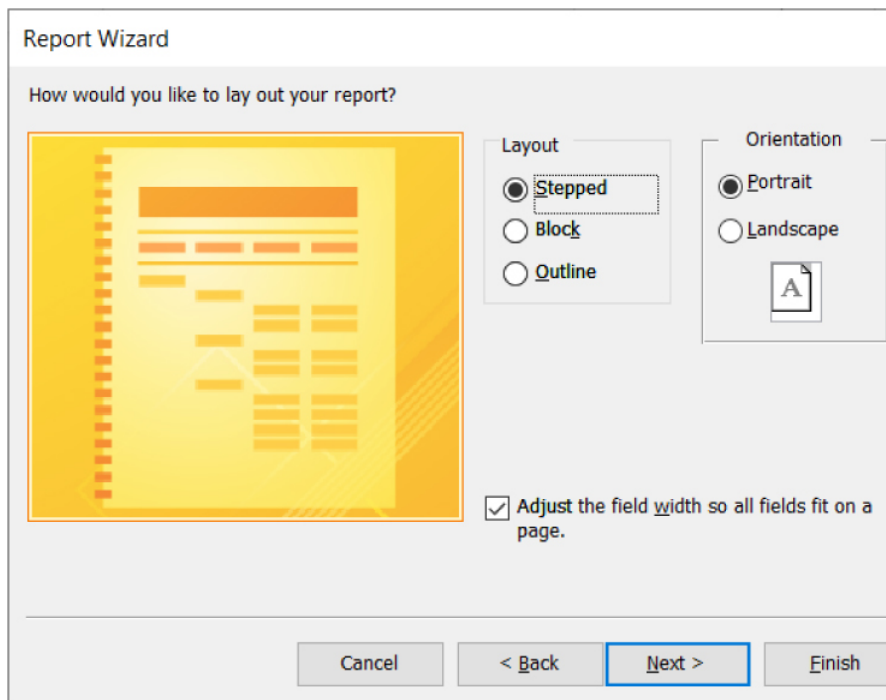
You can sort records by up to four fields, in either ascending or descending order.

1	ClientID	Ascending
2		Ascending
3		Ascending
4		Ascending

Cancel < Back **Next >** Finish

Figure 13.42: Using the ClientID field with the Report Wizard allows you to sort records in ascending order, from smallest to largest or from A to Z. (Used with permission from Microsoft)

Step 4: After you have set the sort order, the Wizard will ask you to choose the layout for your report (Figure 13.43). You can also set the orientation and ask Access to adjust all fields to make sure they fit on a page.



Report Wizard

How would you like to lay out your report?

Layout

- ☒ Stepped
- ☐ Block
- ☐ Outline

Orientation

- ☒ Portrait
- ☐ Landscape

☒ Adjust the field width so all fields fit on a page.

Cancel < Back **Next >** Finish

Figure 13.43: Setting the report layout in the Report Wizard allows for limited customization of the final report. (Used with permission from Microsoft)

Step 5: The final step in the Wizard allows you to set a unique title for your report and then choose whether you want to provide the results or go into design to modify the report (Figure 13.44). Usually, viewing the report will be useful, as it will show you what

you want to change. When you are done, click Finish to continue the work.

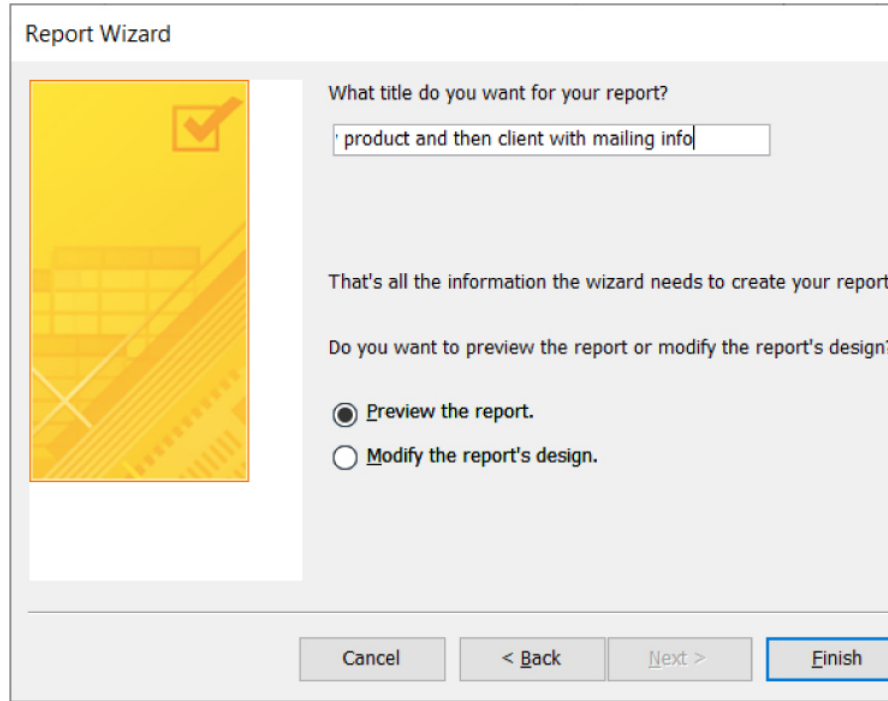
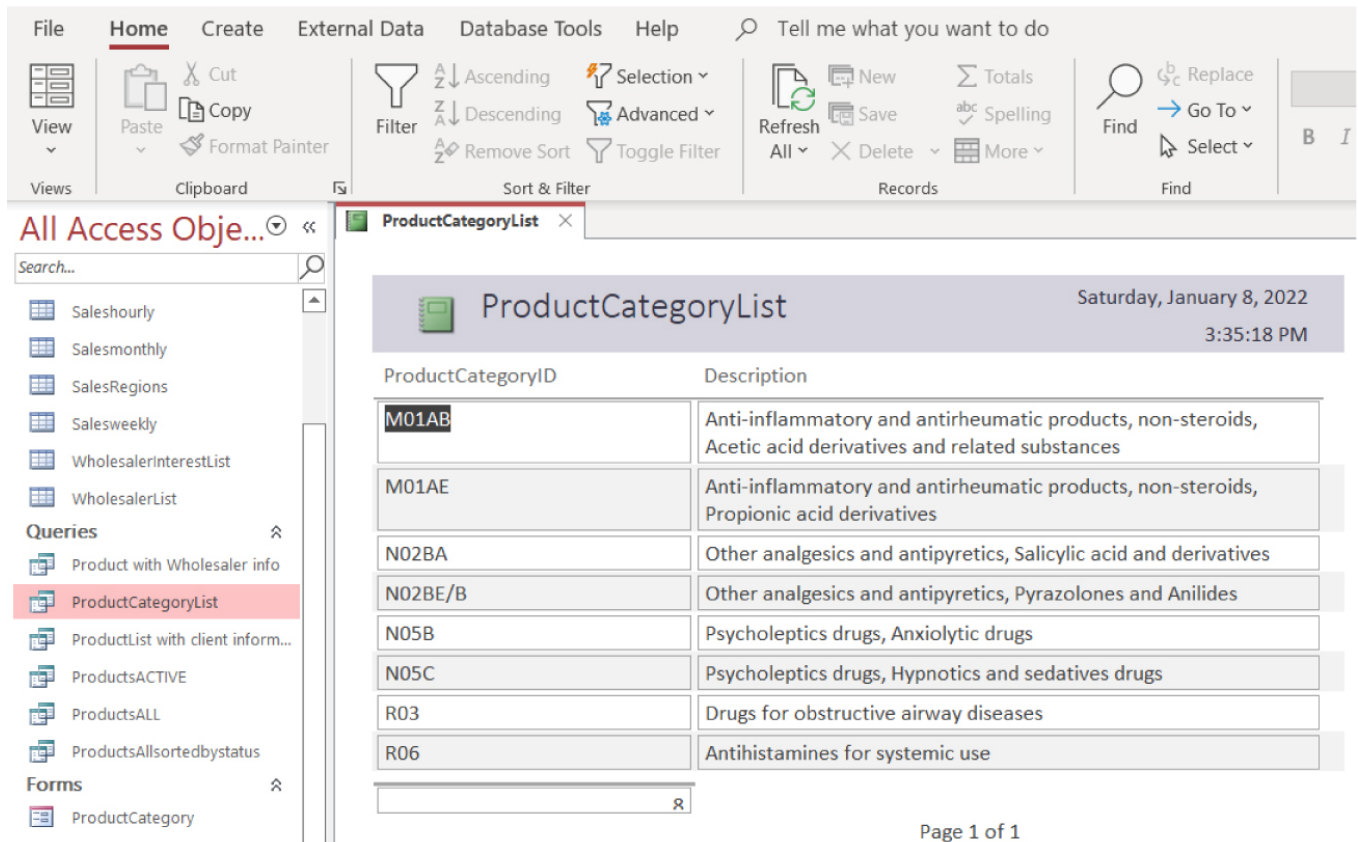
The image shows a 'Report Wizard' dialog box. On the left is a yellow square with a checkmark icon. The main area contains the text 'What title do you want for your report?' followed by a text box containing 'product and then client with mailing info'. Below this is the text 'That's all the information the wizard needs to create your report.' and 'Do you want to preview the report or modify the report's design?'. There are two radio buttons: 'Preview the report.' (selected) and 'Modify the report's design.'. At the bottom are four buttons: 'Cancel', '< Back', 'Next >', and 'Finish' (highlighted with a blue border).

Figure 13.44: Setting a name for the report is the final step before Report Wizard generates a preliminary report. (Used with permission from Microsoft)

Reviewing a Report

When you double-click on a report in the navigation area, the report will open with current data displayed. Let's look at a simple report in Figure 13.45: a listing of all product categories.

This report, named ProductCategoryList, was created quickly by clicking on Create and then Report.



The screenshot shows the Microsoft Access interface. The top ribbon includes tabs for File, Home, Create, External Data, Database Tools, and Help. The Home tab is active, showing options for View, Paste, Copy, Format Painter, Filter, Sort & Filter, Records, and Find. The left pane shows the 'All Access Objects' list, with 'ProductCategoryList' selected under the 'Queries' section. The main window displays the 'ProductCategoryList' report, which is a table with two columns: 'ProductCategoryID' and 'Description'. The report is dated Saturday, January 8, 2022, at 3:35:18 PM. The report is displayed in a window titled 'All Access Objects'.

ProductCategoryID	Description
M01AB	Anti-inflammatory and antirheumatic products, non-steroids, Acetic acid derivatives and related substances
M01AE	Anti-inflammatory and antirheumatic products, non-steroids, Propionic acid derivatives
N02BA	Other analgesics and antipyretics, Salicylic acid and derivatives
N02BE/B	Other analgesics and antipyretics, Pyrazolones and Anilides
N05B	Psycholeptics drugs, Anxiolytic drugs
N05C	Psycholeptics drugs, Hypnotics and sedatives drugs
R03	Drugs for obstructive airway diseases
R06	Antihistamines for systemic use

Page 1 of 1

Figure 13.45: The report generated by the Report Wizard is well organized and easy to read. (Used with permission from Microsoft)

The title at the top of the report shows the date the report was run, followed by two columns that list the product category ID and the description. This report is set up by Access and has not been modified. In Design View (Figure 13.46), this report appears differently, with the design and grid available for modification. This report is usable as is but taking a few minutes to customize it will make it more useful for your audience.

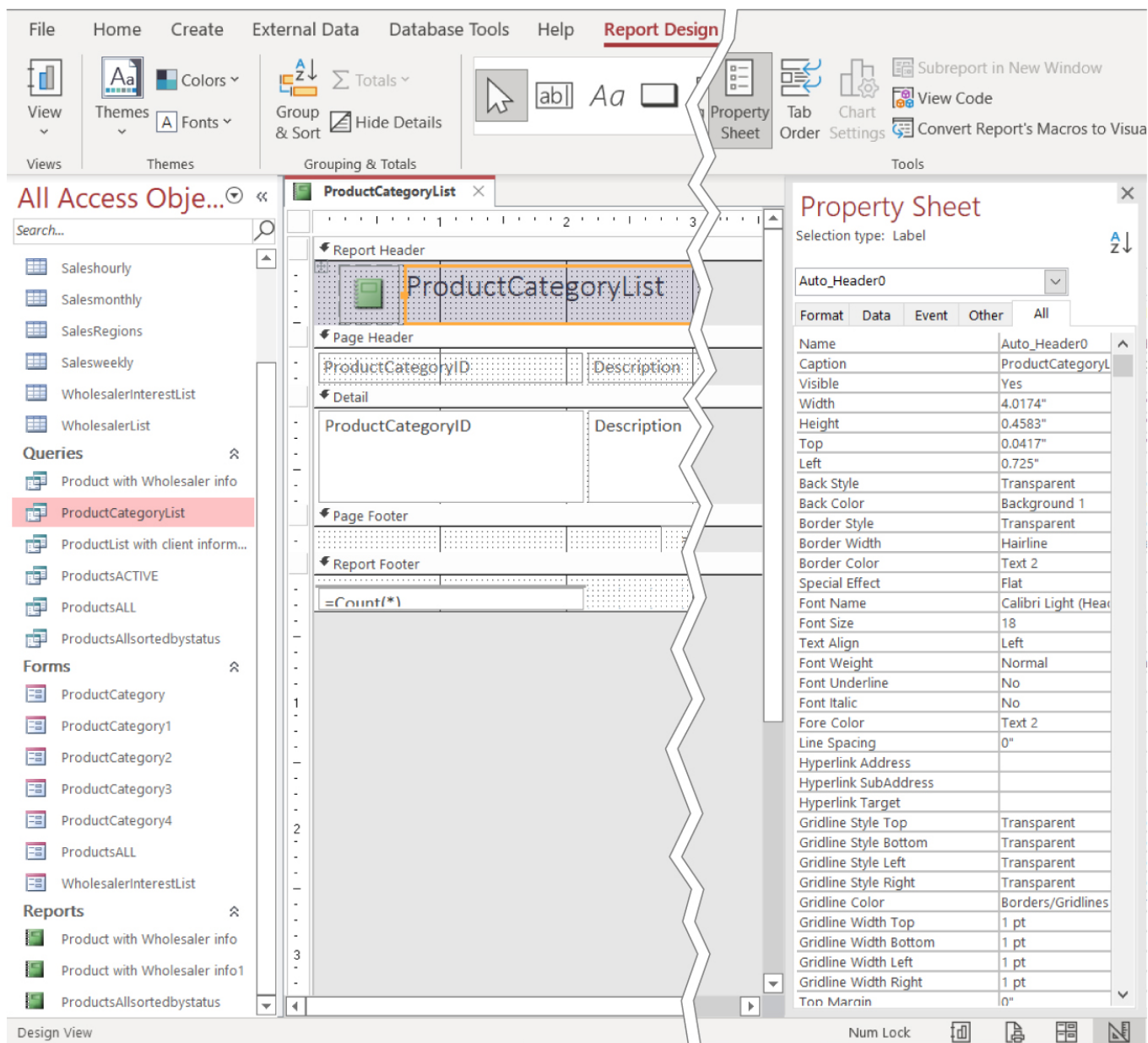


Figure 13.46: Access allows you to fully customize the layout and content of a report depending on your specific needs. (Used with permission from Microsoft)

In Design View, the report header appears on the first page and includes the title and dates of the report. This is followed by the page header which appears on each page and includes information useful to the report user, such as page number or report author. The report detail is the area where data from the database is displayed for each record. Each record will show the product category ID and description. The report has a page footer, which displays useful information, such as the page number, and appears on each page. The report footer shares the number of pages in the entire report. As you build reports, you will likely switch back and forth between the Report View and Design View to change elements such as spacing and formatting so that your report is useful for your audience.

Link to Learning

Personal and company data needs to be safeguarded, and reports that summarize data related to projects and goals also need to be protected. Therefore, one of the responsibilities of anyone dealing with data and reports is data security. To this end, governments have put laws and procedures in place to guide how organizations manage data and reporting. This [article on data privacy laws](https://workforce.libretexts.org/@go/page/64188) provides an overview of some of the laws in this area.

Customizing Reports

Although Access does a nice job of starting a report, you will likely want to modify reports before using them. Following are some of the changes you may want to make:

- Changing the size of fields (e.g., shortening them)
- Formatting the text of the report (e.g., by adding boldface)
- Shifting text boxes to lay the screen out more effectively

Page Layout View

Page layout view gives you a simple way to make changes to the overall appearance of a report. Changes made to a field in page layout view will be applied to all fields of the same type (e.g., if you boldface a field, all fields of the same type will also be bold). Page layout view displays live data while you make your changes, which makes decision-making easier.

To enter Layout View, click on the View command on the Home tab while you have a report open on the screen. You can preview the report by changing to the report view. When you are finished working, save and close your report.

Design View

The Design View screen has more options for customizing your report. To switch to Design View, use the View command on the ribbon.

Design View gives you a layout grid for your form and the ability to move each label and field independently, change the formatting of the text, and even change the attributes for each element on the report.

A common change in Design View is modifying the size of the field on the screen. For example, you may find that Access cuts off some of the label or content in a field. You can modify the size of the field or label by clicking on the element and then using the selection handles on the left and right sides to stretch or shrink the element.

You can also move elements manually by clicking and dragging them or by clicking on multiple items and using the commands on the Arrange tab on the ribbon. These commands allow you to move items up or down, arrange them together, or control how the elements are arranged on the report itself.

Adding an Image or Custom Text

You may want to include a logo or even a disclaimer as an image in a report. These images can add information or (as in the example of a logo) provide identification. Images typically appear at the top or bottom of a report. Remember that if you place an image in the detail section of the report, it will repeat for every line. (This is useful when you want to have each line start with a symbol or logo.)

To add an image in Design View, click on the Form Layout Design tab and choose the image that you want to add. You can also add any image to the page by clicking Insert Image (see Figure 13.47) and then locating the image file. Again, be aware that when images are placed in the detail section of the form, they will repeat for every line. If you would like the image to appear only at the top or bottom of a page, place the image in the page header or footer.

You may want to add elements that are not strictly images but display on the report (e.g., titles, dates, and times). Note that these items will be displayed on the report but will not be stored in the database; they will change each time you open the form.

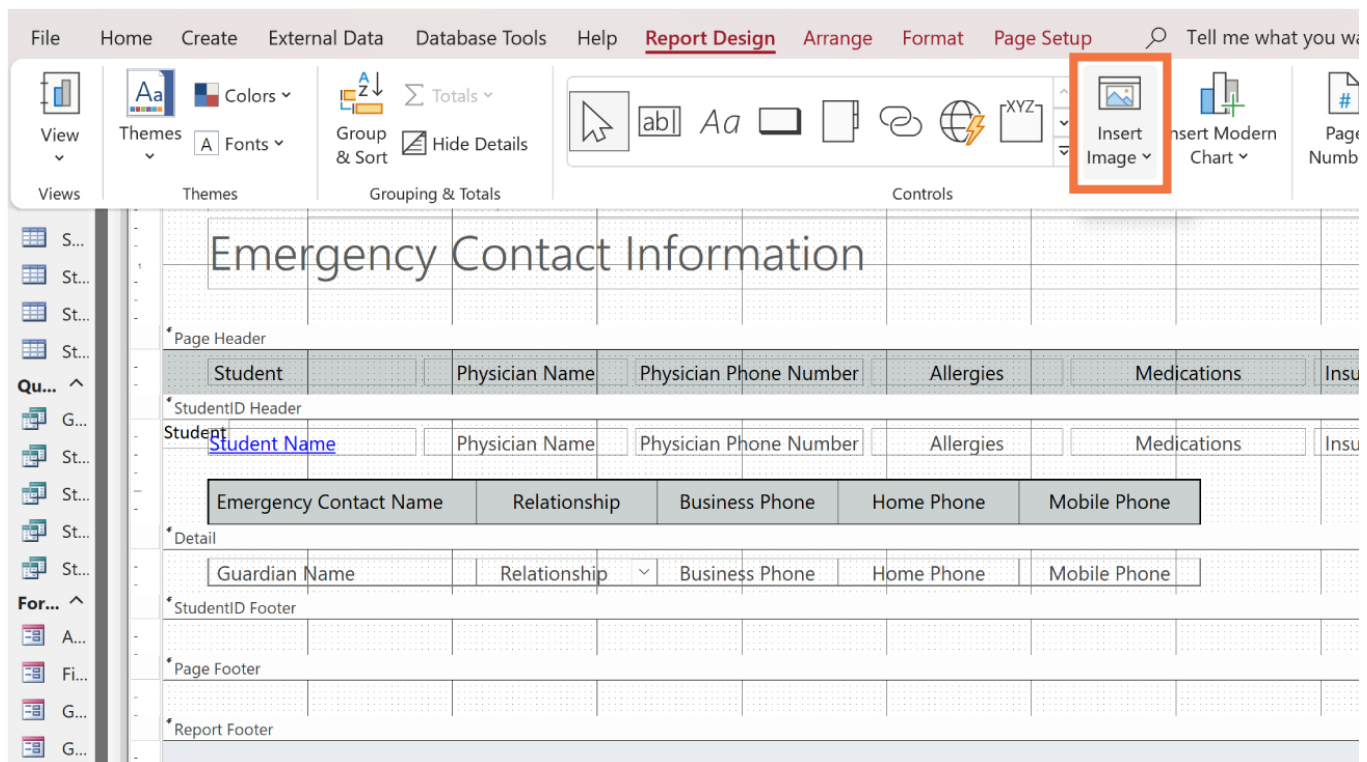


Figure 13.47: You can export selected Access objects to an Excel worksheet. (Used with permission from Microsoft)

When you create and modify a report, Access will prompt you to name and save it. Each time you modify an existing report and close it, Access should prompt you to save it. You can save manually by right-clicking on the name of the tab in the work area and choosing the Save command.

To close a report that is open in the work area, click on the X next to the tab name.

Report Views

Utilizing Access to analyze data results in lots of information. This Information needs to be shared – often will people who do not, or should not, have rights to utilize the Access database. Access **reports** organize and present data in a reader-friendly, visually appealing format by using data from any queries or tables in the database. The tables or queries that provide the underlying data are also known as the report's **record source**. A report is a database object that comes in handy when you want to present the information in your database for any of the following uses:

- Display or distribute a summary of data.
- Archive historical information.
- Provide details about individual records.
- Create labels.

There are four different report views in Access. Two are essentially print views and the other two are report building views. As with all other Access objects, you can toggle the view using the buttons at the bottom right of the screen, or by using the View button on the Ribbon. The four views include:

- **Report view** – shows how a report would appear with dummy data using a continuous page layout.
- **Print Preview** – allows users to see what the report will look like when printed. This view can be used to change the page layout, including the report orientation.
- **Layout view** – allows users to edit the report design while it contains live data. Considered the most intuitive view to use for report modification, this view can be used for nearly all potential changes to a report in Access.
- **Design view** – allows users to view and change any aspect of the report structure, but not the data. This view displays any expressions that are used to generate content, and displays the header and footer bands for the report, page, and groups.. This is the preferred view for adding controls to the report, such as labels, images, lines and rectangles.

The design of a report is divided into sections that can be modified in the Design view. Reports created by using the Report button are plain and simple, and have several sections, as detailed in the table to the right. More complex reports created using the Report Wizard can also have multiple **group header** and **group footer** sections to print grouping details and summary information.

Report Section	Purpose
Report Header	Appears at the top of the first page and displays the report title. Displays only once in a report.
Page Header	Appears at the top of every page and displays the headings (field labels) for each column.
Detail Section	Appears between the page header and page footer and displays the records from the table or query. Can run across many pages dependent on the amount of data.
Page Footer	Appears at the bottom of every page and displays the page number and total number of pages.
Report Footer	This section is optional. Appears on the last page of the report and displays summary information such as grand totals. Displays only once in a report.

Report Sections

Reports can be designed and formatted in a similar way to forms. When you create a report, four new tabs appear in the Report Layout Tools section of the Ribbon specifically for working with reports.

- **Design:** Group and sort elements in the report, add text and links, insert page numbers, and modify the sheet's properties.
- **Arrange:** Adjust the table to be stacked, tabular, etc.; move rows and columns up and down or left and right; merge and split columns and rows; control the margins and bring elements to the "front" or "back" in a layering format.
- **Format:** Includes regular word processor formatting tools like bold, italic, underline, text and background color, number and date formatting, conditional formatting, etc.
- **Page Setup:** Allows adjusting of the page's overall size and toggling between landscape and portrait.

Another popular approach to creating reports is to use the Access Report Wizard. The **Report Wizard** provides more flexibility than by simply using the Report button. The Report Wizard guides users through a series of questions and then generates a report based on these answers. Users can choose the tables and fields, group the data, sort the data, summarize the data, choose a layout and orientation, apply a style, and title the report. Grouping data puts all of the values in a field into a group based on the field's value. By using the Report Wizard, users can create up to four sort levels. If the data has been grouped, the option exists to summarize the data by displaying the sum, average, and minimum or maximum value for each numeric field. Choose to have the report display just the summary data or each detail line and the summary data. Note: Users can also add grouping, sorting, or totals by using the Group, Sort, and Total pane while the report is open in Layout view or Design view.

To print a report, right-click the report in the Navigation Pane and choose Print Preview from the shortcut menu. Print Preview command options include:

- Print the report
- Adjust page size, margins or orientation
- Zoom in or out, or view multiple pages at a time
- Refresh the data on the report
- Export the report to another file format.

To save paper, or to send a print job to a professional printer, or send the report via e-mail, you can create a .pdf or xps format of your report: On the **Print Preview** tab, in the **Data** group, click **PDF or XPS**. Reports can be saved as an object in the Navigation Pane, and at later date, updated and re-printed. Unlike Access queries, the page settings are maintained in Access reports when saved.

This page titled [2.1.7: Creating Reports in Microsoft Access](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **13.5: Creating Reports in Microsoft Access** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **7.4: Access Reports** by Marcus Lacher is licensed [CC BY-NC-SA 4.0](#). Original source: <https://mlpp.pressbooks.pub/businesscomputers2019>.

2.1.8: Customizing Reports

 [Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Create a customized report
- Modify what appears on a report
- Change the appearance of a report

Reports offer a way to view, format, and summarize the information in your Access database. For example, at WorldCorp, you can create an invoice for all your sales, or a summary of total sales across different regions and time periods depending on which corporate business unit on WorldCorp you are surveying. In the chapter on Understanding and Using Databases, you learned about how to create and modify reports using the Report Wizard. In this chapter, we will go deeper into creating and modifying reports.

Building a Custom Report

The Report Wizard does provide a quick and efficient way to create reports. At times, however, you will need to make a report from scratch. Understanding the section types enables you to use them to their fullest potential.

Often, queries are used to combine tables and define fields that will be included in a report at WorldCorp. As with creating a form, queries become the primary building block for developing a desired report that meets the current demands of the ever-changing WorldCorp objectives.

A report is divided into sections that can be seen in Design View. To create professional-looking reports, you will need to understand how each section works. For example, the section in which you choose to place a calculated control determines how Access calculates the results. What you want on the cover page of the report may be very different than what you want at the top of the pages that follow.

In addition to the report header, page header, report footer, and page footer, which we covered in Understanding and Using Databases, there are other sections in the report that you should know:

- Group Header: This section appears at the beginning of each group in the report and is used to display summary information about each group.
- Detail: This section is where the actual data from the report is displayed. It appears for each record in the report.
- Group Footer: This section appears at the end of each group and is used to display summary information about the group.

Using these sections allows you to create a well-organized, easy-to-read report that contains all the necessary information.

Modifying a Report

There are several tools and views that you can use to create, format, and customize reports. The different views allow you to see and preview your report in different ways, each of which may be useful to you at times. We reviewed these different views and some of these tools in Understanding and Using Databases, but here, we will go into more depth.

Layout View

Layout View is a great way to see the print preview grid lines, with the ability to change the size and placement of all labels, text boxes, and design elements. Right-clicking and selecting Layout View shows how much of the report will fit on a page. In Layout View, each control on the report displays real data, making this a useful view for setting the size of controls or performing tasks that affect the report's appearance and usability, as Figure 14.50 shows.

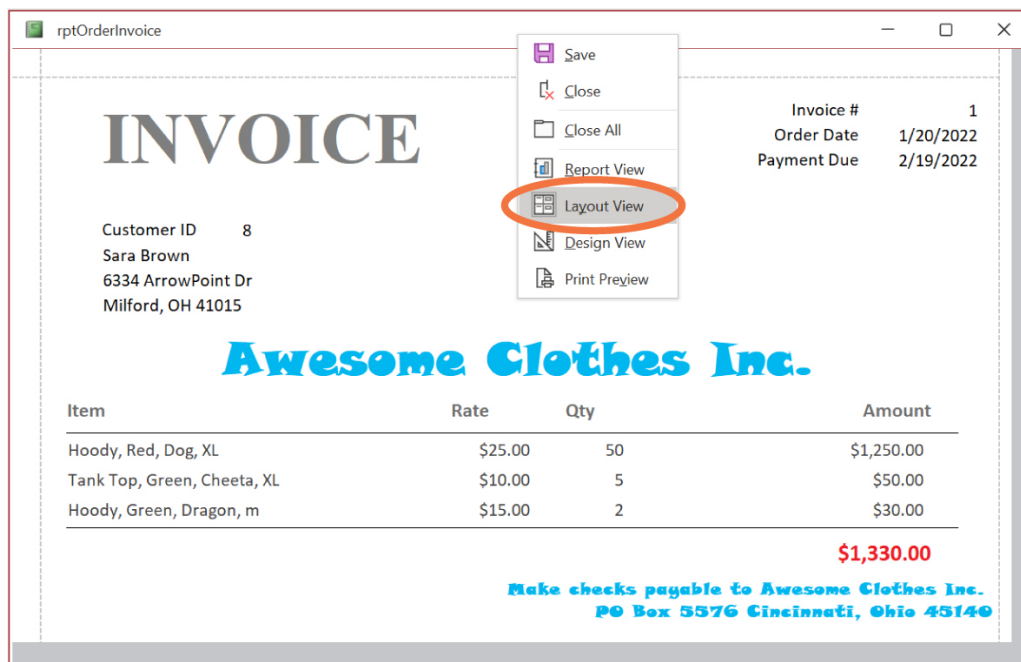


Figure 14.50: In the Layout View, you can make changes while viewing the form to see how the user will see the form with data included. (Used with permission from Microsoft)

Design View

In Access, Design View is a view that allows you to create and customize the layout and structure of a report. Design View is an important feature for creating professional-looking reports, as it allows you to customize the design and appearance of the report to suit your needs.

When you open a report in Design View, you'll see a grid that represents the layout of the report. The report is divided into several sections, including the Report Header, Page Header, Group Header, Detail, Group Footer, Page Footer, and Report Footer. Each section has its own set of properties that you can modify to change the appearance and behavior of the section. You can also add various types of controls to the report, including text boxes, labels, buttons, and images in this view. Grouping and sorting options to the report can help organize and display the data in a meaningful way.

Design View also allows you to set properties for the entire report, such as its orientation, margins, and page size. You can also add themes and styles to give the report a consistent and professional appearance. Overall, this view allows you to create custom reports that suit your needs in inventory management and retail store operations while at WorldCorp. By taking advantage of the many customization options available in Design View, you can create reports that are both informative and visually appealing.

As you can see in Figure 14.51, the Design View includes all sections of the report.



Figure 14.51: The gridlines in Design View are helpful to keep the form looking professional by lining up controls and sizing elements appropriately for the fields. (Used with permission from Microsoft)

Grouping and Sorting in a Report

Tools that can group and sort your information will be of utmost importance when organizing your report. Information is often easier to understand when it is divided into groups. For example, in the invoice for Awesome Clothes Inc., the sections are used to group the details and relevant information. In addition, placing a total (a sum of all Amounts) in the Report Footer section can replace a lot of manual interaction with a calculator. The customer can quickly see the total of the grouped sections, showing the amount owed.

Grouping and sorting are important features in Access reports that allow you to organize data in a meaningful way. Additional considerations why you might want to use grouping and sorting in your Access reports are endless. Grouping and sorting can make it easier for users to read and understand the data in a report. By grouping related data together and sorting it in a logical order, you can help users quickly identify patterns and trends. It also allows you to summarize data by creating subtotals or totals for each group. Sorting can help you identify outliers or anomalies in your data. For example, if you're sorting a list of customers by purchase amount, you might notice that one customer has made an unusually large purchase, which could indicate a potential sales opportunity or a data entry error.

Grouping and sorting can help you make more informed decisions by presenting the data in a clear way. By using these tools in different ways, you can analyze the data from multiple perspectives and identify opportunities for improvement. By using these features, you can create reports that are more informative and easier to understand, which can ultimately help you make better business decisions that affect our success at WorldCorp.

Filtering Records

Another useful tool in Access is the filtering tool. This gives you the ability to filter a report prior to printing or presenting the information. There are just a couple quick steps you can take every time to ensure the information that is obtained in the report is correctly filtered to the desired data points.

To start, open the report in Design View. Click on the Sorting and Grouping button in the Grouping & Totals section of the Design tab on the ribbon. In the Sorting and Grouping dialog box, select the field that you want to filter by. Simply click on the "..." button next to the Filter property. In the Filter dialog box, select the criteria that you want to use for the filter. You can select from a variety of options, including Equals, Not Equals, Contains, Does Not Contain, Between, and more.

Enter the value or values that you want to use for the filter. You can enter a single value, multiple values separated by commas, or a range of values. Next, click the OK button to close the Filter dialog box and save and close the report. Open the report in Print Preview or Report View to see the filtered results. Note that you can also apply filters to a report while viewing the report in Print Preview or Report View. To do this, click on the Filter button in the Sort & Filter section of the Home tab on the ribbon, and then select the criteria that you want to use for the filter.

To use an actual example, let's filter our customers living in Ohio. Simply right-click on states listed as OH, and click Text Filters - Equals OH (see Figure 14.52).

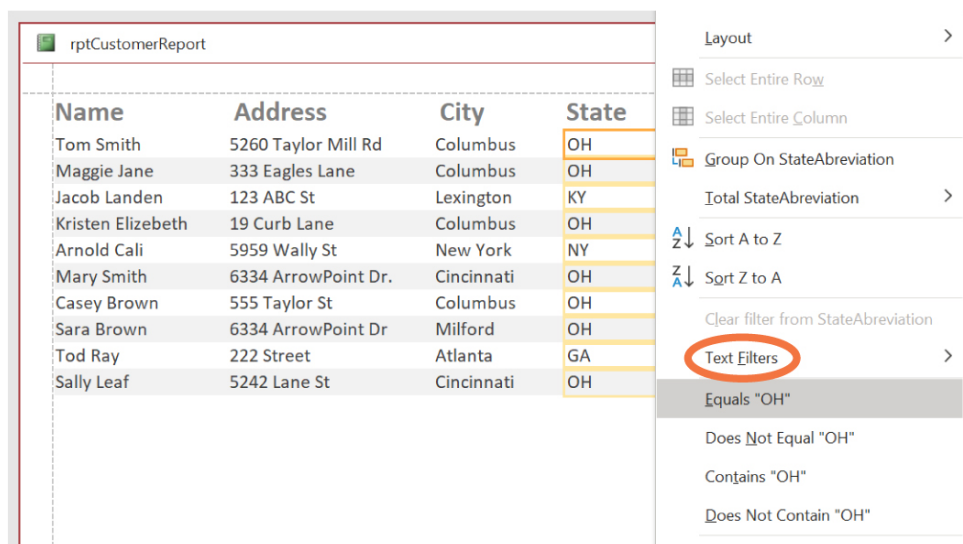


Figure 14.52: A report of information gathered from data entered into the form can be sorted and filtered by text or by fields. (Used with permission from Microsoft)

In Figure 14.53, Access applies the filter, and you can now print the report including only the data selected from the filter.

Name	Address	City	State	Zip
Tom Smith	5260 Taylor Mill Rd	Columbus	OH	41015
Maggie Jane	333 Eagles Lane	Columbus	OH	45140
Kristen Elizebeth	19 Curb Lane	Columbus	OH	45211
Mary Smith	6334 ArrowPoint Dr.	Cincinnati	OH	45140
Casey Brown	555 Taylor St	Columbus	OH	45140
Sara Brown	6334 ArrowPoint Dr	Milford	OH	41015
Sally Leaf	5242 Lane St	Cincinnati	OH	41015

Figure 14.53: The report will only display the information based on the filter; the remaining information will be hidden but is still contained in the table. (Used with permission from Microsoft)

Adding Totals and Subtotals

Adding totals and subtotals to a report in Microsoft Access is useful for summarizing data and providing meaningful insights to numerous activities at WorldCorp. A few reasons why you might want to add totals and subtotals to your report could include summarizing data, grouping data, identifying outliers, and comparing datasets.

Totals and subtotals allow you to summarize data in a report by providing aggregate values such as sum, average, count, minimum, maximum, and more. This can help users quickly understand the overall trends in the data. Grouping data by a particular field or criteria, such as by department or product category is very useful for subtotals by allowing users to analyze the data from different perspectives. Totals can be used to identify outliers or anomalies in the data. For example, if the total sales for a particular product are much higher than the other products, it may indicate that there is a potential opportunity for growth. In addition, they can be used to compare data across different time periods, geographic locations, or other criteria. This can help users identify trends and patterns in the data that may not be immediately apparent. Layout View be used to add totals or averages to a report.

To add a total in a report, start by opening the report in Design View. Click on the field where you want to add the total. Then, in the Properties window, scroll down to the Total property and select the type of total you want to add. See Figure 14.54 for the available options. Finally, save the report and reopen it in Print Preview or Report View to see the total.

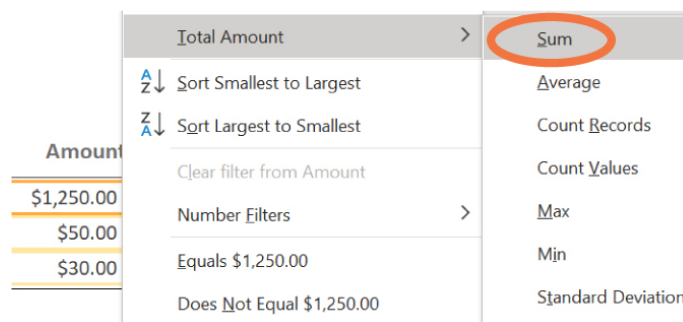


Figure 14.54: Access provides some tools to help you analyze information, although not to the extent that you can do data analysis in Excel. (Used with permission from Microsoft)

Adding Fields to a Report

To add fields to a report in Access, start by opening the report in Design View. Select the Add Existing Fields button in the Controls section of the Design tab on the ribbon. In the Add Existing Fields dialog box, select the table or query that contains the fields that you want to add to the report. Select the fields that you want to add to the report by checking the boxes next to their names. Click the Add button to add the selected fields to the report.

In addition, you can drag the fields to the desired location on the report once they are added. You can also resize the fields by dragging the edges of the control box. Make sure to save the report. Once you are done adding the fields to the report, you can customize their appearance and formatting by adjusting the properties in the Properties window.

As an example, we will review how text boxes can be added to a report to improve its ability to communicate information at WorldCorp. Start by opening your report in Layout View or Design View. Note: If your database does not contain any tables, the Field List pane will be empty. When you find the field(s) you want to add, double-click the field and then drag it from the Field List pane to the form or report, as Figure 14.55 shows.

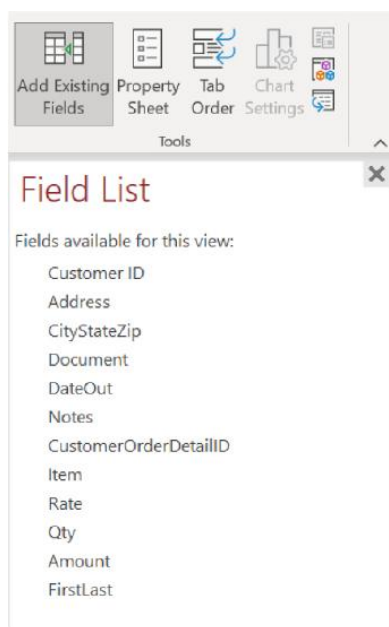


Figure 14.55: From the Report Layout Design tab you can add additional fields to the report from the Tools command group. (Used with permission from Microsoft)

Adding Time and Date

One area that is very important on an invoice is the invoice date and due date. In addition to the original purchase date, it can be helpful to calculate a due date or add a mailing date. In the following example, we will see the procedure for adding three dates: origin date, due date, and mailing date. This custom build will clearly explain to the customer the payment due date parameters.

Start in Layout View or Design View. The DateOut field, as shown in Figure 14.56 was included in the field list. This information can be presented on the report and used later in calculating the additional information.

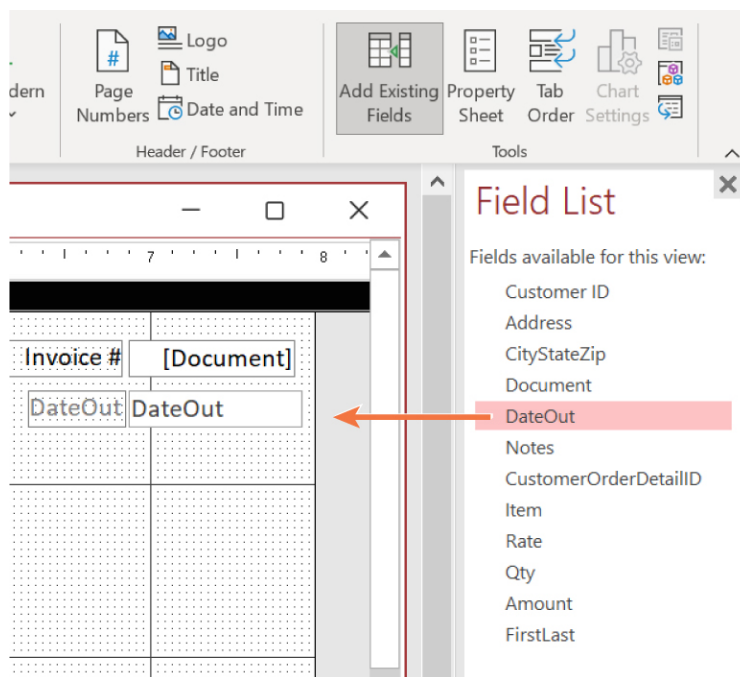


Figure 14.56: Choose the field you want added to the report and place in the report in the desired location. (Used with permission from Microsoft)

Formatting a text box prior to copying it is highly recommended to save time and keep attributes alike. In the example shown in Figure 14.57, copy and paste the label and text box twice, pasting the second instance directly below the first. Then, change the labels to represent the three dates the customer should see: Purchase date, Invoice date, and Due Date (Figure 14.58). For appearance, the borders on each label were changed to transparent.

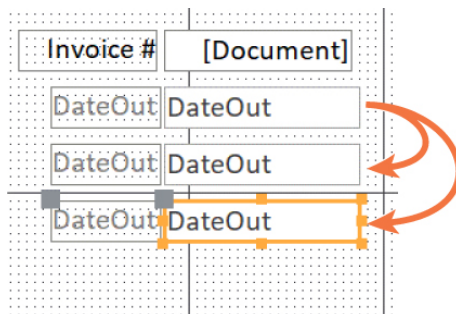


Figure 14.57: If you are using a similar layout for multiple fields in the report, you can insert once and copy and paste for the additional fields. (Used with permission from Microsoft)

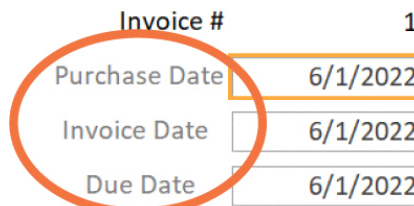


Figure 14.58: Descriptive titles are needed for each field in the report. (Used with permission from Microsoft)

Next, open the property sheet and select the Invoice Date text box. Change the control source to `=Date()`—a quick change that will provide the current date for the text box, as shown in Figure 14.59.

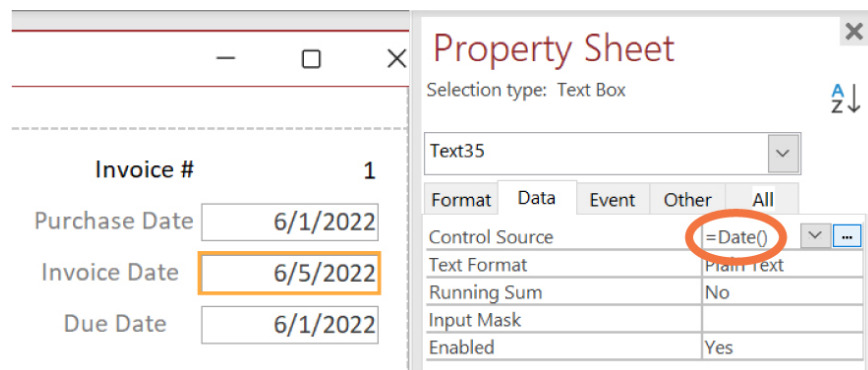


Figure 14.59: By changing the properties for the inserted field, you can add formulas to automatically populate certain aspects of the form. (Used with permission from Microsoft)

Select the Due Date text box and open the property sheet. Change the control source to `=[DateOut] + 30`. The control source is the field to which your text box is linked; it will display the data in that field and any updates you make to the field. This sample expression makes a calculation: It takes the field DateOut and adds 30 days to the field. The text box will display the expected due date of the invoice, as you can see in Figure 14.60. As this is probably the most important date in the report for the customer, change the font to red and make it boldface. This subtle change will help direct the customer's eyes to the important due date field.

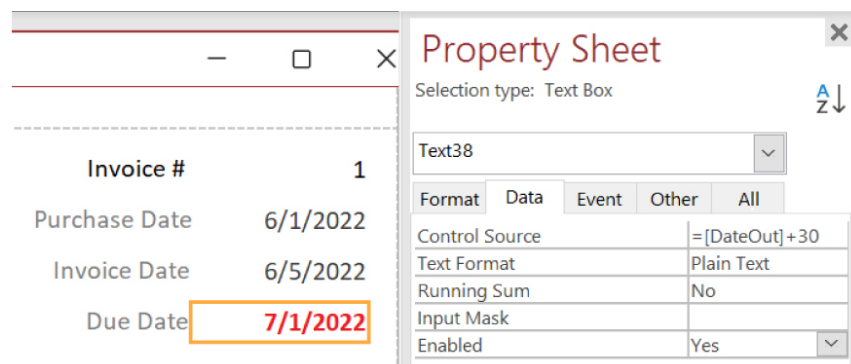


Figure 14.60: Mathematical operators can be utilized in field properties to modify dates as needed. (Used with permission from Microsoft)

Changing the Appearance of a Report

In Understanding and Using Databases, we briefly covered how to add images and custom text to your report. Here, we will go into detail about further customizing the appearance of your report.

Adding and Formatting Text

Many companies require particular directions when they invoice their customers. Terms and conditions are required and standardized through contracts. Placement, size, and accessibility are important, especially for current or future customers. In the example of the customer invoice, suppose you want to include a direction in the Page Footer section to instruct the customer on how to make a payment. In the following example, a label is used in the Report Footer section as direction for making payment. It provides information on how to direct the payment and where to send it. In designing this label, you will need to know how to set the size, font, and color to meet company guidelines or standards.

To do this, open the report in Layout View and select the Label option from the Report Layout Design Controls (see Figure 14.61). Draw the label in the Footer section and proceed to enter the payment directions.

Figure 14.61: The tools available to customize a report look very similar to those used to customize a form. (Used with permission from Microsoft)

While still on the label, select the Label option Format and adjust the font, size, and color within the font section. Within the label, font styles, sizes, and colors can be changed from the ribbon, as in all Microsoft Office products (Figure 14.62).

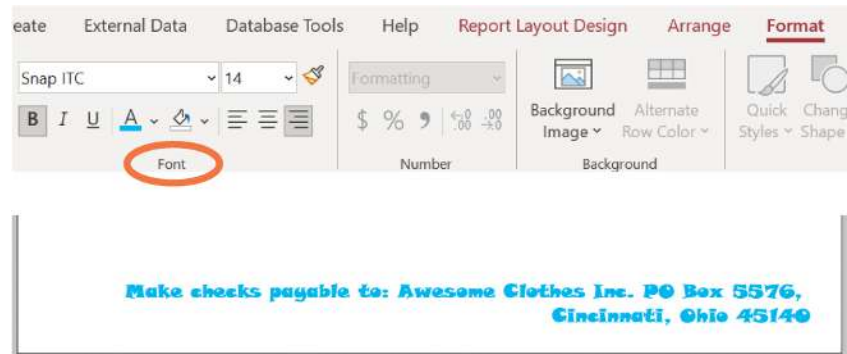


Figure 14.62: Text contained on a form can be adjusted by size, font type, and others similar to changing fonts in Word. (Used with permission from Microsoft)

Once complete, review the report in Print Preview View. Make sure the message is displayed in the correct position and with the correct formatting. You can repeat these steps for any other information that may be standardized on the report (e.g., contract guidelines or customer service contact information).

Adding Page Numbers

Open the report in Design View or Layout View. On the Design tab, in the Header/Footer group, click Page Numbers. When the Page Numbers dialog box appears, choose the format, position, and alignment that you want (Figure 14.63). Note: If you do not want a number to appear on the first page of the report, be sure to uncheck the option to clear the Show Number on the First page checkbox. When you are done, click OK, and the page numbers will be added to the report. (To see how the page numbers will look when the report is run, switch to Print Preview.)

Figure 14.63: (a) In the Header/Footer command group, page numbers can be added to multi-page reports. (b) Just as in Word, you can determine the formatting and location of the page number on the report. (Used with permission from Microsoft)

Adding Logos and Images

In the example of the customer invoice, a company logo helps clearly represent the company's ownership and brand. Beyond simply using font styles and colors to distinguish your company, a company logo can make reports look professional. Note: Before

finalizing a report that may go out to a large audience, be sure to check with your company's guidelines on placement, size, and color guidelines for logos. Larger companies often offer a marketing toolkit to help guide employees in the use of logos.

To better understand the use of logos, we will walk through steps for adding and modifying a static image on a form, using our clothing company example.

First, choose the image that you want to use as the logo for the clothing company. Ideally, this image has been given to you or made available. Company logos are usually saved as a JPEG or PNG file type. Open the report in Design View when you are ready to add the logo. Select the Image button in the Controls section of the Design tab on the ribbon. This will add an image control to the report.

Resize the image control to the desired size by dragging the edges of the control box. Right-click on the image control and select Properties from the context menu. In the Properties window, go to the Picture property and click on the ellipsis button (...) to select the image file that is the saved logo.

In the Picture Size Mode property, choose the Clip option to ensure that the image is displayed at its original size. Save the report once complete. You can also modify the appearance of the image by adjusting the properties in the Properties window. For example, you can change the border style, color, and thickness, as well as the background color of the control box. In this example, shown in Figure 14.64, the logo was placed in the center of the invoice header section, replacing the company name.

Figure 14.64: Company logos are an important part of reports such as invoices sent customers and are easily added in the header.
(Used with permission from Microsoft)

Now it is time to print your report. Before you do, make sure you are happy with the final version. Remember that although forms are designed for users, reports are for audiences and stakeholders. It's important to consider whether your message and data are clearly presented and whether the report meets all company guidelines for publication.

Spotlight on Ethics - Privacy in Two Acts: FERPA and HIPPA

Personal information such as addresses, work history, and in some cases medical history, could be contained in a variety of different databases and with a variety of different organizations or companies. Our personal information is regularly exchanged and sold to entities to solicit sales, send mailers, or for political purposes just to name a few examples. More and more people are becoming concerned with the use and privacy of their information.

Two areas of particular concern are medical data and academic/educational data. You might be familiar with FERPA – The Family Educational Rights and Privacy Act of 1974. This piece of legislation was passed to protect your academic records. The Act also gives protections to individuals for how their educational records are used by government agencies such as the Department of Education. Your right to keep those records private is part of the legislation. In addition, you have the right to request your records at any time. The data essentially belongs to you. There are certain provisions that allow usage of the data for reporting and auditing purposes. But when the data is used, personally identifiable information is generally not disclosed.

In a similar manner, HIPPA, or the Health Insurance Portability and Accountability Act of 1995, protects your medical information. Your specific medical information is kept private under most circumstances. Information such as treatment plans or medication might be shared with insurance companies but that information is protected through the insurance companies from outside entities under the same HIPPA legislation.

In general, under both legislative acts, unless there is a justifiable compelling reason to disclose your personal information, it is kept private. Both acts give options for you to disclose to certain entities. For example, you can decide to allow your parents access to your educational records or your spouse/partner have access to your medical records. Signed approval for such disclosure is needed to release the information to someone other than yourself.

This page titled [2.1.8: Customizing Reports](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.4: Customizing Reports** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.9: Using Macros

 [Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Learning Objectives

By the end of this section, you will be able to:

- Record a macro that automates a common task
- Record a macro that enhances database functions
- Demonstrate some understanding of macro code

A macro is a set of instructions that can automate repetitive tasks or perform complex operations with a single click. Macros can be created using the Macro Builder, which provides a graphical interface for creating and editing macros. Examples of how a macro can be used in Microsoft Access are many.

As an example, your database under construction that tracks inventory for retail stores that WorldCorp would like to launch has numerous options to consider. Each time a new transaction is entered, you need to generate an invoice for the customer that includes their name, the date of the sale, and a list of the items bought. To automate this process, you can create a macro that performs the following tasks:

1. Open the Orders table and find the most recent order.
2. Get the customer's name from the Customers table.
3. Get the date of the order.
4. Get a list of the items ordered from the Order Details table.
5. Create a new record in the Invoices table.
6. Populate the new record with the customer's name, the date of the order, and the list of items ordered.
7. Save the new invoice record.

Once you have created this macro, you can run it whenever you need to generate a new receipt. Access will automatically perform all the necessary steps, saving you time and reducing the risk of errors. Macros provide the ability to automate repetitive tasks, improve productivity, and reduce errors in database management.

Creating and Recording Macros

Macros can make tasks that you routinely perform in Access quicker through automation. Macros are built using computer coding language to perform specific tasks such as running reports, printing forms, or copying data from one table into another table.

To begin using macros, open the macro design builder, located under the Create tab. This will create a blank macro builder, which can be saved and stored for later use in the All Access Objects area. The Macro Designer on the right side (Figure 14.65) contains the Action Catalog, where all of your actions will be stored in folders. You have Data Entry options, Data Import/Export, and so on, and on the left in the main area you have a blank canvas to start building your macro.

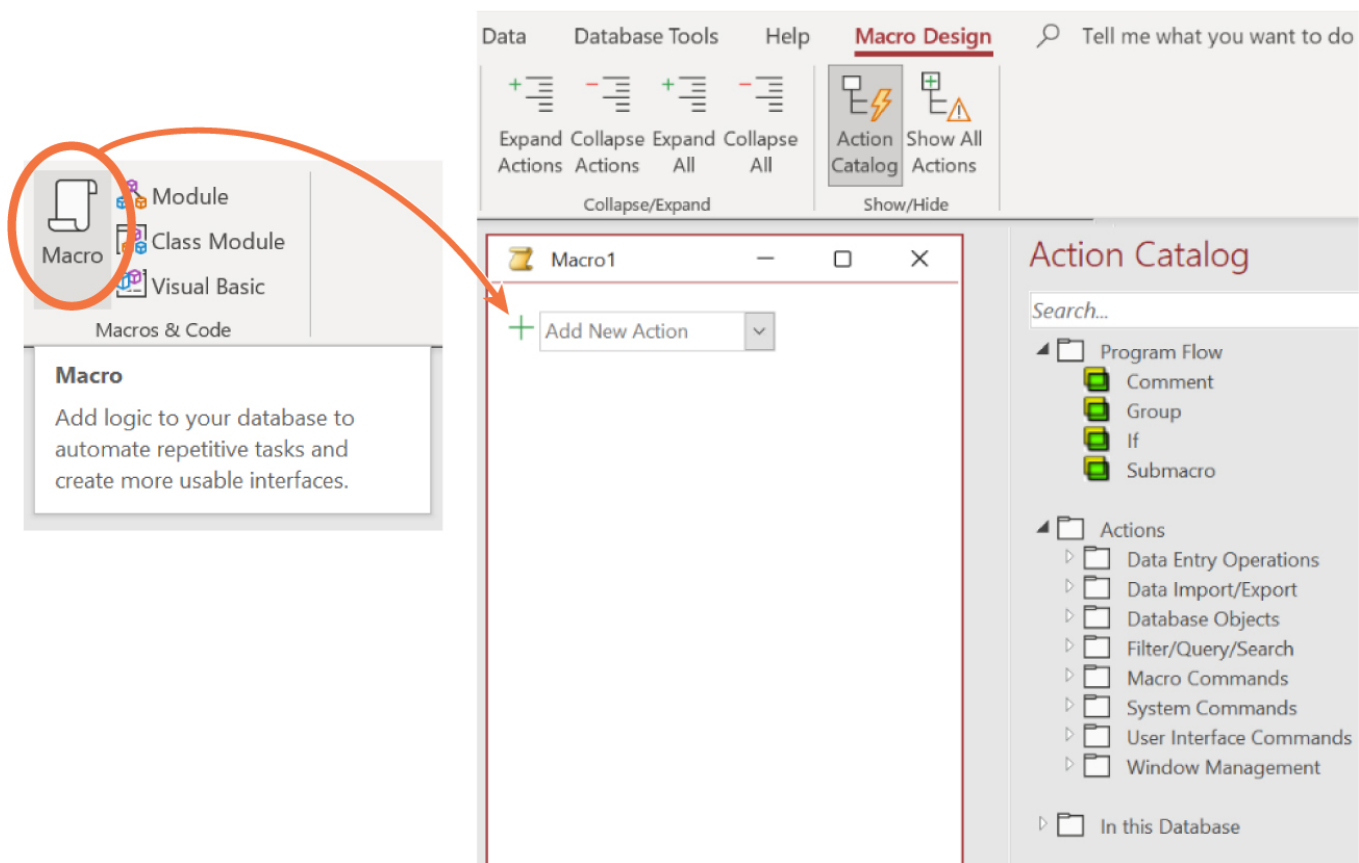


Figure 14.65: Macros are computer code written within the program to tell the app an action or actions to perform. (Used with permission from Microsoft)

Recording a Macro to Automate Common Tasks

Recording a macro to automate tasks in Access is a common way to save time and increase productivity. When you record a macro in Access, you essentially create a set of instructions that Access can use to automatically perform a sequence of tasks.

Let's say you frequently need to create a report that shows the total sales for each region of the country in the growing area of retail sales. You can use the Macro Recorder to automate this process. First, select the Create tab in the ribbon and select Macro. In the Macro Builder, click the Record button to start recording the macro. Perform the actions you want the macro to automate. For example, you could select the fields you want to include in the report, specify the criteria for grouping the data by region, and apply formatting to the report.

When you have finished performing the actions, click the Stop Recording button in the Macro Builder. Give the macro a descriptive name and save it. Now, whenever you need to create a report that shows the total sales for each retail store, you can simply run the macro you recorded (Figure 14.66). Access will automatically perform all the steps you recorded, saving you time and reducing the risk of errors.

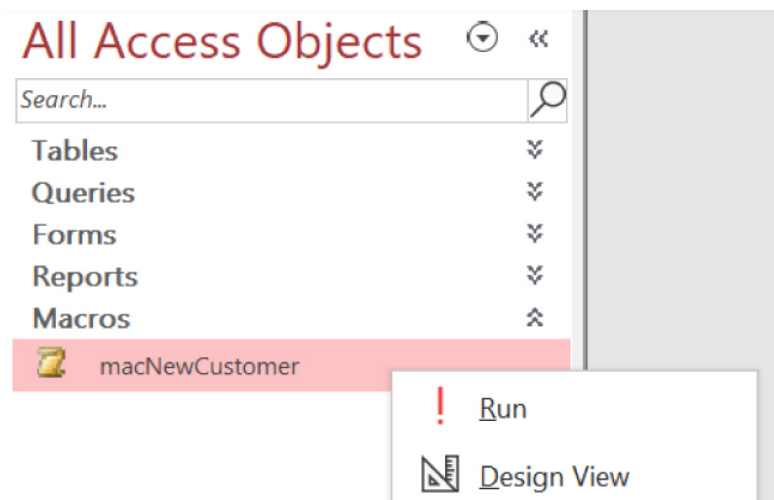


Figure 14.66: Recording a macro is an easy way to automate a task such as running a report or printing a form that you perform regularly. (Used with permission from Microsoft)

A common use of macros is to add a button to a form. In the previous section, we discussed how a Main Menu form can help improve user experience by creating a navigation map. Buttons can have macros embedded that open and close certain forms; they can also control what record source is displayed.

For example, suppose you want to add your customer form to the Main Menu. The original goal is to add a button that will open the customer form. However, if you want to add a button that will add a new customer, you can create a macro that will open the customer form and move the record to the new source. The open form is shown in Figure 14.67.

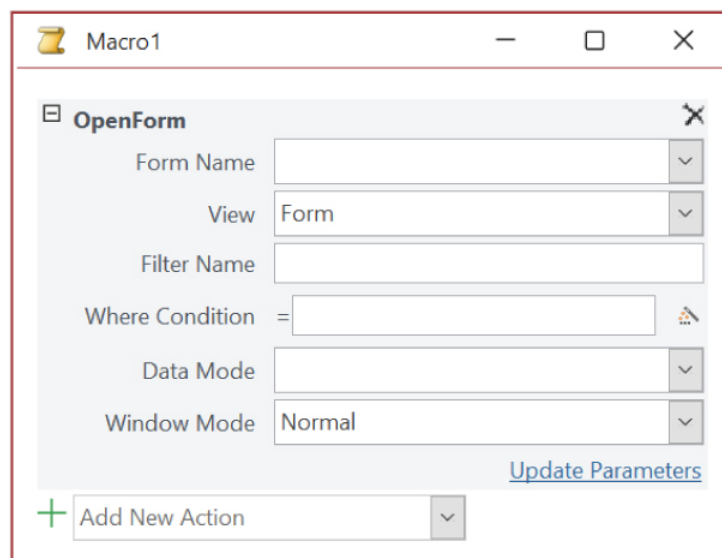


Figure 14.67: Macros can be added to forms through buttons and can be programmed to perform a wide variety of actions such as adding a new field. (Used with permission from Microsoft)

You will see the form name with a drop-down arrow, which will show you the forms available in your database. For each form, there is a View option, which will allow you to open the form in Form View, Design View, or Print Preview View. You also can apply a filter name or a condition. In Figure 14.68, the macNewCustomer macro that was created centered on finding particular records. You will notice this option is available to modify.

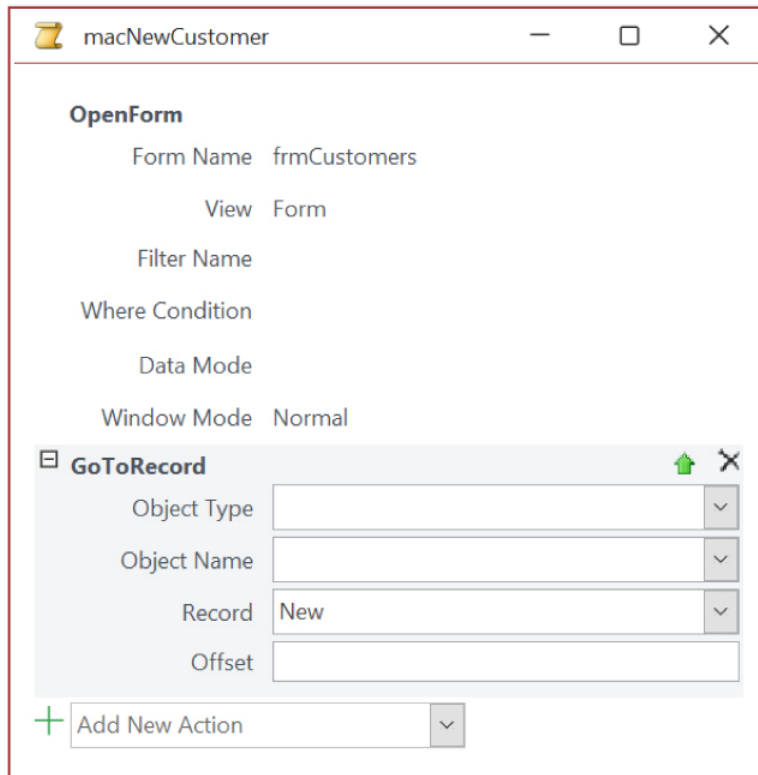


Figure 14.68: The drop-down menus give you options such as adding or deleting records in the form. (Used with permission from Microsoft)

Recording a Macro to Enhance Database Functions

By customizing your database with macros, you can create a system for accomplishing many similar operations easily and efficiently. Two common applications for macros are displaying message boxes and validating data:

- **Displaying a message box:** You can use the MsgBox action to display alerts, warnings, and other useful information. The MsgBox action has four arguments: Message, Beep, Type, and Title.
- **Data validation:** Often used in Microsoft Excel, data validation is an extremely helpful step to take prior to adding the data to the database. To ensure that only valid data is entered in a form, you can specify a validation rule for the control in the form or set record and field validation rules in the underlying table design. For more complex data validation, you can use a macro or an event procedure to specify the rule.

As another example, you can use the RunApplication macro to run a Microsoft Windows-based application, such as Excel, Word, or PowerPoint, from within Microsoft 365. For example, you may want to paste Excel spreadsheet data into your Access database. This can greatly enhance your database's ability to work with imported data.

Reviewing Completed Macros

One of the easiest ways to review a macro is to simply run the macro and see if the desired outcome occurs. At any point, a macro can be edited to better serve the database developer's goal for how it will function.

To review completed macros in Access that have been developed, you can start by opening the database that contains the macros you want to review. Go to the Navigation Pane on the left side of the screen and select the Macros option. This will display a list of all the macros in the database.

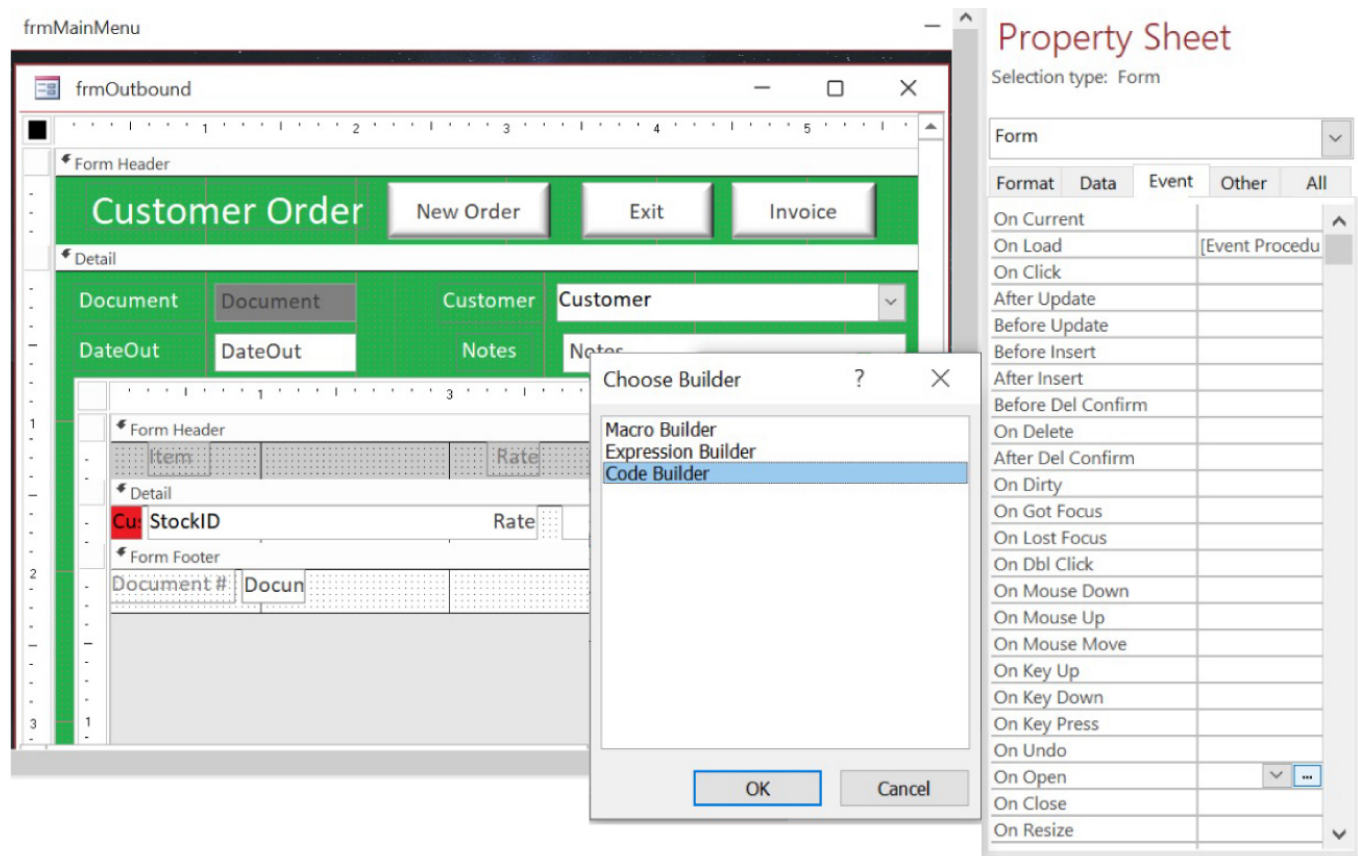
Double-click on the macro you want to review. This will open the macro in Design View. In this view, you can see all the actions and conditions that make up the macro. You can review each action to ensure that it performs the intended task and that it is configured correctly.

To run the macro and test its functionality, click the Run button in the Macros tab of the ribbon. This will execute the macro and perform all the actions defined within it. If you encounter any issues or errors while running the macro, you can use the Debug mode to review the code and identify the problem. To exit the Design View of the macro, click the Close button in the top-right

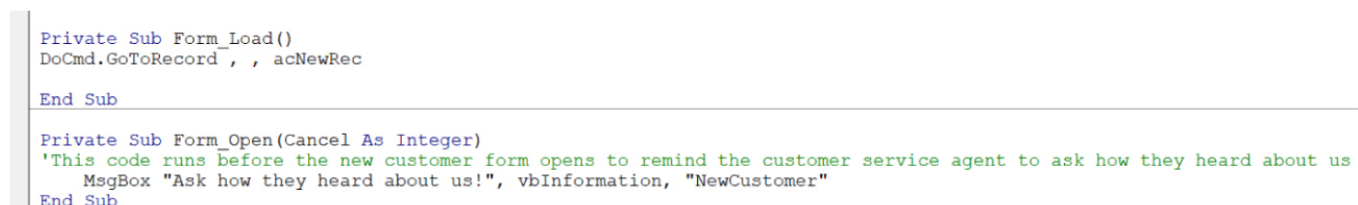
corner of the screen. By reviewing completed macros in Access, you can ensure that they function as intended. Additionally, you can use the insights gained from reviewing completed macros to improve future developments with your overall programming skills. These are big steps are mastering relationship databases.

Understanding Macro Code

Figure 14.69a shows how to add code to an event procedure. Although there is a steep learning curve for understanding the syntax, its applications are unlimited. In this example, we want to create a message box that will appear when a user opens the form. The code is written out just below the comments (shown in green in Figure 14.69b) that were added. Whenever you add new code, either directly or by using a macro, be sure to include a comment. If there is a problem or a future need for a change, comments can help clarify what the code was designed to accomplish.



(a)



(b)

Figure 14.69: (a) Using the Code Builder tool you can write the computer code to perform specific actions in Access. (b) The Code Builder tool adds in the specific requirements for the computer code to work such as “ ” and commands. (Used with permission from Microsoft)

After opening the builder options menu, select Code Builder. This will open a new window, as shown, that will allow the developer to type in the desired code directly. This code can be broken into four sections.

First, MsgBox has been identified as the action. Because it is placed in the event procedure On Open, a message box will appear when it opens.

Second, text for the message box—in this example, “Ask how they heard about us!”—is included to remind associates to ask the customer this basic engaging question while taking a new customer order.

Finally, there is a request for which form to open after the message box appears. The following message box shown in Figure 14.70 was created through the code.

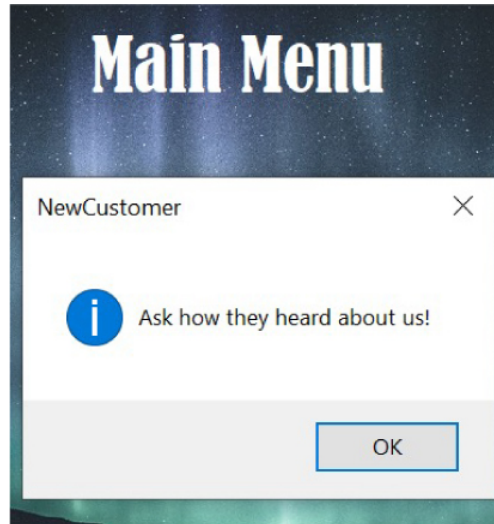


Figure 14.70: Message boxes can be added to forms to give users specific instructions. (Used with permission from Microsoft)

Understanding Visual Basic

Visual Basic for Applications (VBA) is a programming language that is integrated into Access, allowing you to create custom solutions for data management, reporting, and other tasks. VBA provides a powerful set of tools for developers to create robust and scalable applications within Access.

Some examples of how VBA can help developers in Access include automating repetitive tasks. With VBA, developers can create macros and scripts to automate repetitive tasks, such as importing data, running reports, or updating records. This can save time and reduce errors by eliminating the need for manual data entry.

Customizing user interfaces can be written in VBA and can be used to create custom forms and dialog boxes to provide a user-friendly interface for working with data. Developers can add buttons, menus, and other controls to simplify complex tasks and make the application more intuitive. VBA can also create custom functions and algorithms for analyzing data within Access. Developers can create advanced data models and calculations to help gain insight data. One of the keys to VBA is opening up the ability to integrate with other applications. It can be used to integrate Access with other applications, such as Excel or Outlook. This allows developers to create custom workflows and exchange data between applications.

VBA can be used to create custom reports within Access, including dynamic reports that update based on user input or other data sources. In summary, VBA is a powerful tool that seems to offer endless abilities and functionality in Access. It provides a wide range of features and capabilities for creating custom solutions to meet the specific needs of users and organizations. With VBA, you can automate repetitive tasks, customize user interfaces, enhance data analysis, integrate with other applications, and create custom reports, and that is just scratching the surface of possibilities.

Link to Learning

Using VBA will take some practice as you are essentially writing computer code. VBA has a wide variety of applications to assist you in working Access and also in Excel. Most users can record macros to meet most of their automation needs in Access. However, if you have more advanced needs and are familiar with computer coding, VBA can be quite powerful. The best way to show how VBA can expand the power and agility of Access is by providing a demonstration. Please view this [video tutorial on how to use VBA to create databases within Access](https://www.youtube.com/watch?v=K11Lk0tUg54) to learn more.

This page titled [2.1.9: Using Macros](#) is shared under a [CC BY-NC-SA 4.0](#) license and was authored, remixed, and/or curated by [OpenStax](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.

- **1.1.1: Introduction to Microsoft Excel** by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.
- **14.5: Using Macros** by OpenStax is licensed [CC BY 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

2.1.10: Microsoft Access Terminology

[Click here to view Microsoft Access - Building Proficiency Chapter Sections](#)

Microsoft Access Terminology

Term	Definition
action query	query that specifies a particular action, such as creating a new table, deleting a row or rows from a table, updating records, or creating new records
aggregate query	instruction that carries out calculations on a group of records rather than on individual records
analytical database	a database used by an organization for analytics and decision-making
Append	to add on to the end of an object, for example, adding records to the end of an existing table in a database
append query	query that adds records from one or more tables to the end of another table or tables
arithmetic operator	used to calculate a value from two or more numbers or to change the sign of a number from positive to negative, or vice versa
Arrange tab	a tab in the Form or Report Layout toolbar that allows you to make changes to the arrangement of tables, rows, and columns in a Form or Report
Autonumber	a data type that describes a unique sequential or random number assigned by Access as each record is entered and that is useful for data that has no distinct field the unique. This data type cannot be manually changed or typed in
Blank form	allows you to create a new, blank form, which you can then customize and format the way you want
Bound	a control that retrieves data from an underlying table or query
columnar layout	a form layout that shows a single record
command control	tool that is used to start an action or a set of actions
comparison operator	used to compare values and return a result that is true, false, or null
control source	area in the property sheet that controls which data is displayed in a particular field, as well as any updates you make to the field
criteria	expressions that a database management system uses to limit the results of a query based on matching field values
crosstab query	instructions that calculate and restructure data for easier identification of trends or patterns
dark data	data that is typically not used for analysis and therefore may or may not be valuable to an organization
dashboard	allows users to view account activity at a glance and quickly find records and reports
Data	Facts about people, events, things, or ideas
data lake	a place to store data that keeps raw copies of the original documents
data landscape	the entirety of data storage and sharing in an organization
data mart	a subset of a data warehouse, optimized for specific workgroups or functions
Data type	classification identifying the kind of data that can be stored in a field such as numbers, text, or dates
data validation	process of ensuring that data is entered correctly into a form by specifying a validation rule for the control in the form
data warehouse	a large store of data within an organization, often holding historical information and used for decision-making
database	a structured set of data stored in a way that allows us to make use of it
datasheet layout	a form layout that lists all records as they would appear in a datasheet
Datasheet view	the Access view that displays data organized in columns and rows similar to an Excel spreadsheet
DBA	stands for Database Administrator, an IT professional whose primary responsibilities include database installation, configuration, design, migration, performance troubleshooting, as well as backup and data recovery
delete query	query that deletes a group of records from one or more tables
Design tab	a tab in the Form or Report Layout toolbar that allows you to make modifications to the design of the Form or Report
Design view	an Access view that displays the detailed structure of a table, query, form, or report. Also known as developer's view. For Forms and Reports, this may be the only tasks are performed. Only the controls, not the data are displayed in this view
Detail	the area in a Form or Report that data is displayed and interacted with; may include text boxes, checkboxes, combo boxes, labels, pictures, and other data editing controls
Detail section	information that appears once for every row in the record source that is being referenced
Developer's view	also known as Design View. The most powerful view in Access
development database	a database used for experimental testing so that problems can be found and resolved before being pushed to the production database
digital database	a computerized file that stores data in a structured or organized way and also allows users to make changes to transform data
digital platform	a place where users can interact with and explore the products and services offered by a company
Enable content	Microsoft Office's built-in security feature
End users	the person or people that use the database
Export	also known as a 'dump', extracting and saving a database file that can be used for storage, distribution, or importing into another system
external data	data stored outside the current database
Field	a single piece of information that is stored in every record and is represented by a column in a database table
Field name	the description that identifies a field
Flat databases	a simple database file that is not related or linked to any other collections of data
flat files	individual digital files that have a tabular structure, such as spreadsheets, tables, and lists

Term	Definition
Form	a database object that you can use to enter new records into a table, or to edit, delete, and display existing records in a table
form control	part of a form used to enter, edit, or display data; controls can be bound, unbound, or calculated
Form design	allows you to work with the underlying structure of your form
Form footer	the bottom margin of each page of an Access form
Form header	the top margin of each page of an Access form
form property	option that is set in Design View on the property sheet; they control how the form looks, works, and interacts with the rest of the database
Form view	the Access view in which you can view, modify, delete or add records in a table but you cannot change the layout or design of the form
Form wizard	a step by step way to create a form
Format tab	a tab in the Report Layout toolbar that allows you to make changes to the formatting of a Report such as Fonts and Backgrounds
group	heading that organizes data items that have something in common
Group Footer	area where summary information appears at the end of each group of records
Group Header	area that displays names at the beginning of each new group of records
Group, Sort and Total	a pane that displays at the bottom of the Report window in design view in which you can control how information is sorted, grouped, and totaled in a report
Importing	the process of copying data from another file, such as an Excel Spreadsheet, into a separate file, such as an Access table
Information	data that is accurate, timely, and organized in a useful manner
Is Null	criteria used in queries that searched for fields that are empty
join	a clause used to create a relationship or connection between two records in a table
justified layout	a layout showing a record-by-record form, arranged across the screen
key field	a field in a table that serves as a unique identifier for an individual record in a table
Key Terms	
Knowledge	processed information that is useful when making important decisions
Label	a control on a form or report that contains descriptive information, usually a field name or title
Layout view	the Access view in which you can make changes to a form or report while the data from the underlying data and source displays
logical operator	used to combine two Boolean values and return a true, false, or null result; also known as Boolean operators
macro	tool that allows you to automate tasks and add functionality to your forms, reports, and controls
make-table query	query that creates a new table from all or part of the data in one or more existing tables
many-to-many relationship	relationship in which each record in one table can have many related records in another table, and vice versa; this type of relationship, which really consists of two relationships, is made possible only by defining a table
message box	used to display alerts, warnings, and other useful information
multiple-item form	a form that displays information about more than one record or item at a time
navigation form	a form that allows you to add formatting, text, and commands to act as an interface, or navigation area, for your database
Navigation pane	an area of the Access window that displays and organizes the names of the objects in a database and allows objects to be opened for use
NoSQL database	database that stores large amounts of data, including documents and images, in their original format, not in tables
null criteria	query used to find records that have a null value in a particular field
object	an item in a relational database
Object type	a way to sort objects in the navigation pane so they are grouped by type
Objects	the basic parts of a database that you create to store your data and to work with your data; includes tables, queries, forms, and reports
One-To-Many	the most common type of database relationship between two tables where one record in the first table corresponds to many records in the second table
one-to-many relationship	relationship in which each record in one table has one or more related records in another table, but a record in the second table has only one matching record in the
one-to-one relationship	relationship in which each record in one table has at most one related record in another table
page footer	the bottom section of a document page, which typically repeats across pages and contains information useful to the report reader
page header	the top section of a document page, which typically repeats across pages and contains information useful to the report reader
Page setup	a tab in the Report Layout toolbar that allows you to make changes to the Setup of a report to prepare it for printing, such as Page Size and Page Layout
parameter query	query that allows you to prompt users whenever a query runs
Populate	the action of filling a database table with records
predefined queries	queries used to answer frequently asked questions that meet specific organizational needs
Primary key	a required field that uniquely identifies a record in a table
Print preview	a Report view that displays what the printed report will look like
production database	a database that runs transactions related to an organization's core business functions
Property sheet	a list of characteristics, or properties, for fields or controls on a form or report in which you can make precise changes to each property associated with the field or

Term	Definition
Queries	a database object that retrieves specific data from one or more database objects, either tables or queries, and displays the requested data in a datasheet
query	a structured data request
query criterion	expression that Access compares with query field values to determine whether to include the record that contains each value
Query grid	the bottom pane of the Query Design View window in which you specify the fields, sort order, and limiting criteria for the query
Query wizard	a step by step way to create four types of queries to use in accessing and modifying the data in your database
range criteria	query used to specify a range of values for a field
record navigation buttons	options that open whenever records are shown, enabling users to execute commands in a database
Redundancy	in a database, information that is duplicated in a manner that indicates poor database design
relational database	a database whose design and structure enable it to communicate with other systems that hold data
relational database management system (RDBMS)	computer software that provides the functions for a relational database
Relational databases	a sophisticated type of database that has multiple collections of data within the file that are related to one another
relationship	the association between two or more tables based on common fields or columns. It represents the way data is related and connected across different tables within the database
Relationships	an association that you establish between two tables based on common fields
report	formatted summary of specific information in a database
Report	also known as the Report Tool, a quick and easy way to create a new Report
report detail	the area where data from the Access database is displayed for each record
report footer	the bottom section of a report that does not typically repeat across pages and contains the name and dates of the report
report header	the top section of a report that does not typically repeat across pages and contains the number of pages in the entire report
Report view	a Report view is the default view used when you double-click a report in the Navigation Pane
Report wizard	a step by step way to create a Report
select query	query that retrieves data from one or more tables and displays the result in a datasheet
Short text	An Access data type that describes text which can be a combination of letters and numbers that are not used in calculations, such as zip code
Show	a checkbox in the Query Grid that enables fields to be shown or hidden
simple criteria	query that specifies an exact value or expression to filter results
simple form	a form that displays and allows editing of a single record at a time
single point of truth (SPOT)	the idea that there is one place storing information that all business units agree is factual
split form	a form that has two functions; for example, it might display a single record as well as a view of the datasheet in which the record is held
SQL query	query created by using a statement in SQL, a programming language designed to manage data stored in relational databases
SQL View	a view that shows a query in structured query language form
structured query language (SQL)	language used to create and run queries in many relational databases
subform	form that is inserted into another form (the main form)
syntax	set of rules by which the words and symbols in an expression are correctly combined
tab order	helps control which section of the form the cursor moves to next
table	the structure that holds individual data records in a database
Tables	a database object that organizes and presents text and data in columns and rows and is the foundation for databases
tabular layout	a form layout that shows all records listed in columns, similar to a datasheet
Text box	a bound control on a form or report that displays the data from an underlying table or query
Unbound label	a control that does not have a source of data, such as the title in a form or report
update query	query that makes global changes to a group of records in one or more tables
User friendly	easy to use and understand, it is a goal to create databases that is easy to use
Visual Basic for Applications (VBA)	event-driven programming language and environment from Microsoft that provides a graphical user interface (GUI), which allows programmers to modify code by dropping objects and defining their behavior and appearance
wildcard	symbol used to represent an unknown character in a search string

2.1.10: Microsoft Access Terminology is shared under a [CC BY-NC-SA](#) license and was authored, remixed, and/or curated by LibreTexts.

- [1.1.1: Introduction to Microsoft Excel](#) by LueCrecy Ragan - Arkansas State University - Beebe is licensed [CC BY-NC-SA 4.0](#). Original source: <https://openstax.org/details/books/workplace-software-skills>.

Index

A

Access

- 1.1.7: Entering and Editing Records
- 1.1.9: All About Queries
- 2.1.2: Advanced Queries in Microsoft Access
- 2.1.4: Form Basics

actionable information

- 2.1.7: Creating Reports in Microsoft Access

advanced database applications

- 2.1.1: Microsoft Access and Data

advanced queries

- 2: Microsoft Access - Building Proficiency

audience

- 2.1.7: Creating Reports in Microsoft Access

automation

- 2.1.9: Using Macros

B

business analysis

- 1.1.1: Introduction to Microsoft Access

business intelligence

- 1.1.1: Introduction to Microsoft Access
- 2.1.1: Microsoft Access and Data

C

create form

- 2.1.6: Multiple Table Forms

criteria

- 1.1.7: Entering and Editing Records

criteria types

- 2.1.2: Advanced Queries in Microsoft Access

custom form

- 2.1.5: Customizing Forms

customer experience

- 1.1.8: Querying a Database

customization

- 2.1.7: Creating Reports in Microsoft Access

customizing reports

- 2.1.8: Customizing Reports

D

data accuracy

- 1.1.3: Database Design and Four Main Objects

data analysis

- 1.1.1: Introduction to Microsoft Access
- 1.1.2: Microsoft Access- Main Features and Navigation
- 1.1.8: Querying a Database
- 2.1.1: Microsoft Access and Data
- 2.1.2: Advanced Queries in Microsoft Access
- 2.1.8: Customizing Reports

data comprehensibility

- 1.1.5: Primary Keys and Table Relationships

data consistency

- 1.1.5: Primary Keys and Table Relationships

data entry

- 1.1.4: Creating a Table
- 1.1.6: Maintaining Records in a Database
- 1.1.7: Entering and Editing Records
- 2.1.4: Form Basics
- 2.1.5: Customizing Forms

data integration

- 2.1.1: Microsoft Access and Data

data integrity

- 1: Microsoft Access - Setting a Foundation
- 1.1.2: Microsoft Access- Main Features and Navigation

Navigation

- 1.1.7: Entering and Editing Records
- 2: Microsoft Access - Building Proficiency
- 2.1.3: Creating Forms in Microsoft Access
- 2.1.6: Multiple Table Forms

data lineage

- 1.1.6: Maintaining Records in a Database

data management

- 1: Microsoft Access - Setting a Foundation
- 1.1.1: Introduction to Microsoft Access
- 1.1.6: Maintaining Records in a Database
- 1.1.8: Querying a Database
- 1.1.9: All About Queries
- 2: Microsoft Access - Building Proficiency
- 2.1.1: Microsoft Access and Data
- 2.1.9: Using Macros

data manipulation

- 1.1.2: Microsoft Access- Main Features and Navigation

data organization

- 1.1.1: Introduction to Microsoft Access

data processing

- 1.1.1: Introduction to Microsoft Access

data professionals

- 1.1.6: Maintaining Records in a Database

data protection

- 2.1.8: Customizing Reports

data redundancy

- 1.1.3: Database Design and Four Main Objects

data reporting

- 1.1.2: Microsoft Access- Main Features and Navigation

data retrieval

- 1.1.2: Microsoft Access- Main Features and Navigation
- 1.1.9: All About Queries

data security

- 2: Microsoft Access - Building Proficiency
- 2.1.1: Microsoft Access and Data

data types

- 1: Microsoft Access - Setting a Foundation

data validation

- 1: Microsoft Access - Setting a Foundation
- 2.1.9: Using Macros

data visualization

- 1.1.1: Introduction to Microsoft Access

data warehouse

- 1.1.2: Microsoft Access- Main Features and Navigation

database

- 1.1.8: Querying a Database
- 1.1.9: All About Queries
- 2.1.4: Form Basics
- 2.1.9: Using Macros

database design

- 1: Microsoft Access - Setting a Foundation
- 1.1.3: Database Design and Four Main Objects
- 2: Microsoft Access - Building Proficiency
- 2.1.1: Microsoft Access and Data
- 2.1.3: Creating Forms in Microsoft Access
- 2.1.5: Customizing Forms

database efficiency

- 1.1.5: Primary Keys and Table Relationships

database management

- 1.1.2: Microsoft Access- Main Features and Navigation
- 1.1.6: Maintaining Records in a Database
- 1.1.7: Entering and Editing Records
- 2.1.2: Advanced Queries in Microsoft Access
- 2.1.3: Creating Forms in Microsoft Access
- 2.1.6: Multiple Table Forms

database management systems

- 1.1.1: Introduction to Microsoft Access

database objects

- 1.1.3: Database Design and Four Main Objects

database queries

- 2.1.2: Advanced Queries in Microsoft Access

datasheet

- 2.1.4: Form Basics

datasheet view

- 1.1.6: Maintaining Records in a Database

date entries

- 1.1.4: Creating a Table

deleting records

- 1.1.6: Maintaining Records in a Database

design view

- 2.1.4: Form Basics

desktop database

- 1.1.2: Microsoft Access- Main Features and Navigation

E

editing records

- 1.1.6: Maintaining Records in a Database

effective forms

- 2.1.3: Creating Forms in Microsoft Access

effective reports

- 2.1.7: Creating Reports in Microsoft Access

efficiency

- 2.1.6: Multiple Table Forms

F

filtering

- 1.1.7: Entering and Editing Records

filtering data

- 2.1.2: Advanced Queries in Microsoft Access

foreign key

- 1.1.5: Primary Keys and Table Relationships

form design

- 2.1.3: Creating Forms in Microsoft Access
- 2.1.5: Customizing Forms

form view

- 1.1.6: Maintaining Records in a Database

Form Wizard

- 2.1.4: Form Basics

forms

- 1.1.3: Database Design and Four Main Objects
- 1.1.7: Entering and Editing Records
- 2: Microsoft Access - Building Proficiency
- 2.1.3: Creating Forms in Microsoft Access
- 2.1.4: Form Basics

G

graphical user interface

- 2.1.5: Customizing Forms

I

information retrieval

2.1.2: Advanced Queries in Microsoft Access

information systems

1.1.1: Introduction to Microsoft Access

1.1.9: All About Queries

input mask

1.1.4: Creating a Table

inventory database

2.1.5: Customizing Forms

inventory management

2: Microsoft Access - Building Proficiency

2.1.1: Microsoft Access and Data

2.1.8: Customizing Reports

J

joins

1.1.9: All About Queries

K

key performance indicators

2.1.7: Creating Reports in Microsoft Access

L

layout

2.1.4: Form Basics

learning objectives

2.1.2: Advanced Queries in Microsoft Access

2.1.3: Creating Forms in Microsoft Access

2.1.7: Creating Reports in Microsoft Access

2.1.8: Customizing Reports

logical operators

1.1.4: Creating a Table

M

macro

2.1.9: Using Macros

master/detail form

2.1.6: Multiple Table Forms

meaningful insights

2.1.8: Customizing Reports

Microsoft Access

1: Microsoft Access - Setting a Foundation

1.1.2: Microsoft Access- Main Features and Navigation

1.1.6: Maintaining Records in a Database

2: Microsoft Access - Building Proficiency

2.1.1: Microsoft Access and Data

2.1.3: Creating Forms in Microsoft Access

2.1.5: Customizing Forms

2.1.8: Customizing Reports

2.1.9: Using Macros

multiuser database

2.1.5: Customizing Forms

N

numeric entries

1.1.4: Creating a Table

O

organizing report

2.1.7: Creating Reports in Microsoft Access

P

primary key

1.1.5: Primary Keys and Table Relationships

programming

2.1.9: Using Macros

Q

queries

1: Microsoft Access - Setting a Foundation

1.1.3: Database Design and Four Main Objects

1.1.8: Querying a Database

1.1.9: All About Queries

2.1.1: Microsoft Access and Data

query design

1.1.9: All About Queries

2.1.2: Advanced Queries in Microsoft Access

2.1.6: Multiple Table Forms

R

records

1.1.7: Entering and Editing Records

relational database

1.1.1: Introduction to Microsoft Access

1.1.3: Database Design and Four Main Objects

1.1.5: Primary Keys and Table Relationships

1.1.8: Querying a Database

1.1.9: All About Queries

2.1.6: Multiple Table Forms

relational database management system

1.1.2: Microsoft Access- Main Features and Navigation

relational databases

1: Microsoft Access - Setting a Foundation

relational operators

1.1.4: Creating a Table

report design

2.1.8: Customizing Reports

report generation

2.1.9: Using Macros

report layout

2.1.7: Creating Reports in Microsoft Access

reports

1: Microsoft Access - Setting a Foundation

1.1.3: Database Design and Four Main Objects

2: Microsoft Access - Building Proficiency

required field

1.1.4: Creating a Table

retail operations

2.1.8: Customizing Reports

S

sales data

2.1.7: Creating Reports in Microsoft Access

sales records

1.1.8: Querying a Database

schema

1.1.3: Database Design and Four Main Objects

service records

2.1.6: Multiple Table Forms

sorting

1.1.7: Entering and Editing Records

split form

2.1.4: Form Basics

SQL

1.1.9: All About Queries

SQL queries

2.1.2: Advanced Queries in Microsoft Access

SQL View

1.1.8: Querying a Database

stakeholder engagement

2.1.8: Customizing Reports

T

table relationships

1: Microsoft Access - Setting a Foundation

1.1.5: Primary Keys and Table Relationships

tables

1.1.3: Database Design and Four Main Objects

U

unique index

1.1.4: Creating a Table

user data

1.1.8: Querying a Database

user experience

2.1.3: Creating Forms in Microsoft Access

2.1.5: Customizing Forms

user interface

2: Microsoft Access - Building Proficiency

2.1.3: Creating Forms in Microsoft Access

2.1.4: Form Basics

2.1.6: Multiple Table Forms

2.1.9: Using Macros

V

validation rule

1.1.4: Creating a Table

VBA

2.1.9: Using Macros

vehicle maintenance

2.1.6: Multiple Table Forms

Detailed Licensing

Some of the pages in this book are re-mixes of open content from other books in the LibreTexts Library. Sometimes sections from several sources are used on the same page and there may be no unique open license. Where there is a conflict, the licensing may appear as *undeclared*. On any page, by toggling AutoAttribution on the Tools menu, you can display the detailed attribution on a paragraph level. Material from different books will be color coded and the source displayed by mousing over the paragraph.

Content in the Front and Back Matter, directories and similar can be uncopywriteable, being simple listings and these are also shown as *undeclared*.

Overview

Title: CPSI10003 Microcomputer Applications 1 - Microsoft Access

Webpages: 34

Applicable Restrictions: Noncommercial

All licenses found:

- [CC BY-NC-SA 4.0](#): 70.6% (24 pages)
- [Undeclared](#): 26.5% (9 pages)
- [Other](#): 2.9% (1 page)

By Page

- [CPSI10003 Microcomputer Applications 1 - Microsoft Access - CC BY-NC-SA 4.0](#)
 - [Front Matter - Undeclared](#)
 - [TitlePage - Undeclared](#)
 - [InfoPage - Other](#)
 - [Table of Contents - Undeclared](#)
 - [Licensing - Undeclared](#)
 - [1: Microsoft Access - Setting a Foundation - CC BY-NC-SA 4.0](#)
 - [1.1: Microsoft Access - Setting a Foundation Sections - CC BY-NC-SA 4.0](#)
 - [1.1.1: Introduction to Microsoft Access - CC BY-NC-SA 4.0](#)
 - [1.1.2: Microsoft Access- Main Features and Navigation - CC BY-NC-SA 4.0](#)
 - [1.1.3: Database Design and Four Main Objects - CC BY-NC-SA 4.0](#)
 - [1.1.4: Creating a Table - CC BY-NC-SA 4.0](#)
 - [1.1.5: Primary Keys and Table Relationships - CC BY-NC-SA 4.0](#)
 - [1.1.6: Maintaining Records in a Database - CC BY-NC-SA 4.0](#)
 - [1.1.7: Entering and Editing Records - CC BY-NC-SA 4.0](#)
 - [1.1.8: Querying a Database - CC BY-NC-SA 4.0](#)
 - [1.1.9: All About Queries - CC BY-NC-SA 4.0](#)
- [2: Microsoft Access - Building Proficiency - CC BY-NC-SA 4.0](#)
 - [2.1: Microsoft Access - Building Proficiency Sections - CC BY-NC-SA 4.0](#)
 - [2.1.1: Microsoft Access and Data - CC BY-NC-SA 4.0](#)
 - [2.1.2: Advanced Queries in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.3: Creating Forms in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.4: Form Basics - CC BY-NC-SA 4.0](#)
 - [2.1.5: Customizing Forms - CC BY-NC-SA 4.0](#)
 - [2.1.6: Multiple Table Forms - CC BY-NC-SA 4.0](#)
 - [2.1.7: Creating Reports in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.8: Customizing Reports - CC BY-NC-SA 4.0](#)
 - [2.1.9: Using Macros - CC BY-NC-SA 4.0](#)
 - [2.1.10: Microsoft Access Terminology - CC BY-NC-SA 4.0](#)
- [Back Matter - Undeclared](#)
 - [Index - Undeclared](#)
 - [Glossary - Undeclared](#)
 - [Detailed Licensing - Undeclared](#)
 - [Detailed Licensing - Undeclared](#)

Detailed Licensing

Overview

Title: CPSI10003 Microcomputer Applications 1 - Microsoft Access

Webpages: 33

Applicable Restrictions: Noncommercial

All licenses found:

- [CC BY-NC-SA 4.0](#): 72.7% (24 pages)
- [Undeclared](#): 24.2% (8 pages)
- [Other](#): 3% (1 page)

By Page

- [CPSI10003 Microcomputer Applications 1 - Microsoft Access - CC BY-NC-SA 4.0](#)
 - [Front Matter - Undeclared](#)
 - [TitlePage - Undeclared](#)
 - [InfoPage - Other](#)
 - [Table of Contents - Undeclared](#)
 - [Licensing - Undeclared](#)
 - [1: Microsoft Access - Setting a Foundation - CC BY-NC-SA 4.0](#)
 - [1.1: Microsoft Access - Setting a Foundation Sections - CC BY-NC-SA 4.0](#)
 - [1.1.1: Introduction to Microsoft Access - CC BY-NC-SA 4.0](#)
 - [1.1.2: Microsoft Access- Main Features and Navigation - CC BY-NC-SA 4.0](#)
 - [1.1.3: Database Design and Four Main Objects - CC BY-NC-SA 4.0](#)
 - [1.1.4: Creating a Table - CC BY-NC-SA 4.0](#)
 - [1.1.5: Primary Keys and Table Relationships - CC BY-NC-SA 4.0](#)
 - [1.1.6: Maintaining Records in a Database - CC BY-NC-SA 4.0](#)
 - [1.1.7: Entering and Editing Records - CC BY-NC-SA 4.0](#)
 - [1.1.8: Querying a Database - CC BY-NC-SA 4.0](#)
 - [1.1.9: All About Queries - CC BY-NC-SA 4.0](#)
 - [2: Microsoft Access - Building Proficiency - CC BY-NC-SA 4.0](#)
 - [2.1: Microsoft Access - Building Proficiency Sections - CC BY-NC-SA 4.0](#)
 - [2.1.1: Microsoft Access and Data - CC BY-NC-SA 4.0](#)
 - [2.1.2: Advanced Queries in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.3: Creating Forms in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.4: Form Basics - CC BY-NC-SA 4.0](#)
 - [2.1.5: Customizing Forms - CC BY-NC-SA 4.0](#)
 - [2.1.6: Multiple Table Forms - CC BY-NC-SA 4.0](#)
 - [2.1.7: Creating Reports in Microsoft Access - CC BY-NC-SA 4.0](#)
 - [2.1.8: Customizing Reports - CC BY-NC-SA 4.0](#)
 - [2.1.9: Using Macros - CC BY-NC-SA 4.0](#)
 - [2.1.10: Microsoft Access Terminology - CC BY-NC-SA 4.0](#)
 - [Back Matter - Undeclared](#)
 - [Index - Undeclared](#)
 - [Glossary - Undeclared](#)
 - [Detailed Licensing - Undeclared](#)