

CHAPTER-1 INTRODUCTION TO DATABASES

Introduction

Microsoft Access 2013 is a **database creation and management** program. To understand Access, you must first understand **databases**.

You will learn about databases and how they are used. You will familiarize yourself with the differences between data management in **Access** and **Microsoft Excel**. Finally, you will get a look ahead at the rest of the Access Chapters.

What is a database?

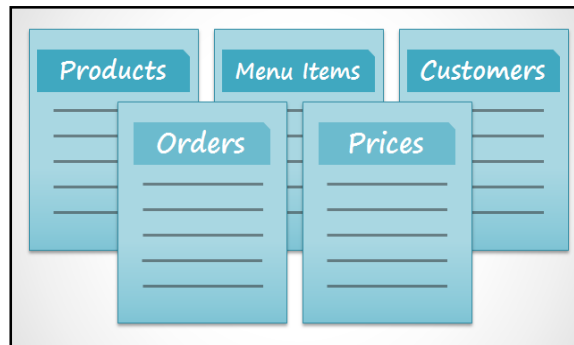
A **database** is a collection of data that is stored in a computer system. Databases allow their users to **enter, access, and analyze** their data quickly and easily. They're such a useful tool that you see them all the time. Ever waited while a doctor's receptionist entered your personal information into a computer, or watched a store employee use a computer to see whether an item was in stock? Then you've seen a database in action.

The easiest way to understand a database is to think of it as a **collection of lists**. Think about one of the databases we mentioned above: the database of patient information at a doctor's office. What lists are contained in a database like this? Well, to start with, there's a list of patients' names. Then there's a list of past appointments, a list with medical history for each patient, a list of contact information, and so on.

This is true of all databases, from the simplest to the most complex. For instance, if you like to bake you might decide to keep a database containing the types of cookies you know how to make and the friends you give these cookies to. This is one of the simplest databases imaginable. It contains two lists: a list of your friends, and a list of cookies.



However, if you were a professional baker, you would have many more lists to keep track of: a list of customers, a list of products sold, a list of prices, a list of orders, and so on. The more lists you add, the more **complex** the database will be.



In Access, lists are a little more complex than the ones you write on paper. Access stores its lists of data in **tables**, which allow you to store even more detailed information. In the table below, the **People** list in the amateur baker's database has been expanded to include other relevant information on the baker's friends.

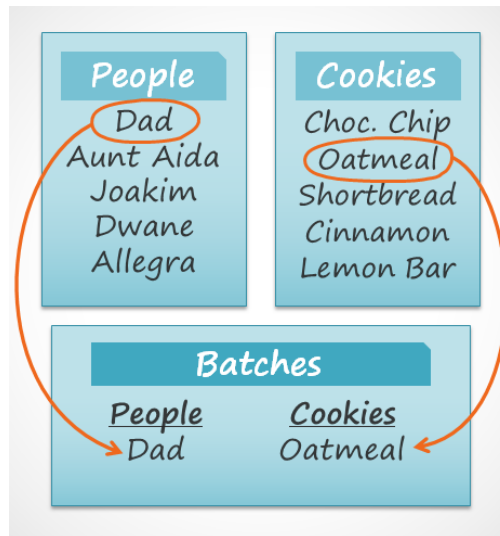
If you are familiar with other programs in the Microsoft Office suite, this might remind you of Excel, which allows you to organize data in a similar way. In fact, you could build a similar table in Excel.

ID	Name	Cell Phone	Birthday	Nut Allergy?
1	Dad	555-0404	June 3	Yes
2	Aunt Aida	555-9890	July 8	No
3	Joakim	555-0462	September 19	No
4	Dwane	555-9975	January 5	No
5	Allegra	555-0099	January 14	Yes

Why use a database?

If a database is essentially a collection of lists stored in tables and you can build tables in Excel, why do you need a real database in the first place? While Excel is great at storing and organizing numbers, Access is far stronger at handling **non-numerical data**, like names and descriptions. Non-numerical data plays a significant role in almost any database, and it's important to be able to sort and analyze it.

However, the thing that really sets databases apart from any other way of storing data is **connectivity**. We call a database like the ones you'll work with in Access a **relational database**. A relational database is able to understand how lists and the objects within them **relate** to one another. To explore this idea, let's go back to the simple database with two lists: names of your friends, and the types of cookies you know how to make. You decide to create a third list to keep track of the batches of cookies you make and who they're for. Because you're only making cookies you know the recipe for and you're only going to give them to your friends, this new list will get all of its information from the lists you made earlier.



See how the third list uses words that appeared in the first two lists? A database is capable of understanding that the **Dad** and **Oatmeal** cookies in the **Batches** list are the same things as the **Dad** and **Oatmeal** in the first two lists. This relationship seems obvious, and a person would understand it right away. However, an Excel workbook wouldn't.

Excel would treat all of these things as distinct and unrelated pieces of information. In Excel, you'd have to enter every single piece of information about a person or type of cookie each time you mentioned it because that database wouldn't be **relational** like an Access database. Simply put, relational databases can recognize what a human can: If the same words appear in multiple lists, they **refer** to the same thing.

The fact that relational databases can handle information this way allows you to **enter**, **search for**, and **analyze** data in more than one table at a time. All of these things would be difficult to accomplish in Excel, but in Access even complicated tasks can be simplified and made fairly user friendly.

CHAPTER-2 INTRODUCTION TO OBJECTS

Introduction

Databases in Access are composed of four objects: **tables**, **queries**, **forms**, and **reports**. Together, these objects allow you to enter, store, analyze, and compile your data however you want.

In this lesson, you will learn about each of the four **objects** and come to understand how they interact with each other to create a fully functional relational database.

Tables

By this point, you should already understand that a database is a collection of data organized into many connected **lists**. In Access, all data is stored in **tables**, which puts tables at the heart of any database.

You might already know that tables are organized into vertical **columns** and horizontal **rows**.

ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrold	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St.	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC
11	Katy	Jones	456 Denver Rd.	Cary	NC
12	Beatrix	Joslin	85 North West St.	Raleigh	NC
13	Mariah	Allen	12 Jupe	Raleigh	NC
14	Jennifer	Hill	2100 Field Ave.	Raleigh	NC
15	Jaleel	Smith	123 Hill Top Drive	Garner	NC

In Access, rows and columns are referred to as **records** and **fields**. A **field** is more than just a column; it's a way of organizing information by the **type** of data it is. Every piece of information within a field is of the same **type**. For example, every entry in a field called **First Name** would be a name, and every entry in field called **Street Address** would be an address.

ID	First Name	Last Name	Street Address
1	Tracey	Beckham	7 East Walker Dr.
2	Lucinda	George	789 Brewer St.
3	Jerrold	Smith	211 St. George Ave.
4	Brett	Newkirk	47 Hillsborough St.
5	Chloe	Jones	23 Solo Ln.
6	Quinton	Boyd	4 Cypress Cr.
7	Alex	Hinton	1011 Hodge Ln.
8	Nisha	Hall	123 Huntington St.
9	Hillary	Clayton	2516 Newman
10	Kiara	Williams	9014 Miller Ln.
11	Katy	Jones	456 Denver Rd.
12	Beatrix	Joslin	85 North West St.

Likewise, a **record** is more than just a row; it's a unit of information. Every cell in a given row is part of that row's record.

ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrold	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St.	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC
11	Katy	Jones	456 Denver Rd.	Cary	NC

Notice how each record spans several fields. Even though the information in each record is organized into fields, it belongs with the other information in that record. See the **number** at the left of each row? It's the **ID number** that identifies each record. The ID number for a record refers to every piece of information contained on that row.

ID	First Name	Last Name
40	Vig	Aurelio
41	Jeffery	Bergman
42	William	Bittiman
43	Megan	
44		
45	Marjan	
46	Colin	Hopkins
47	Hakim	Auden
48	Pilar	Semana
49	Eliza	Harris
50	Chloe	Ford
51	Juanita	Harris

Tables are good for storing **closely related information**. Let's say you own a bakery and have a database that includes a table with your customers' names and information, like their phone numbers, home addresses, and email addresses. Because these pieces of information are all details on your customers, you'd include them all in the same **table**. Each customer would be represented by a unique **record**, and each type of information about these customers would be stored in its own field. If you decided to add any more information—say, a customer's birthday—you would simply create a new field within the same table.

Forms, queries, and reports

Although tables store all of your data, the other three objects—**forms**, **queries**, and **reports**—offer you ways to work with it. Each of these objects interacts with the **records** stored in your database's tables.

Forms

Forms are used for **entering**, **modifying**, and **viewing** records. You likely have had to fill out forms on many occasions, like when visiting a doctor's office, applying for a job, or registering for school. The reason forms are used so often is that they're an easy way to guide people toward entering data correctly. When you enter information into a form in Access, the data goes exactly where the database designer wants it to go: in one or more related tables.

Forms make entering data easier. Working with extensive tables can be confusing, and when you have connected tables you might need to work with more than one at a time to enter a set of data. However, with forms it's possible to enter data into multiple tables at once, all in one place. Database designers can even set restrictions on individual form components to ensure all of the needed data is entered in the correct format. All in all, forms help keep data consistent and organized, which is essential for an accurate and powerful database.

Queries

Queries are a way of **searching** for and **compiling** data from one or more tables. Running a query is like asking a detailed **question** of your database. When you build a query in Access, you are **defining specific search conditions** to find exactly the data you want.

Queries are far more powerful than the simple searches you might carry out within a table. While a **search** would be able to help you find the name of one customer at your business, you could run a **query** to find the name and phone number of every customer who's made a purchase within the past week. A well-designed query can give information you might not be able to find just by looking through the data in your tables.

Field:	First Name	Last Name	Pickup Date	
Table:	Customers	Customers	Orders Table	
Total:	Group By	Group By	Count	
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:				
or:				

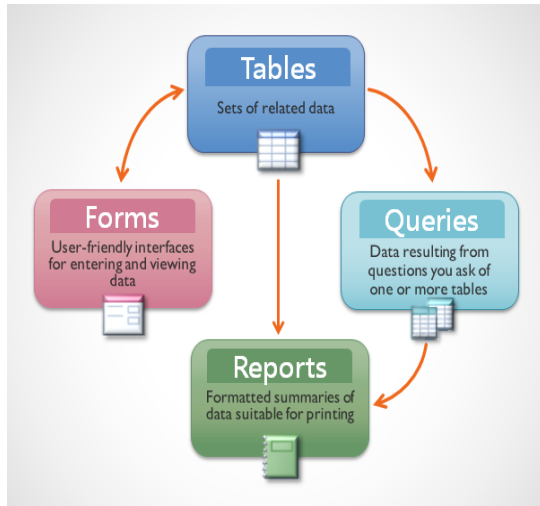
Reports

Reports offer you the ability to **present** your data in **print**. If you've ever received a computer printout of a class schedule or a printed invoice of a purchase, you've seen a database report. Reports are useful because they allow you to present components of your database in an easy-to-read format. You can even customize a report's appearance to make it visually appealing. Access offers you the ability to create a report from any **table** or **query**.

First Name	Last Name	Phone Number	Pickup Date
Andrzej	Wujek	919-555-0450	12/4/2010
Dick	Whitman	919-555-5042	12/4/2010
Xy'nya	Bell	919-555-0758	12/9/2010
Xiaoxi	Zheng	919-555-2786	12/10/2010
Hakim	Auden	919-555-0045	12/14/2010
Zoey	Altman	919-555-6688	12/15/2010
Raphaelle	Duvalier	919-555-1547	12/15/2010
Zoey	Altman	919-555-6688	12/16/2010

Putting it all together

Even if you have a good idea of how each object can be used, it can initially be difficult to understand how they all work together. It helps to remember that they all work with the same data. Every piece of data a **query**, **form**, or **report** uses is stored in one of your database **tables**.



Forms allow you to both **add** data to tables and **view** data that already exists. Reports **present** data from tables and also from queries, which then **search for** and **analyze** data within these same tables.

These relationships sound complicated, but in fact they work together so well and naturally that we often don't even notice when we're using connected database objects. Have you ever used an electronic card catalog to search for a book at the library? Chances are, you entered your search in something that looks like this:

Search the library catalog. Fill in at least one field. The more words you search for, the smaller and more refined your results list will be.

Search by:

Title Keyword AND

Author Keyword AND

General Keyword AND

Subject Keyword

Refine your search (optional): You can refine your search by selecting a limit or sort option.

Limits

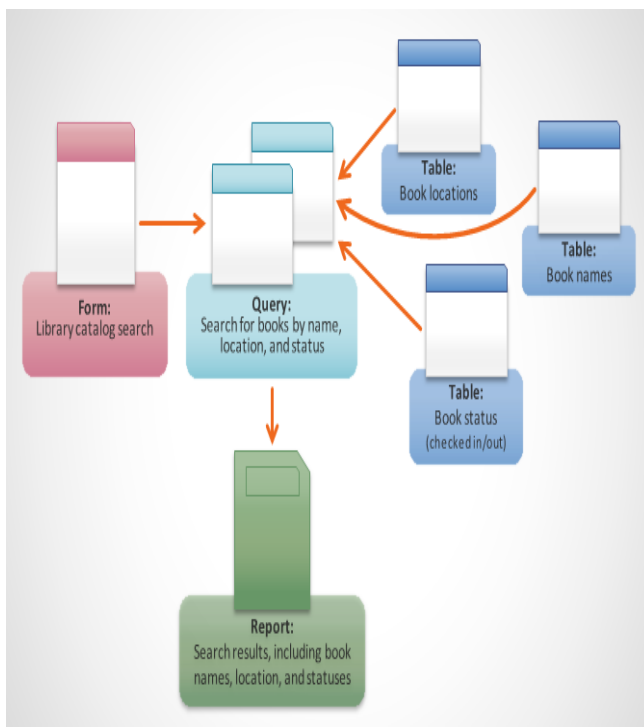
Library Branch	Type of Book
Athens Drive Community Library	All Audio Books
Cameron Village Regional Library	Downloadable Audio Books
Cary Public Library	Adult Audio Fiction
Duraleigh Road Library	Adult Audio Non Fiction
East Regional Library	Children's Audio

Sorting

Sort your results by selecting a sort option.

Select...

When you performed your search, you were entering your search terms into a **form** that then created and ran a **query** based on your request. When the query finished searching the database's **tables** for records that matched your search, you were shown a **report** that drew information from the query and the related tables—in this case, a list of books matching your search terms. You could represent the connections between the objects like this:



Let's say instead of using these tools you had to search within a giant table containing every book in the library system. The relevant records would likely be spread out across many tables: a table for book titles and descriptions, a table containing information on which books are checked in or out, and a table with each branch of the library, just to name a few.

You'd have to search at least three tables just to find a book, learn its location, and see whether it's checked in! It's easy to imagine how difficult it could become to find the right book. If you weren't careful, you might even mess something up by accidentally deleting or editing a record. It's easy to see how the database objects make this search much more manageable.

In our **Introduction to Databases Chapter**, we discussed the concept of a **relational database**, which is a database that is able to understand how different sets of data **relate** to one another. Situations like the example above are exactly why people find relational databases so useful. Without a relational database, what should be a simple task—searching for a book and seeing if it's checked in and where—becomes incredibly complicated and time consuming. Knowing how to use the four Access objects can make even complicated tasks fairly user friendly.

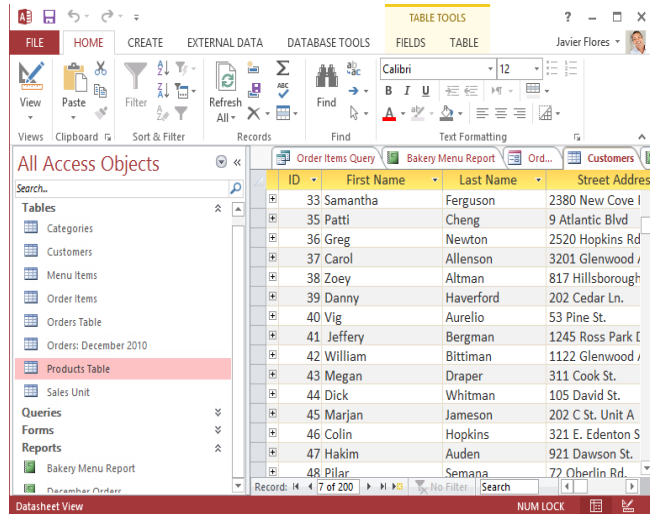
CHAPTER-3 GETTING STARTED WITH ACCESS

Introduction

Whenever you're learning a new program, it's important to familiarize yourself with the program window and the tools within it. Working with Access is no different. Knowing your way around the Access environment will make learning and using Access much easier.

In this Chapter, you will familiarize yourself with the Access environment, including the **Ribbon**, **Backstage view**, **Navigation pane**, **Document Tabs bar**, and more. You will also learn how to navigate with a **navigation form**, if your database includes one.

Access 2013 uses the **Ribbon** to organize commands, just like in Access 2010 and 2007. If you've used these versions before, Access 2013 will feel familiar. But if you are new to Access or have more experience with older versions, you should first take some time to become familiar with the **Access 2013 interface**.

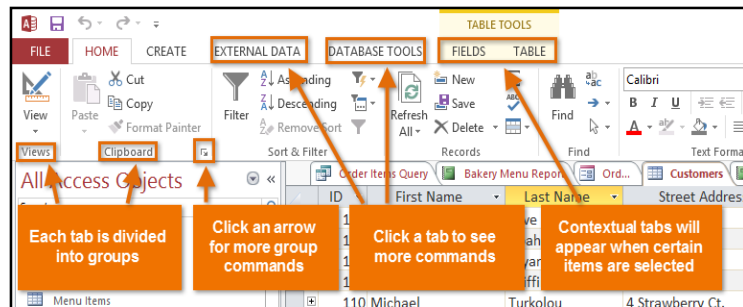


Working with your Access environment

If you've previously used Access 2010 or 2007, Access 2013 will feel familiar. It continues to use features like the **Ribbon** and the **Quick Access toolbar**—where you will find commands to perform common tasks in Access—as well as **Backstage view**.

The Ribbon

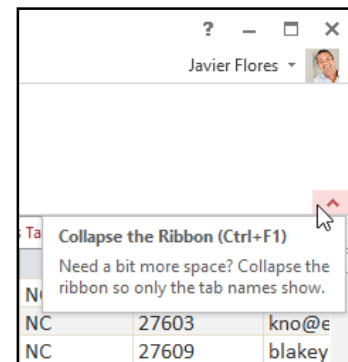
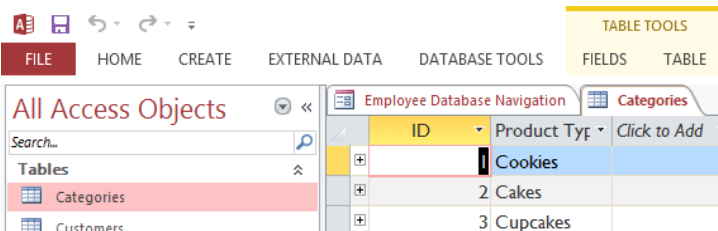
Access 2013 uses a **tabbed Ribbon system** instead of traditional menus. The **Ribbon** contains **multiple tabs**, each with several **groups of commands**. You will use these tabs to perform the most **common tasks** in Access.



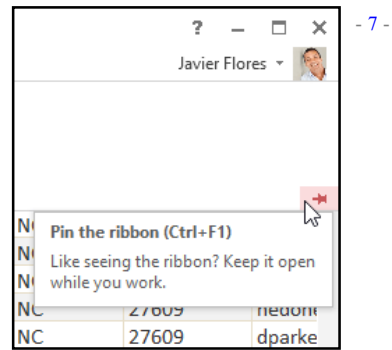
To minimize and maximize the Ribbon:

The Ribbon is designed to respond to your current task; however, you can choose to **minimize** the Ribbon if you find that it takes up too much screen space.

1. Click the **arrow** in the lower-right corner of the Ribbon to minimize it.
2. The **Ribbon** will be minimized. Click a **tab** to make the Ribbon reappear. It will disappear again when not in use.

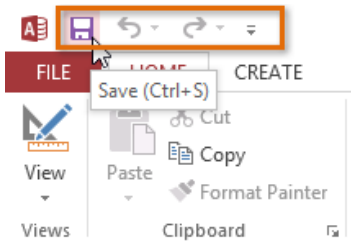


- To **maximize** the Ribbon, click a tab, then click the **pin** icon in the lower-right corner. The Ribbon will appear at all times.



The Quick Access toolbar

The **Quick Access toolbar**, located above the Ribbon, lets you access common commands no matter which tab you are on. By default, it shows the **Save**, **Undo**, and **Repeat** commands. If you'd like, you can **customize** it by **adding additional commands**.



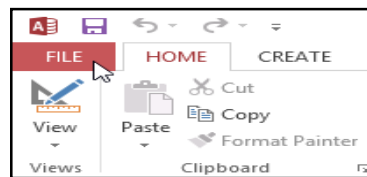
Note that the **Save** command only saves the current open object. In addition, the **Undo** command will not undo certain actions, like adding a record. Pay close attention to your information when using the **Undo** command to make sure it has the desired effect.

Backstage view

Backstage view gives you various options for saving, opening, and printing your database.

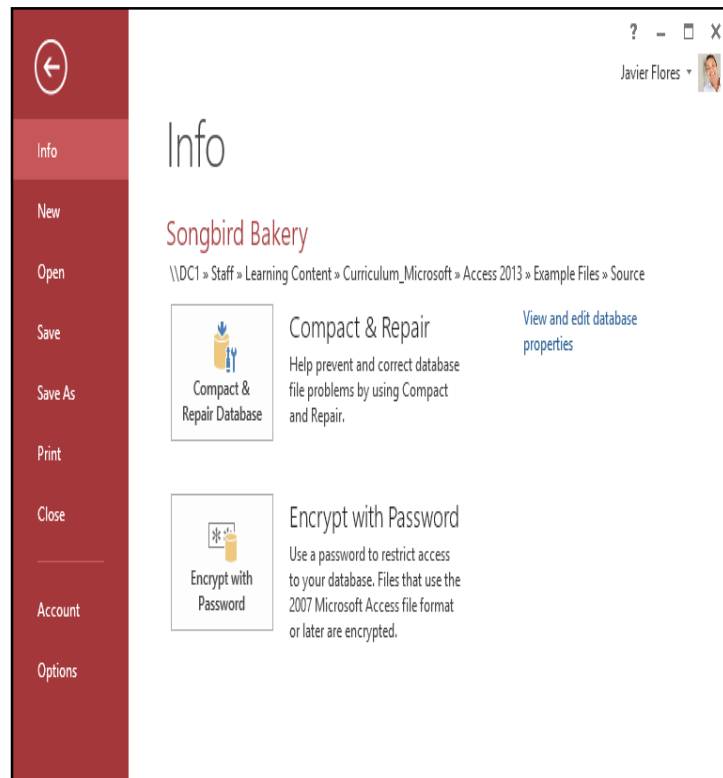
To access Backstage view:

- Click the **File** tab on the **Ribbon**.



- Backstage view** will appear.

See below Backstage view-



The Navigation pane

The **Navigation pane** is a list containing every object in your database. For easier viewing, the objects are organized into groups by type. You can **open**, **rename**, and **delete** objects using the Navigation pane.

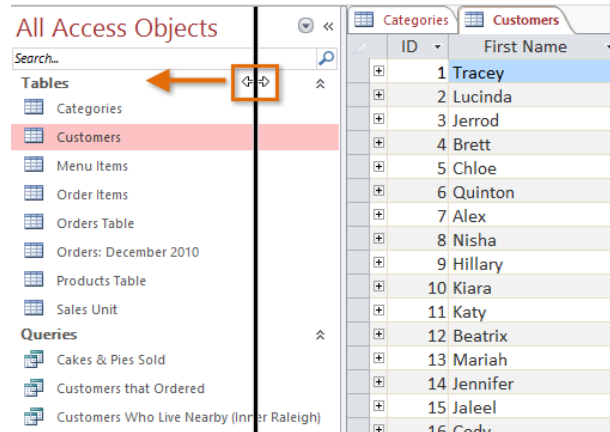
To minimize and maximize the Navigation pane:

The Navigation pane is designed to help you manage all of your objects; however, if you feel that it takes up too much of your screen space you can **minimize** it.

1. To minimize the Navigation pane, click the **double arrow** in the upper-right corner.



2. The Navigation pane will be minimized. Click the **double arrow** again to maximize it.



If you want to make the Navigation pane smaller without fully minimizing it, you can **resize** it. Simply click and drag the right border of the Navigation pane. When it is the desired size, release your mouse.

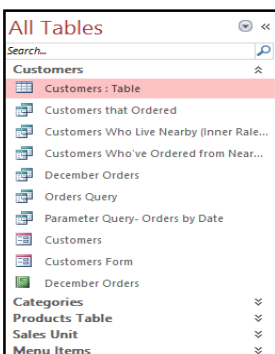
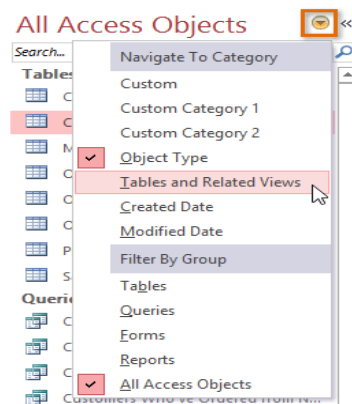
Object sorting in the Navigation pane

By default, objects are sorted by **type**, with tables in one group, forms in another, and so on. However, if you want you can sort the objects in the Navigation pane into groups of your choosing. There are four sort options:

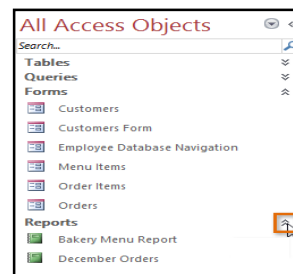
- **Custom** allows you to create a custom group for sorting objects. After applying the sort, simply drag the desired objects to the new group.
- **Object Type** groups objects by type. This is the default setting.
- **Tables and Related Views** groups forms, queries, and reports with the tables they refer to.
- **Created Date** or **Modified Date** sorts objects based on when they were created or last edited.

To sort objects in the Navigation pane:

1. Click the **drop-down arrow** to the right of **All Access Objects**, then select the desired sort from the drop-down menu.
2. The objects in the Navigation pane will now be sorted to reflect your choice.



To further customize the appearance of the Navigation pane, you can **minimize groups** of objects you don't want to see. Simply click the **upward double arrow** next to the group name. To show a group, click the **downward double arrow**.



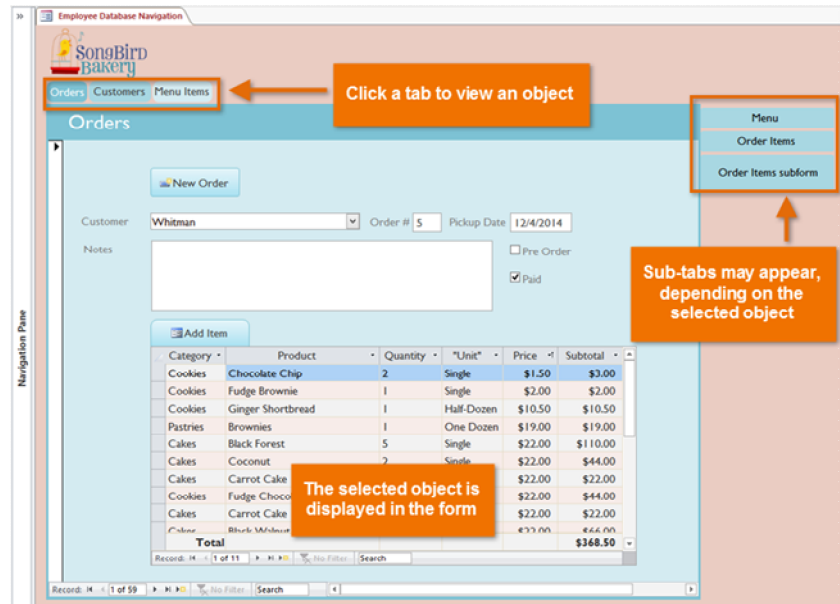
Databases with navigation forms

Some databases include a **navigation form** that opens automatically when the database is opened. Navigation forms are designed to be a **user-friendly** replacement for the **Navigation pane**. They contain **tabs** that allow you to view and work with common forms, queries, and reports. Having your frequently used objects available to you in one place lets you access them quickly and easily.

To open an object from a navigation form, click its tab. The object will be displayed within the navigation form. Once an object is open, you can work with it as you normally would.

Generally, navigation forms include only the objects a typical user will need to work with fairly regularly, which is why your navigation form may not include every single form, query, or report. This makes it easier to navigate the database. By hiding tables and rarely used forms, queries, and reports, it also reduces the chance of the database being damaged by users accidentally editing or deleting necessary data.

For this reason, it's important to ask your database designer or administrator before working with objects that are not available in your navigation form. Once you have the go-ahead, you can simply **maximize** the **Navigation pane** and open the objects from there.



CHAPTER-4 MANAGING DATABASES AND OBJECTS

Introduction

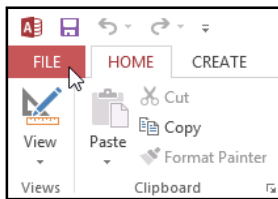
Each Access database consists of multiple **objects** that let you interact with data. Databases can include **forms** for entering data, **queries** for searching within it, **reports** for analyzing it, and **tables** for storing it. Whenever you work with your database, you are working with many of these objects at once. Fortunately, Access makes managing these objects pretty easy.

In this lesson, you will learn how to **open** and **close** databases, as well as how to **open**, **close**, and **save** objects.

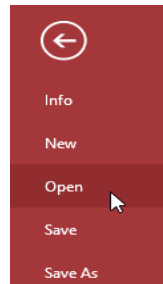
To open an existing database:

Before you enter data or modify your objects, you will need to open your database.

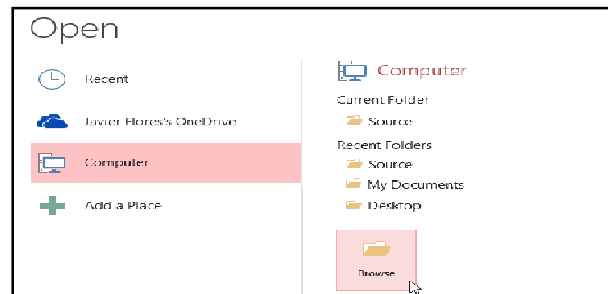
1. Select the **File** tab to go to **Backstage view**.



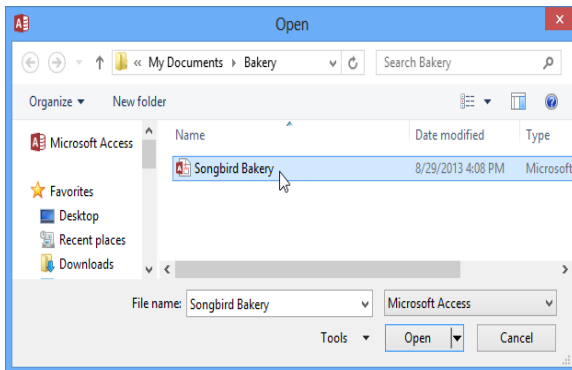
2. Click **Open**.



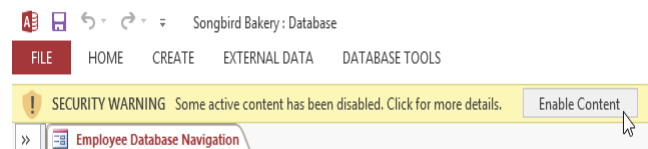
3. Select **Computer**, then click **Browse**.



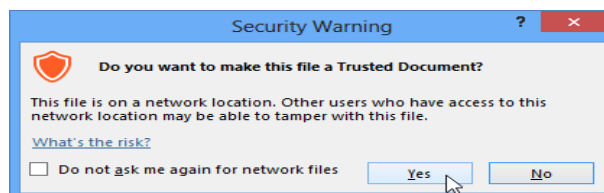
4. The **Open** dialog box will appear. Locate and select the database, then click **Open**.



5. One or more **warning messages** may appear when you open your database. If the database contains customized functions, a yellow bar with a security warning may appear below the Ribbon. If you trust the source of your database, click **Enable Content** for your database to display correctly.



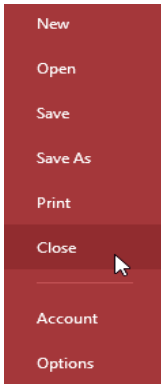
6. After enabling all content in the database, you may see a message asking if you want to make the database a **Trusted Document**. Click **Yes** if you would like all content to be automatically enabled each time you open the database.



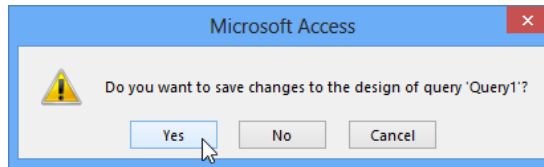
You may also be prompted to **sign in** to the database. Select your name from the login list. If your name does not appear, click **Add User** to enter your information.

To close a database:

1. Select the **File** tab to go to **Backstage view**.
2. Select **Close**.



3. If you have any unsaved objects, a dialog box will appear for each one asking if you would like to save it. Select **Yes** to save the object, **No** to close it without saving, or **Cancel** to leave your database open.



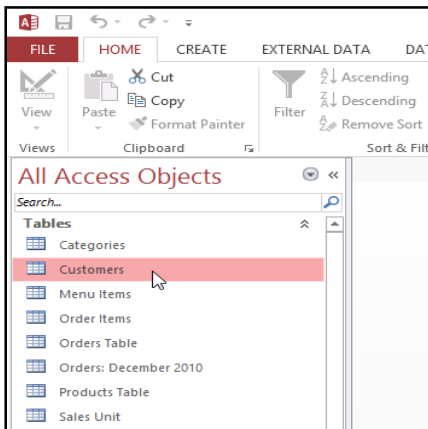
Working with objects

It's helpful to think of your database as a large binder or folder in which you store your data. The data itself is contained in database **objects**. Access treats each of these objects as separate documents, which means you will have to **open** and **save** them individually in order to work with them.

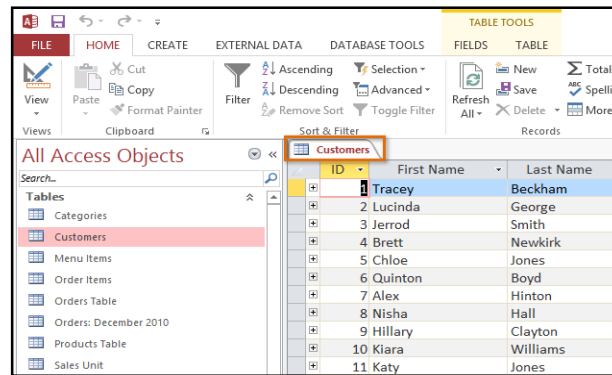
You may have noticed that this lesson contains no instructions for saving a database. This is because you cannot save an entire database at once. Rather, you must individually save the objects contained within the database.

To open an object:

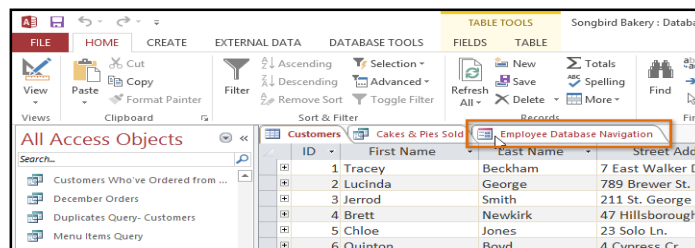
1. In the **Navigation pane**, locate and double-click the desired object.



2. The object will appear as a **tab** in the **Document Tabs bar**.

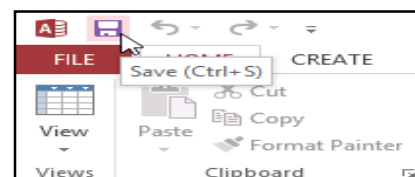


By default, the most recently opened object will appear in the main window as the **current object**. To view another open object, click its tab in the **Document Tabs bar**.



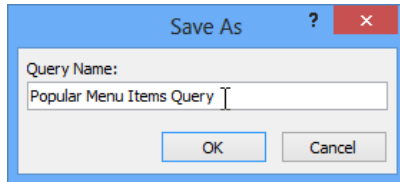
Saving objects

You'll need to **save** any changes you make to each object before closing your database. Remember, saving early and often can prevent your work from being lost. However, you will also be prompted to save any unsaved work when you attempt to close your database.



To save a new object:

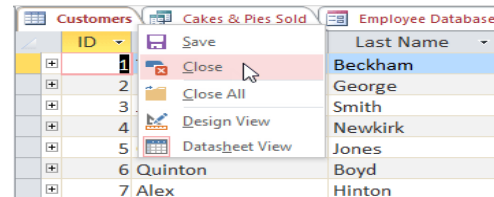
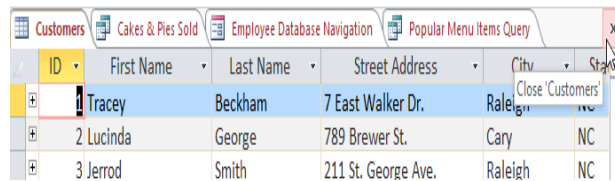
1. Select the object you want to save by clicking its tab in the **Document Tabs bar**.
2. Click the **Save** command on the **Quick Access toolbar**, or press **Ctrl+S** on your keyboard.
3. The first time you save an object, you will be prompted to name it. Enter the desired object name, then click **OK**.



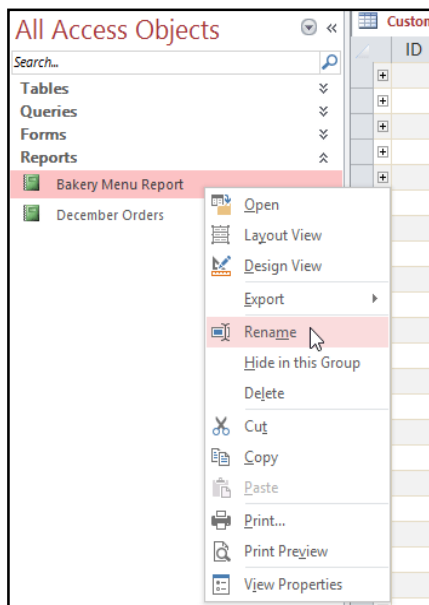
4. The object will be saved. Click the **Save** command again to save any changes to the object.

To close an object:

1. Select the object you want to close, then click the **X** on the right of the **Document Tabs bar**.
2. If there are any unsaved changes to the object, you will be prompted to save it. Select **Yes** to save, **No** to close it without saving your changes, and **Cancel** to leave the object open.



You can also close an object by right-clicking its tab on the Document Tabs bar and selecting **Close**. Select **Close All** to close all open objects.

**To rename an object:**

1. If the object you want to rename is open, **close** it.
2. In the **Navigation pane**, **right-click** the desired object, then select **Rename**.
3. Type the new object name, then press **Enter** on your keyboard.



CHAPTER-5 WORKING WITH TABLES


Introduction

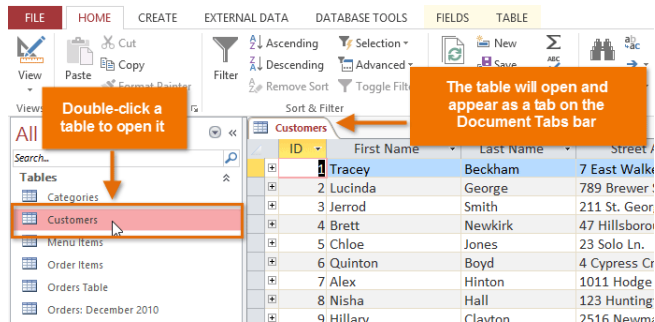
While there are four types of database objects in Access, **tables** are arguably the most important. Even when you're using forms, queries, and reports, you're still working with tables because that's where all of your **data** is stored. Tables are at the heart of any database, so it's important to understand how to use them.

In this Chapter, you will learn how to **open tables**, **create** and **edit records**, and **modify the appearance** of your table to make it easier to view and work with.

Table basics

To open an existing table:

1. Open your database, and locate the **Navigation pane**.
2. In the Navigation pane, locate the table you want to open. Tables are marked with the  icon.
3. Double-click the desired table. It will open and appear as a **tab** in the **Document Tabs bar**.



Understanding tables

All tables are composed of horizontal **rows** and vertical **columns**, with small rectangles called **cells** in the places where rows and columns intersect. In Access, rows and columns are referred to as **records** and **fields**.

ID	First Name	Last Name	Street Address	City	State
52	Denver	Emory	856 Cook St.	Raleigh	NC
53	John	Emory	99 Hillsborough St.	Garner	NC
54	Ebony	Farmer	872 W. Morgan St.	Raleigh	NC
55	Kim	Doe	553 Wayne St.	Raleigh	NC
56	Coretta	Diaz	781 Bloodsworth St.	Raleigh	NC
57	Victor	Denver	31 St. Mary's St.	Raleigh	NC
58	Hamish	David	21 Cameron Ct.	Raleigh	NC
59	Erin	Counts	763 E. West St.	Raleigh	NC
60	Harris	Collman	455 E. Edenton St.	Raleigh	NC
61	Melissa	Chantay	3201 Glenwood Ave.	Raleigh	NC
62	Emmanual	Colin	11 Sassafras Way	Raleigh	NC
63	Illa	Carlson	563 Oberlin Rd.	Raleigh	NC
64	Tyrone	Rodgers	453 Pine St.	Raleigh	NC

A **field** is a way of organizing information by type. Think of the **field name** as a question and every cell within that field as a response to that question.

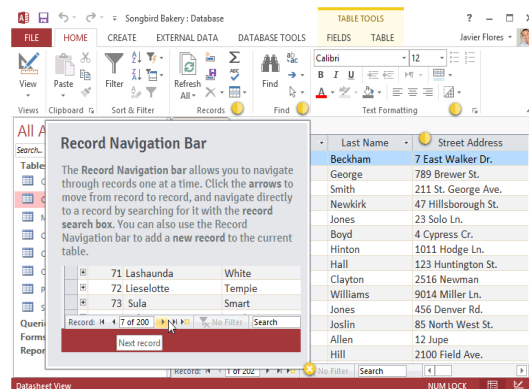
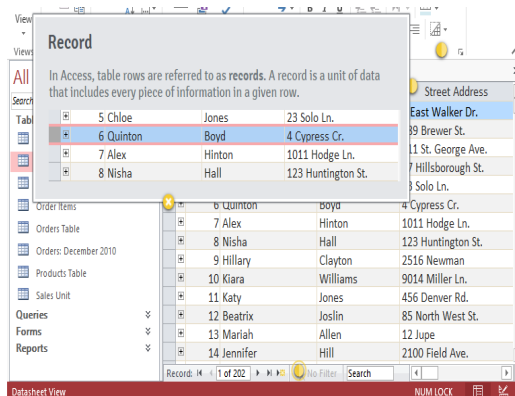
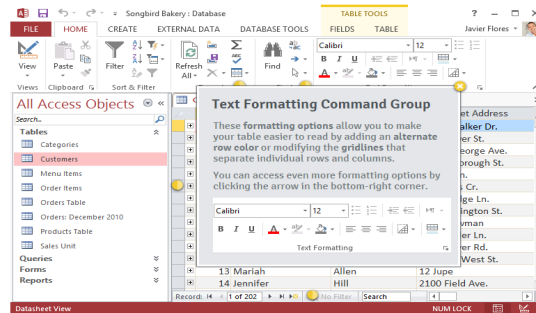
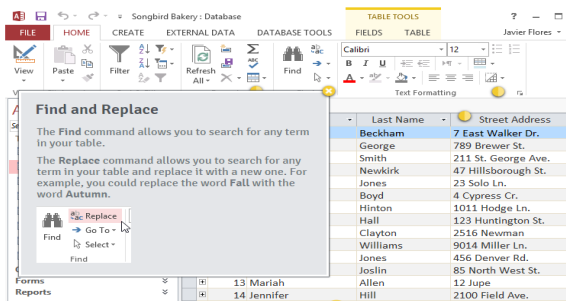
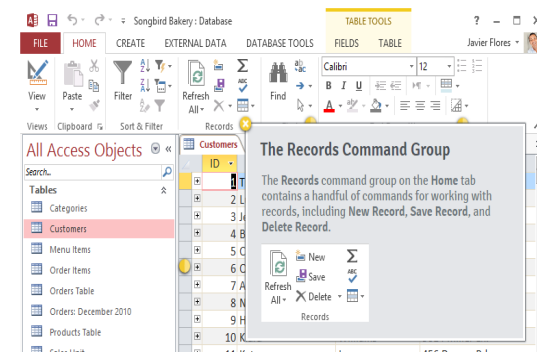
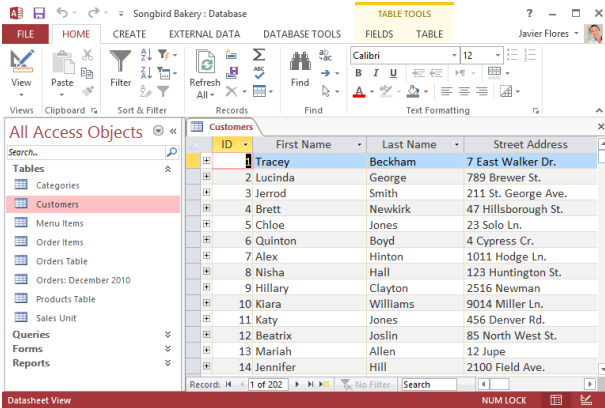
ID	First Name	Last Name	Street Address
1	Tracey	Beckha	7 East Walker Dr.
2	Lucinda	George	789 Brewer St.
3	Jerrold	Smith	211 St. George Ave.
4	Brett	Newkirk	47 Hillsborough St.
5	Chloe	Jones	23 Solo Ln.
6	Quinton	Boyd	4 Cypress Cr.
7	Alex	Hinton	1011 Hodge Ln.
8	Nisha	Hall	123 Huntington St.
9	Hillary	Clayton	2516 Newman
10	Kiara	Williams	9014 Miller Ln.
11	Katy	Jones	456 Denver Rd.
12	Beatrix	Joslin	85 North West St.

A **record** is one unit of information. Every cell on a given row is part of that row's record. Each record has its own **ID number**. Within a table, each ID number is unique to its record and refers to all of the information within that record. The ID number for a record cannot be changed.

ID	First Name	Last Name
40	Vig	Aurelio
41	Jeffery	Bergman
42	William	Bittiman
43	Megan	Harris
44	Marjan	Hopkins
45	Marjan	Hopkins
46	Colin	Hopkins
47	Hakim	Auden
48	Pilar	Semana
49	Eliza	Harris
50	Chloe	Ford
51	Juanita	Harris

Each cell of data in your table is part of both a **field** and a **record**. For instance, if you had a table of names and contact information, each person would be represented by a record, and each piece of information about each person—name, phone number, address, and so on—would be contained within a distinct field on that record's row.

See below to learn how to navigate a table-



Navigating within tables

To navigate through records in a table, you can use the **up and down arrow keys**, **scroll up and down**, or use the arrows in the **Record Navigation bar** located at the bottom of your table. You can also find any record in the currently open table by **searching** for it using the **record search box**. Simply place your cursor in the search box, type any word that appears in the record you want to find, and press the **Enter** key. To view additional records that match your search, press **Enter** again.

To navigate between fields, you can use the **left and right arrow keys** or **scroll left and right**.

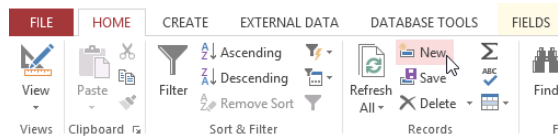
Adding records and entering data

Entering data into tables in Access is similar to entering data in Excel. To work with records, you'll have to enter data into **cells**.

To add a new record:

There are three ways to add a new record to a table:

- In the **Records** group on the **Home** tab, click the **New** command.
- On the **Record Navigation bar** at the bottom of the window, click the **New record** button.



ID	First Name	Last Name
52	Denver	Ferguson
53	John	Emory
54	Ebony	Farmer
55	Kim	Doe
56	Coretta	Diaz
57	Victor	Denver
58	Hamish	David
New (blank) record		

- Begin typing in the row below your last added record.

ID	First Name	Last Name	Street Address	City
198	Alex	Yuen	8 Marcus Ln.	Cary
199	Lia	Richards	890 Garvey St.	Durham
200	Karla	Nichols	981 DuBois Ct.	Durham
201	Tyrese	Hanlon	31 Crispus Ct. Apt B	Cary
202	Juan	Flores	122 Luna St.	Durham
204	Tobias	Inman	34 Peace St.	Raleigh
205	Teyonah	Lamb	321 Home Dr.	Raleigh
206	Gregoire	Pick	604 Hatsy Way	Charlotte
207	Mallary	Pont	91 Julianna Way	Charlotte
208	Rajsev	Parthasarathy	1009 Raleigh Street	Hillsborough
209	Will	Good	38 Learnfree St.	Raleigh
212	David	Barrett	434 Hill St.	Raleigh
(New)				

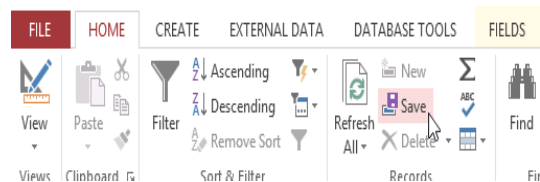
Sometimes when you enter information into a record, a window will pop up to tell you that the information you've entered is invalid. This means the field you're working with has a **validation rule**, which is a rule about the type of data that can appear in that field. Click **OK**, then follow the instructions in the pop-up window to **re-enter** your data.

31	Crispus Ct. Apt B	Cary	NC	27513
				27714
				27603
				27603
				28203
				28201
				27278
38	Learnfree St.	Raleigh	NC	27701
434	Hill St.	Raleigh	NC	27609
1874	Thistle-down Ave	Raleigh	North Carolina	

To save a record:

Access is designed to save records automatically. After you enter a record, you can either select a different record or close the object, and Access will save the record. However, in certain situations you may want to save a record manually. For example, if you needed to edit an existing record, you could save the record to ensure your changes are saved.

1. Select the **Home** tab, and locate the **Records** group.
2. Click the **Save** command. The record will be saved.



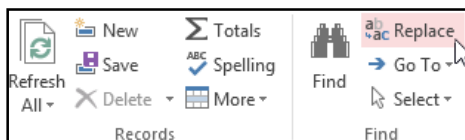
Editing records

To quickly edit any record within a table, you can click it and type your changes. However, Access offers you the ability to **find and replace** a word within multiple records and **delete** records entirely.

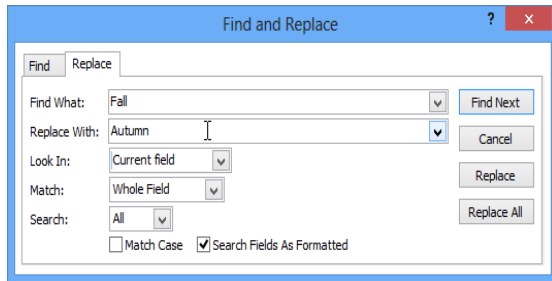
To replace a word within a record:

You can edit multiple occurrences of the same word by using **Find and Replace**, which searches for a term and replaces it with another term.

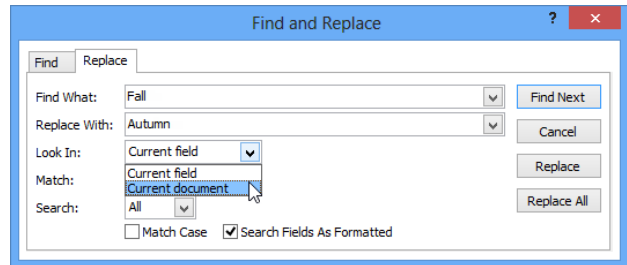
1. Select the **Home** tab, and locate the **Find** group.
2. Select the **Replace** command. The **Find and Replace** dialog box will appear.



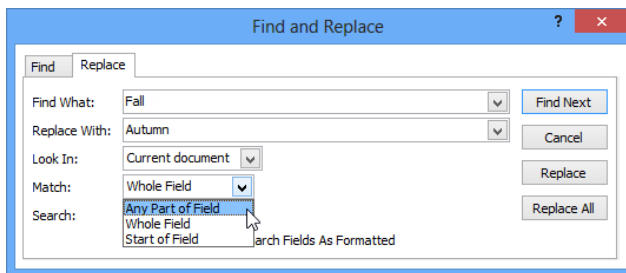
- In the **Find What:** field, type the word you want to find, then in the **Replace With:** field type the word you would like to replace the original word. In our example, we'll find instances of the word **Fall** and replace it with **Autumn**.



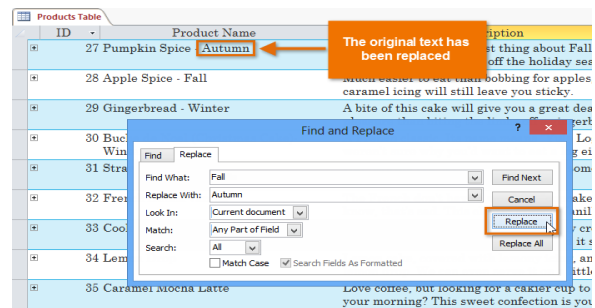
- Click the **Look In:** drop-down arrow to select the area you want to search.
 - Select **Current Field** to limit your search to the currently selected field.
 - Select **Current Document** to search within the entire table.



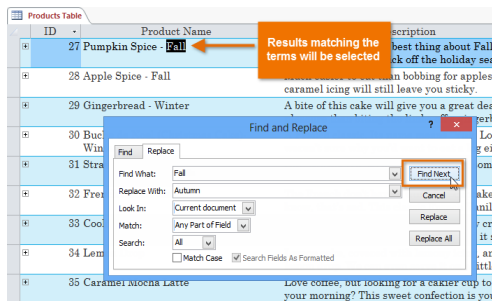
- Click the **Match:** drop-down arrow to select how closely you'd like results to match your search.
 - Select **Any Part of Field** to search for your search term in any part of a cell.
 - Select **Whole Field** to search only for cells that match your search term exactly.
 - Select **Beginning of Field** to search only for cells that start with your search term.



- Click **Find Next**. If the text is found, it will be **selected**.

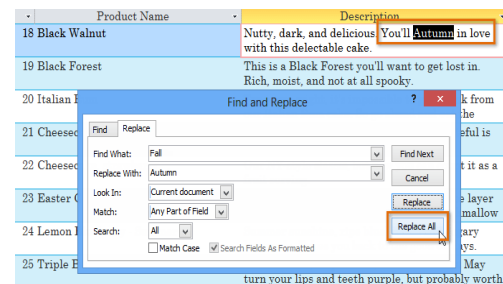


- Review the text to make sure you want to replace it. Click **Replace** to replace the original word with the new one.



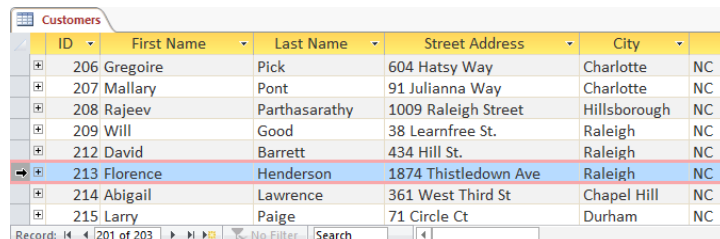
- Access will move to the next instance of the text in the object. When you are finished replacing text, click **Cancel** to close the dialog box.

The **Replace All** option is powerful, and it may actually change some things you don't want to change. In the example below, the word **fall** did not refer to the season, so replacing it with **Autumn** would be incorrect. Using the normal **Replace** option allows you check each instance before replacing the text. You can click **Find Next** to skip to the next instance without replacing the text.

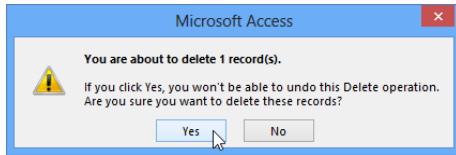
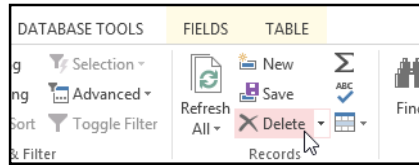


To delete a record:

- Select the entire record by clicking the **gray border** on the left side of the record.

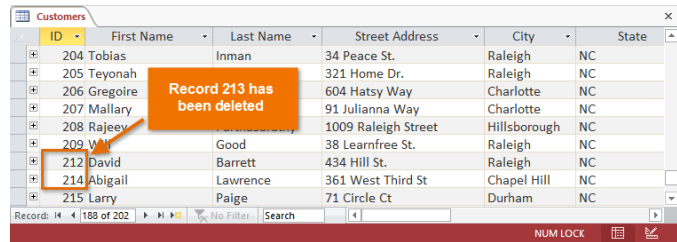


2. Select the **Home** tab and locate the **Records** group.
3. Click the **Delete** command.
4. A dialog box will appear. Click **Yes**.



5. The record will be permanently deleted.

The ID numbers assigned to records stay the same even after you delete a record. For example, if you delete the 213th record in a table the sequence of record ID numbers will read ...212, **214**, 215... rather than ...212, **213**, 214, 215...

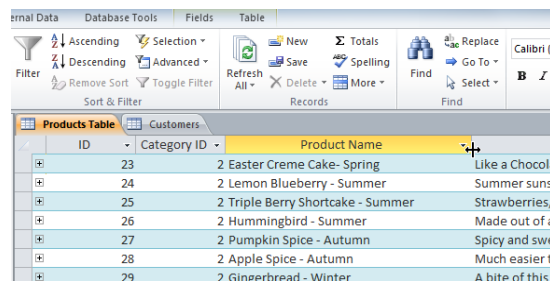
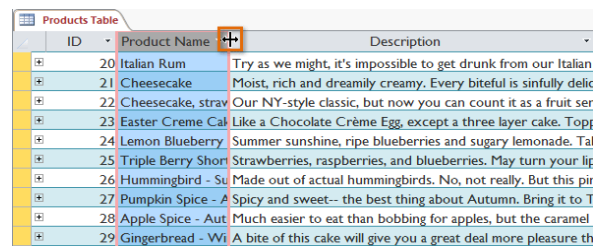


Resizing fields and rows

If your fields and rows are too small or large for the data contained with them, you can always **resize** them so all of the text is displayed.

To resize a field:

1. Place your cursor over the **right gridline** in the **field title**. Your mouse will become a **double arrow**.
2. Click and drag the gridline to the right to increase the field width or to the left to decrease the field width, then release the mouse. The field width will be changed.



To resize a row:

1. Place your cursor over the **bottom gridline** in the **gray area** to the left of the row. Your mouse will become a **double arrow**.
2. Click and drag the gridline downward to increase the row height or upward to decrease the row height, then release the mouse. The row height will be changed.

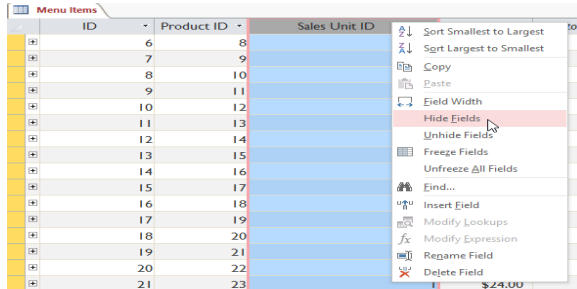


Hiding fields

If you have a field you don't plan on editing or don't want other people to edit, you can **hide** it. A hidden field is invisible but is still part of your database. Data within a hidden field can still be accessed from forms, queries, reports, and any related tables.

To hide a field:

1. Right-click the **field title**, then select **Hide Fields**.
2. The field will be hidden.



If you decide you want the field to be visible again, you can **unhide** it. Simply right-click any field title, then select **Unhide Fields**. A dialog box will appear. Click the checkboxes of any fields you want to be visible again, then click **Close**.

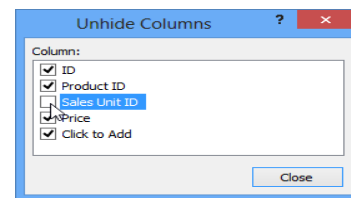


Table formatting options

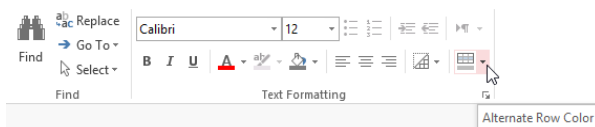
Alternate row color

By default, the background of every other row in an Access table is a few shades darker than the background of the rest of the table. This darker **alternate row color** makes your table easier to read by offering a **visual distinction** between each record and the records directly above and below it.

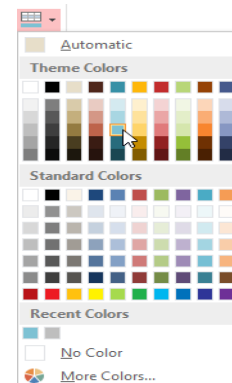
ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrold	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St.	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC

To change the alternate row color:

1. Select the **Home** tab, locate the **Text Formatting** group, and click the **Alternate Row Color** drop-down arrow.



2. Select a color from the drop-down menu, or select **No Color** to remove the alternate row color.



- The alternate row color will be updated.

ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrold	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St.	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC

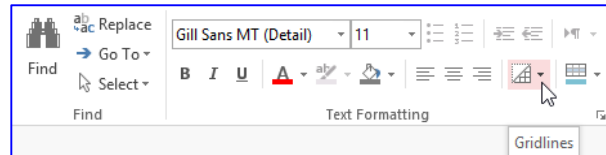
Modifying gridlines

Another way Access makes your tables easier to read is by adding **gridlines** that mark the borders of each cell. Gridlines are the **thin lines** that appear between each cell, row, and column of your table. By default, gridlines are dark gray and appear on every side of a cell, but you can change their **color** and **hide** undesired gridlines.

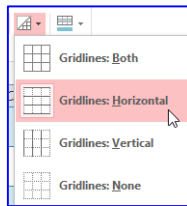
ID	Product Name	Description
1	Coconut	Ever tried cracking a coconut? It's hard! Better sti our most Coconut Cake-- all you need is your fo Natty, dark, and delicious. You'll fall in love with t delectable cake.
1	Black Walnut	This is a Black Forest you'll want to get lost in. Ric and not at all spooky.
2	Italian Rum	Try as we might, it's impossible to get drunk from Italian Rum cake. So go ahead and eat the whole th Moist, rich and dreamily creamy. Every bitiful is si delicious.
2	Cheesecake	Our NY-style classic, but now you can count it as serving.

To customize which gridlines appear:

1. Select the **Home** tab, locate the **Text Formatting** group, and click the **Gridlines** drop-down arrow.



2. Select the gridlines you want to appear. You can choose to have **horizontal** gridlines between the rows, **vertical** gridlines between the columns, **both** types of gridlines, or **none** at all.

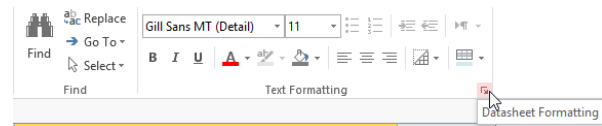


3. The gridlines on your table will be updated.

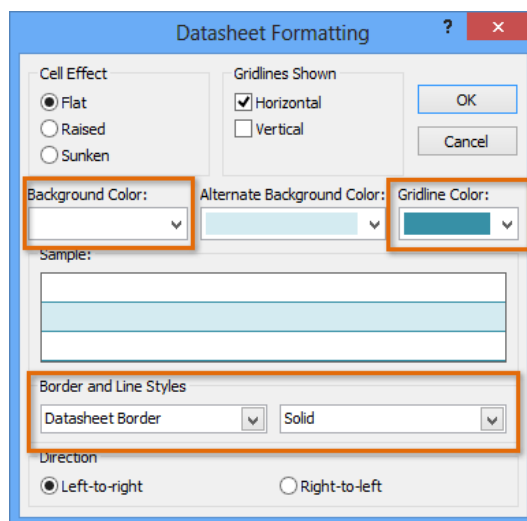
ID	Product Name
17	Coconut
18	Black Walnut
19	Black Forest
20	Italian Rum
21	Cheesecake
22	Cheesecake, strawberry

Additional formatting options

To view additional formatting options, click the **Datasheet Formatting** arrow in the bottom-right corner of the **Text Formatting** group.



The **Datasheet Formatting** dialog box offers several advanced formatting options, including the ability to modify background color, gridline color, and border and line style. It even includes the ability to view a **sample** table with your formatting choices, so play around with the various formatting options until you get your table looking the way you want it.



LESSON 6: WORKING WITH FORMS

Introduction

While you can always enter data directly into database tables, you might find it easier to use **forms**. Forms ensure you're entering the right data in the right location and format. This can help keep your database accurate and consistent. This lesson will address the **benefits of using forms** in a database. You will review examples of different forms and form components. Finally, you will learn how to **use forms** to **enter** new records and **view** and **edit** existing ones

Why use forms?

Many of us fill out forms so often that we hardly notice when we're asked to use them.

Forms are so popular because they're useful to the person asking for the information and to the person providing it. They are a way of requiring information in a specific format, which means the person filling out the form knows exactly which information to include and where to put it.



This is just as true of forms in Access. When you enter information into a form in Access, the data goes exactly where it's supposed to go: into one or more related tables. While entering data into simple tables is fairly straightforward, data entry becomes more complicated as you start populating tables with records from elsewhere in the database. For instance, the **orders table** in a bakery's database might link to information on customers, products, and prices drawn from related tables. For example, in the Orders Table below the Customer ID field is linked to the Customers table.

ID	Customer ID	Paid	Pre Order
13	139	Yes	Yes
14	130	Yes	No
15	129	Yes	Yes
16	124	Yes	No

In fact, in order to see the entire order you would also have to look at the **Order Items table**, where the menu items that make up each order are recorded.

ID	Order ID	Menu Item ID	Quantity
28	13	50	1
29	13	38	1
30	13	10	1


The records in these tables include **ID numbers** of records from other tables. You can't learn much just by glancing at these records because the ID numbers don't tell you much about the data they relate to. Plus, because you have to look at two tables just to view one order, you might have a difficult time even finding the right data. It's easy to see how viewing or entering many records this way could become a difficult and tedious task.

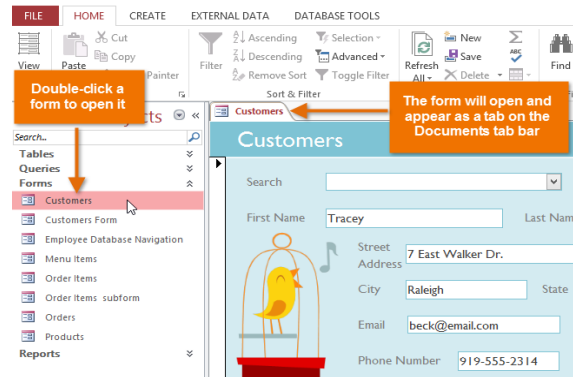
A form containing the same data might look like this:

As you can see, this record is much easier to understand when viewed in a form. Modifying the record also would be easier because you wouldn't have to know any ID numbers to enter new data. When you're using a form, you don't have to worry about entering data into the right tables or in the right format—the form can handle these things itself. There's no need to go back and forth between tables because forms bring all of the information you need together in one place. Not only do forms make the data entry process easier for the user, but they also keep the database itself working smoothly. With forms, database designers can control exactly how users are able to interact with the database. They can even set restrictions on individual form components to ensure all of the needed data is entered and that it's all entered in a valid format. This is useful because keeping data consistent and organized is essential for an accurate and powerful database.

Working with forms

To open an existing form:

1. Open your database, and locate the **Navigation pane**.
2. In the Navigation pane, locate the form you want to open. Forms are marked with the  icon.
3. Double-click the desired form. It will open and appear as a **tab** in the **Document Tabs bar**.



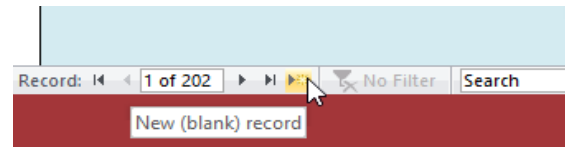
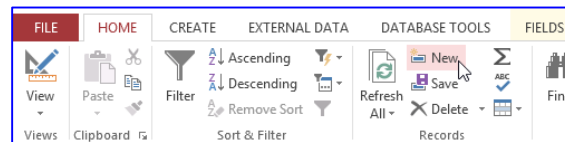
Entering and modifying data

Depending on the database you're using, the forms you work with may include special tools and features that let you perform common tasks with one click of a button. You'll see examples of these tools in the inter actives on the next page. However, no matter what type of form you're working with, you can follow the same procedures for carrying out certain basic tasks.

To add a new record:

There are two ways to add a new record to a form:

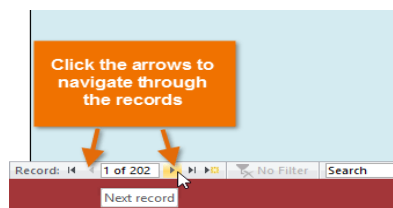
- In the **Records** group on the **Home** tab of the **Ribbon**, click the **New** command.
- On the **Record Navigation bar** at the bottom of the window, click the **New Record** button.



To find an existing record to view or edit:

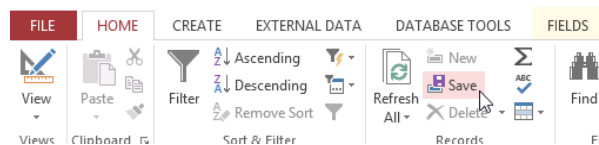
There are two ways to find and view an existing record using a form, and they both use the **Navigation bar** at the bottom of the screen:

- To look through records one at a time, click the **navigation arrows**. The right arrow will take you to the next record, and the left arrow will take you to the previous one.
- To **search** for a record, type a word you know is contained in that record in the **navigation search box**.

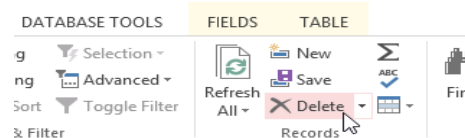


To save the current record:

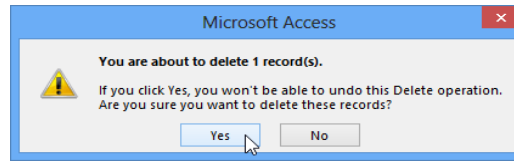
1. Select the **Home** tab and locate the **Records** group.
2. Click the **Save** command. The current record will be saved.



To delete the current record:



1. Select the **Home** tab and locate the **Records** group.
2. Click the **Delete** command.
3. A dialog box will appear. Click **Yes**.
4. The record will be permanently deleted.



Using form features

The exact procedure you use for filling out a form will vary depending on the content and design of the form you're using. The forms in your database might be similar to the examples in the two interactive below. Between them, they include most of the features you'll commonly encounter in forms.

Click the buttons in the interactive below to learn about a **simple form**.

A screenshot of a Microsoft Access form titled "Customers". The form has a search box at the top left and a "New Record" button at the top right. Below these are input fields for "First Name" (Filip), "Last Name" (Piatek), "Street Address" (7 East Walker Dr.), "City" (Raleigh), "State" (NC), "Zip Code" (27605), "Email" (piatekfil@email.com), and "Phone Number" (919-555-1285). There is also a dropdown menu for "Add to Mailing List?" set to "Yes-- Weekly & Events". At the bottom, there are three small icons: a printer, a refresh, and a save icon.

Some forms may include more options, like calendar buttons, drop-down lists, yes-no checkboxes, subforms, and embedded tables.

Click the buttons in the interactive below to learn about a more **complex form**.

A screenshot of a Microsoft Access form titled "Orders". It features a "New Order" button at the top. Below it are fields for "Customer" (Love), "Order #" (39), and "Pickup Date" (12/24/2013). A "Notes" field contains the text: "Apparently, he plans to eat most of this himself!! Maybe that was a joke...". There are two checkboxes: "Pre Order" (checked) and "Paid" (checked). Below the notes is an "Add Item" button and a table with columns: Category, Product, Quantity, "Unit", Price, and Subtotal. The table lists various items like Pumpkin Pie, Peanut Butter Chocolate, Carrot Cake, Italian Rum, German Chocolate, Gingerbread - Winter, Tiramisu, and Buche de Noel (Christmas Cake). A "Total" row at the bottom shows a subtotal of \$251.00. At the very bottom, there is a status bar showing "Record: 1 of 9" and a search box.

Category	Product	Quantity	"Unit"	Price	Subtotal
Pies	Pumpkin	1	Single	\$15.00	\$15.00
Pies	Peanut Butter Chocolate	1	Single	\$16.00	\$16.00
Cakes	Carrot Cake	3	Single	\$22.00	\$66.00
Cakes	Italian Rum	1	Single	\$22.00	\$22.00
Cakes	German Chocolate	1	Single	\$22.00	\$22.00
Cakes	Gingerbread - Winter	1	Single	\$24.00	\$24.00
Cakes	Gingerbread - Winter	1	Single	\$24.00	\$24.00
Pastries	Tiramisu	1	One Dozen	\$30.00	\$30.00
Cakes	Buche de Noel (Christmas Cake)	1	Single	\$32.00	\$32.00
Total					\$251.00

LESSON 7: SORTING AND FILTERING RECORDS

Introduction

Access gives you the ability to work with enormous amounts of data, which means it can be difficult to learn anything about your database just by glancing at it. **Sorting** and **filtering** are two tools that let you customize how you organize and view your data, making it more convenient to work with. In this lesson, you'll learn how to **sort** and **filter** records

About sorting and filtering

Essentially, sorting and filtering are tools that let you **organize your data**. When you sort data, you are **putting it in order**. Filtering data lets you hide unimportant data and focus only on the data you're interested in.

Sorting records

When you **sort** records, you are putting them into a **logical order**, with **similar data grouped together**. As a result, sorted data is often simpler to read and understand than unsorted data. By default, Access sorts records by their **ID numbers**. However, there are many other ways records can be sorted. For example, the information in a database belonging to a bakery could be sorted in a number of ways:

- Orders could be sorted by **order date** or by the **last name** of the customers who placed the orders.
- Customers could be sorted by **name** or by the **city** or **zip code** where they live.
- Products could be sorted by **name**, **category** (like pies, cakes, and cupcakes), or **price**.

You can sort both **text** and **numbers** in two ways: in **ascending** order and **descending** order. **Ascending** means **going up**, so an ascending sort will arrange numbers from **smallest to largest** and text from **A to Z**. **Descending** means **going down**, or **largest to smallest** for numbers and **Z to A** for text. The default ID number sort that appears in your tables is an ascending sort, which is why the lowest ID numbers appear first.

In our example, we will be performing a sort on a table. However, you can sort records in any Access object. The procedure is largely the same.

To sort records:

1. Select a field you want to sort by. In this example, we will sort by customers' last names.
2. Click the **Home** tab on the Ribbon, and locate the **Sort & Filter** group.
3. Sort the field by selecting the **Ascending** or **Descending** command.
 - Select **Ascending** to sort text A to Z or to sort numbers from smallest to largest. We will select this in our example because we want the last names to be in A-to-Z order.
 - Select **Descending** to sort text Z to A or to sort numbers from largest to smallest.

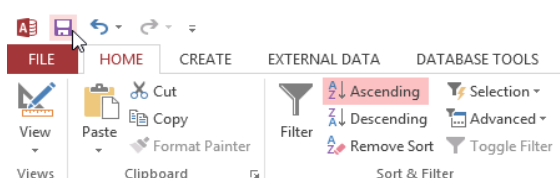
ID	First Name	Last Name	Street Address
1	Tracey	Beckham	7 East Walker Dr.
2	Lucinda	George	789 Brewer St.
3	Jerrold	Smith	211 St. George Ave.
4	Brett	Newkirk	47 Hillsborough St.
5	Chloe	Jones	23 Solo Ln.
6	Quinton	Boyd	4 Cypress Cr.
7	Alex	Hinton	1011 Hodge Ln.
8	Nisha	Hall	123 Huntington St.
9	Hillary	Clayton	2516 Newman
10	Kiara	Williams	9014 Miller Ln.
11	Katy	Jones	456 Denver Rd.
12	Beatrix	Joslin	85 North West St.
13	Mariah	Allen	12 Jupe
14	Jennifer	Hill	2100 Field Ave.



4. The table will now be sorted by the selected field.

ID	First Name	Last Name	Street Address
102	Theodore	Achi	120 Baker St.
195	Kris	Ackerman	1311 Coretta Scott Way
78	Michiko	Akiwana	901 Glenwood Ave.
188	Nathan	Albee	76-C Meadowview Ln.
13	Mariah	Allen	12 Jupe
37	Carol	Allenson	3201 Glenwood Ave. Unit A
38	Zoey	Altman	817 Hillsborough St. Apt E11
163	Franz	Angelou	291 Hinton St.
87	Robert	Armisen	21 MLK Blvd.
47	Hakim	Auden	921 Dawson St.
129	Yann	Augerot	88 Carolina Rd.
40	Vig	Aurelio	53 Pine St.
212	David	Barrett	434 Hill St.

5. To save the new sort, click the **Save** command on the Quick Access toolbar.



After you save the sort, the records will stay sorted this way until you perform another sort or remove the current one. To remove a sort, click the **Remove Sort** command.



Filtering records

Filters allow you to view **only the data you want to see**. When you create a filter, you set **criteria** for the data you want to display. The filter then searches all of the records in the table, finds the ones that meet your search criteria, and temporarily hides the ones that don't.

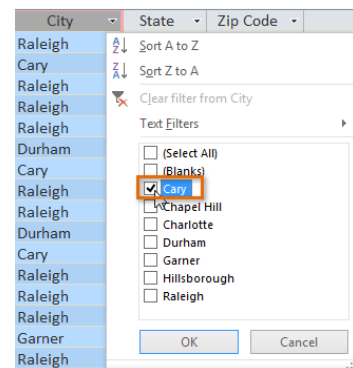
Filters are useful because they allow you to **focus in** on specific records without being distracted by the data you're uninterested in. For instance, if you had a database that included customer and order information, you could create a filter to display only customers living within a certain city or only orders containing a certain product. Viewing this data with a filter would be far more convenient than searching for it in a large table.

In our examples and explanations, we will be applying filters to tables. However, you can apply filters to any Access object. The procedure is largely the same.

To create a simple filter:

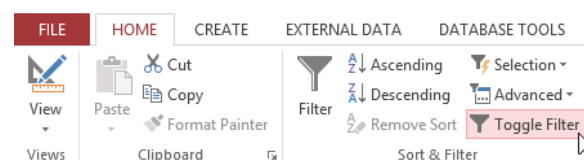
- Click the **drop-down arrow** next to the field you want to filter by. We will filter by city because we want to see a list of customers who live in a certain city.
- A drop-down menu with a checklist will appear. Only checked items will be included in the filtered results. Use the following options to determine which items will be included in your filter:
 - Select** and **deselect** items one at a time by clicking their checkboxes.
 - Click **Select All** to include every item in the filter. If all items are already selected, this option will deselect all items.
 - Click **Blanks** to set the filter to find only the records with no data in the selected field.
- Click **OK**. The filter will be applied. Our customers table now displays only customers who live in Cary.

ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrod	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St.	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC
11	Katy	Jones	456 Denver Rd.	Cary	NC
12	Beatrix	Joslin	85 North West St.	Raleigh	NC



Last Name	Street Address	City	State
George	789 Brewer St.	Cary	NC
Hinton	1011 Hodge Ln.	Cary	NC
Jones	456 Denver Rd.	Cary	NC
Kellerman	76 Murphy Ave.	Cary	NC
Hamm	1221 Coretta Scott Way	Cary	NC
James	4221 Basil Ct.	Cary	NC
Slobodowski	7 Greene St.	Cary	NC
White	911 Oregon Tr.	Cary	NC
Sigrudsdatter	55 Cameron Ct.	Cary	NC
Yuen	8 Marcus Ln.	Cary	NC
Hanlon	31 Crispus Ct. Apt B	Cary	NC

Toggle your filter allows you to turn it on and off. To view the records without the filter, click the **Toggle Filter** command. To restore the filter, click it again.

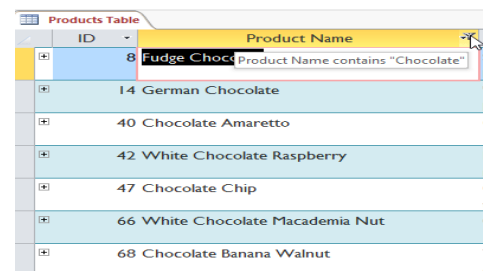
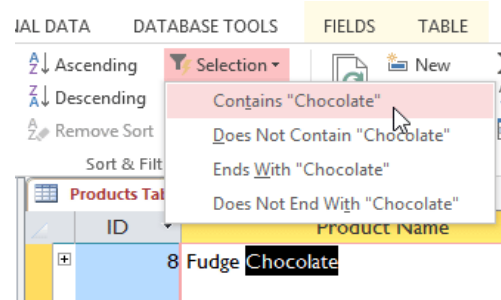
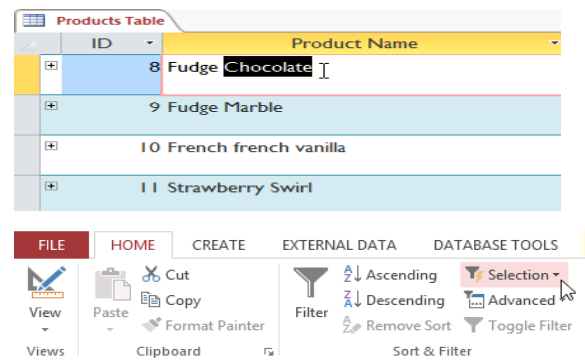


Creating a filter from a selection

Filtering by selection allows you to **select specific data** from your table and find data that is **similar** or **dissimilar** to it. For instance, if you were working with a bakery's database and wanted to search for all products whose names contained the word **chocolate**, you could select that word in one product name and create a filter with that selection. Creating a filter with a selection can be more convenient than setting up a simple filter if the field you're working with contains many items.

To create a filter from a selection:

1. Select the cell or data you want to create a filter with. We want to see a list of all of our products that contain the word **chocolate** in their names, so we'll select the word **chocolate** in the **Product Name** field.
2. Select the **Home** tab on the Ribbon, locate the **Sort & Filter** group, and click the **Selection** drop-down arrow.
3. Select the type of filter you want to apply:
 - **Contains** includes only records with cells that contain the selected data. We'll select this because we want to see records that contain the word **chocolate** anywhere in the title.
 - **Does Not Contain** includes all records **except** for those with cells that contain the selected data.
 - **Ends With** includes only records whose data for the selected field **ends** with the search term.
 - **Does Not End With** includes all records **except** for those whose data for the selected field ends with the search term.
4. The filter will be applied. Our table now displays only products with the word **chocolate** in their names.



Creating a filter from a search term

You can also create a filter by entering a **search term** and specifying the way Access should match data to that term. Creating a filter from a search term is similar to creating a filter from a selection.

Filtering text by a search term

When filtering text by entering a search term, you can use some of the same options you use when filtering by a selection, like **Contains**, **Does Not Contain**, **Ends With**, and **Does Not End With**. You can also choose from the following options:

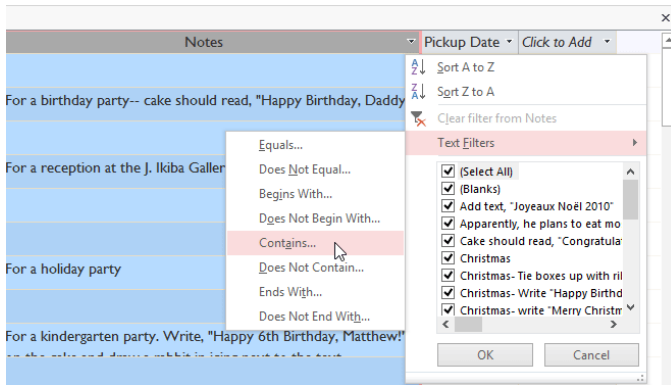
- **Equals**, which includes only records with data that is identical to the selected data
- **Does Not Equal**, which includes all records except for the data that is identical to the selection
- **Begins With**, which includes only records whose data for the selected field **begins** with the search term
- **Does Not Begin With**, which includes all records **except** for those whose data for the selected field begins with the search term

To filter text by a search term:

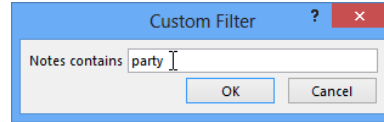
1. Click the **drop-down arrow** next to the field you want to filter by. We want to filter the records in our orders table to display only those that contain notes with certain information, so we'll click the arrow in the **Notes** field.

ID	Customer ID	Paid	Pre Order	Notes
5	44	Yes	No	
6	136	Yes	Yes	For a birthday party-- cake should read, "Happy Birthday, Daddy!"
7	131	Yes	No	
8	145	Yes	Yes	For a reception at the J. Ikiba Gallery

- In the drop-down menu, hover your mouse over **Text Filters**. From the list that appears, select the way you want the filter to match the term you enter. In this example, we want to view only records whose notes indicate the order was placed for a party. We'll select **Contains** so we can search for records that contain the word **party**.



- The **Custom Filter** dialog box will appear. Type the word you want to use in your filter.



- Click **OK**. The filter will be applied.

ID	Customer ID	Paid	Pre Order	Notes
6	136	Yes	Yes	For a birthday party-- cake should read, "Happy Birthday, Daddy"
11	92	Yes	Yes	For a holiday party
13	139	Yes	Yes	For a kindergarten party. Write, "Happy 6th Birthday, Matthew!" on the cake and draw a rabbit in icing next to the text.
15	129	Yes	Yes	For a holiday party
18	96	Yes	No	For a Christmas party
21	71	Yes	Yes	For an office party
41	111	Yes	Yes	For a NYE party

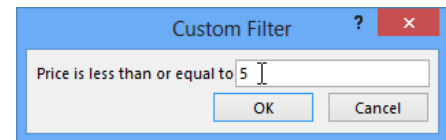
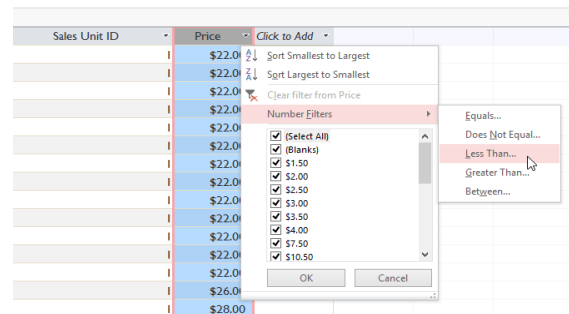
Filtering numbers with a search term

The process for filtering numbers with a search term is similar to the process for filtering text. However, different filtering options are available to you when working with numbers. In addition to **Equals** and **Does not Equal**, you can choose:

- Greater Than** to include only records with numbers in that field that are **greater than or equal to** the number you enter
- Less Than** to include only records with numbers in that field that are **less than or equal to** the number you enter
- Between** to include records with numbers that fall within a certain range

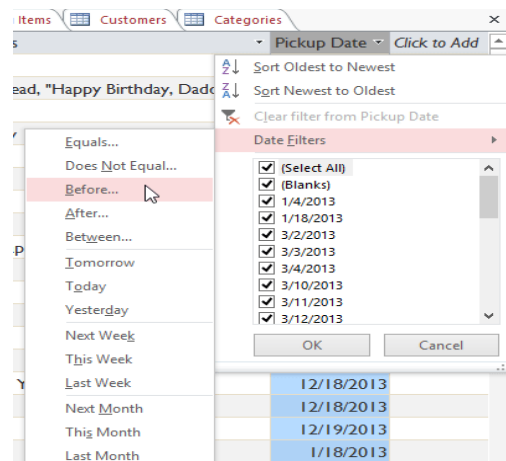
To filter numbers by a search term:

- Click the **drop-down arrow** next to the field you want to filter by. We want to filter the records in our menu items table by price, so we'll click the arrow in the **Price** field.
- In the drop-down menu, hover your mouse over **Number Filters**. From the list that appears, select the way you want the filter to match your search term. In this example, we want to see items that are less than \$5, so we'll select **Less Than**.
- The **Custom Filter** dialog box will appear. Type the number or numbers you want to use in your filter. We'll type **5** so the filter will show us only menu items that cost \$5 or less.
- Click **OK**. The filter will be applied.



ID	Product ID	Sales Unit ID	Price
30	46		\$1.50
33	47		\$1.50
36	58		\$1.50
39	59		\$1.50
42	60		\$1.50
45	61		\$1.50
48	62		\$1.50
51	63		\$1.50
54	64		\$1.50
57	65		\$2.00
60	66		\$2.00

Specific types of numbers may include other filtering options. For instance, **dates** stored in numerical form (mm/dd/yyyy, or 12/01/2013) include options to filter by periods of time.



LESSON 8: DESIGNING A SIMPLE QUERY

Introduction

The real power of a relational database lies in its ability to quickly **retrieve** and **analyze** your data by running a query. **Queries** allow you to **pull information** from one or more tables based on a set of search conditions you define. In this lesson, you will learn how to create a simple **one-table query**.

What are queries?

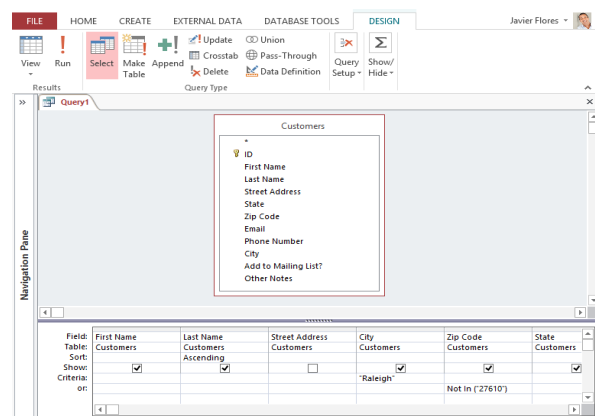
Queries are a way of **searching** for and **compiling** data from one or more tables. Running a query is like asking a **detailed question** of your database. When you build a query in Access, you are **defining specific search conditions** to find exactly the data you want.

How are queries used?

Queries are far more powerful than the simple searches or filters you might use to find data within a table. This is because queries can draw their information from **multiple** tables. For example, while you could use a **search** in the customers table to find the name of one customer at your business or a **filter** on the orders table to view only orders placed within the past week, neither would let you view both customers and orders at once. However, you could easily run a **query** to find the name and phone number of every customer who's made a purchase within the past week. A well-designed query can give information you might not be able to find out just by examining the data in your tables.

When you run a query, the results are presented to you in a table, but when you design one you use a different view. This is called **Query Design view**, and it lets you see how your query is put together.

Click the buttons in the interactive below to learn how to navigate the **Query Design view**.



One-table queries

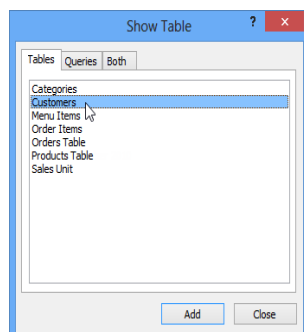
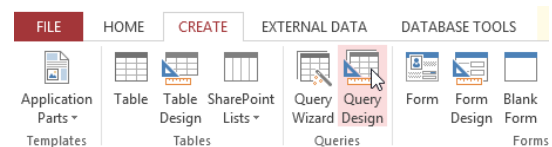
Let's familiarize ourselves with the query-building process by building the **simplest** query possible: a one-table query.

We will run a query on the **Customers** table of our bakery database. Let's say our bakery is having a special event, and we want to invite our customers who live nearby because they are the most likely to come. This means we need to see a list of all customers who live close by, and **only** those customers.

If you think this sounds a little like applying a filter, you're right. A one-table query is actually just an **advanced filter** applied to a table.

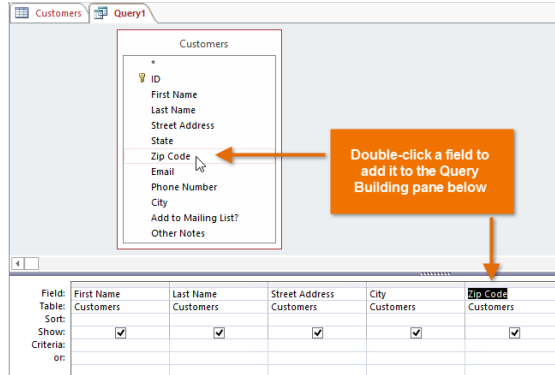
To create a simple one-table query:

1. Select the **Create** tab on the Ribbon, and locate the **Queries** group.
2. Click the **Query Design** command.



3. Access will switch to **Query Design view**. In the **Show Table** dialog box that appears, select the table you want to run a query on. We are running a query about our customers, so we'll select the **Customers** table.
4. Click **Add**, then click **Close**.
5. The selected table will appear as a small window in the **Object Relationship pane**. In the table window, double-click the **field names** you want to include in your query. They will be added to the **design grid** in the bottom part of the screen.

In our example, we want to mail invitations to customers who live in a certain area, so we'll include the **First Name**, **Last Name**, **Street Address**, **City**, and **Zip Code** fields. We aren't planning on calling or emailing our customers, so we don't have to include the **Phone Number** or **Email** fields.



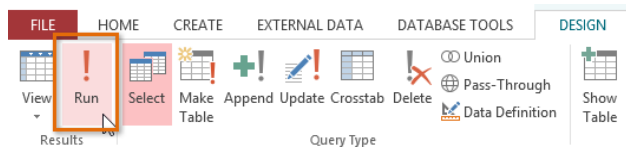
- Set the **search criteria** by clicking the cell in the **Criteria:** row of each field you want to filter. Typing criteria into more than one field in the Criteria: row will set your query to include only results that meet all criteria. If you want to set multiple criteria but don't need the records shown in your results to meet all of them, type the first criteria in the Criteria: row and additional criteria in the **or:** row and the rows beneath it.

For this one-table query, we'll use simple search criteria.

- We want to find our customers who live in the city of **Raleigh**, so in our **City** field we'll type "**Raleigh**". Typing Raleigh in **quotation marks** will retrieve all records with an **exact match** for Raleigh in the City field.
- Some customers who live in the suburbs live fairly close by, and we'd like to invite them as well. We'll add their zip code, **27513**, as another criteria. Because we want to find customers who either live in Raleigh **or** in the 27513 zip code, we'll type "27513" into the **or:** row of the **Zip Code** field.

Field:	City	Zip Code
Table:	Customers	Customers
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"Raleigh"	
or:		"27513"

- After you have set your criteria, **run** the query by clicking the **Run** command on the **Design** tab.



- The query results will be displayed in the query's **Datasheet view**, which looks like a table. If you want, **save** your query by clicking the **Save** command in the Quick Access toolbar. When prompted to name it, type the desired name, then click **OK**.

First Name	Last Name	Street Address	City	Zip Code
Tracey	Beckham	7 East Walker Dr.	Raleigh	27612
Lucinda	George	789 Brewer St.	Cary	27513
Jerrold	Smith	211 St. George Ave.	Raleigh	27610
Brett			Raleigh	27608
Chloe			Raleigh	27609
Alex			Cary	27513
Nisha			Raleigh	27612
Hillary			Raleigh	27606
Katy			Cary	27513
Beatrice	Joslin	85 North West St.	Raleigh	27606
Mariah	Allen	12 Jupe	Raleigh	27605
Jennifer	Hill	2100 Field Ave.	Raleigh	27609
Cody	Hayes	65 North St.	Raleigh	27609
Amaya	Gibson	5 West St.	Raleigh	27612

Now you know how to create the simplest type of query with only **one table**. In the next lesson, you'll learn how to create a query that uses **multiple tables**.

LESSON 9 : DESIGNING A MULTI-TABLE QUERY

Introduction

In the previous lesson, you learned how to create a **simple query** with one table. Most queries you design in Access will likely use **multiple tables**, allowing you to answer more complex questions. In this lesson, you'll learn how to design and create a **multi-table query**.

Designing a multi-table query

Queries can be difficult to understand and build if you don't have a good idea of what you're trying to find and how to find it. A one-table query can be simple enough to make up as you go along, but to build anything more powerful you'll need to plan the query in advance.

Planning a query

When planning a query that uses more than one table, you should go through these four steps:

1. **Pinpoint** exactly what you want to know. If you could ask your database any question, what would it be? Building a query is more complicated than just asking a question, but knowing precisely what question you want to answer is essential to building a useful query.
2. **Identify** every type of information you want included in your query results. Which fields contain this information?
3. **Locate** the fields you want to include in your query. Which tables are they contained in?
4. **Determine** the criteria the information in each field needs to meet. Think about the question you asked in the first step. Which fields do you need to search for specific information? What information are you looking for? How will you search for it?

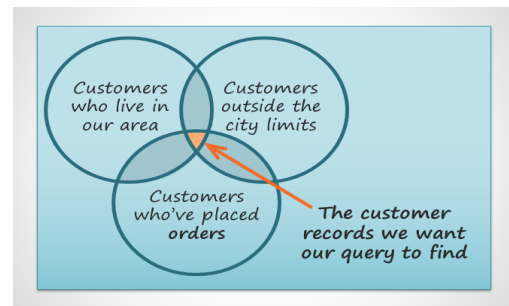
This process might seem abstract at first, but as we go through the process of planning our own multi-table query you should start to understand how planning your queries can make building them a lot easier.

Planning our query

Let's go through this planning process with a query we'll run on our bakery database. As you read through the planning process step by step, think about how each part of the planning process could apply to other queries you might run.

Step 1: Pinpointing the question we want to ask

Our bakery database contains many customers, some of whom have never placed an order but who are in our database because they signed up for our mailing list. Most of them live within the city limits, but others live out of town or even out of state. We want to get our out-of-town customers who've placed orders in the past to come back and give us another try, so we're going to mail them some coupons. We don't actually want our list to include customers who live too far away; sending a coupon to someone who doesn't live in our area probably won't make that person come in. So we just want to find people who don't live in our city but who still live in our area.



In short, the question we want our query to answer is this: **Which customers live in our area, are outside the city limits, and have placed an order at our bakery?**

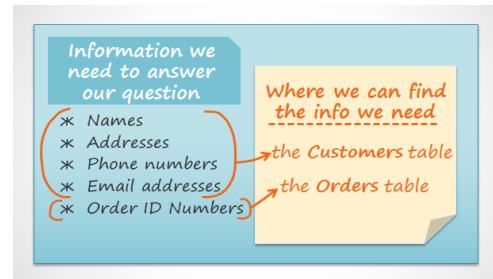
Step 2: Identifying the information we need

What information might we want to see in a list about these customers? Obviously, we'll need the **customers' names** and their **contact information**—their **addresses, phone numbers, and email addresses**. But how are we going to know if they've placed orders? Each record of an order identifies the customer who placed that order. If we include the **order ID numbers**, we should be able to narrow our list down to only customers who have previously placed orders.



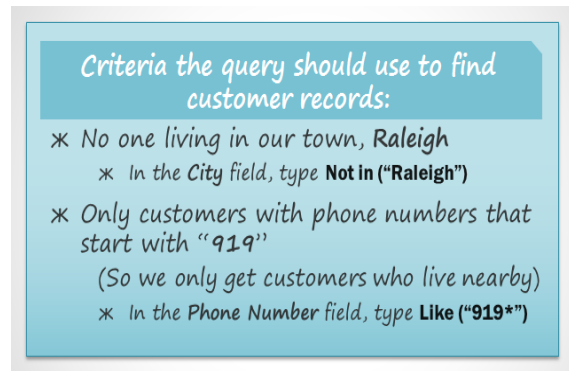
Step 3: Locating the tables containing the information we need

In order to write a query, you need to be familiar with the different tables in your database. From working extensively with our own database, we know that the customer information we need is located in fields in the **Customers** table. Our **Order ID numbers** are in a field in the **Orders** table. We only need to include these two tables to find all of the information we need.



Step 4: Determining the criteria our query should search for

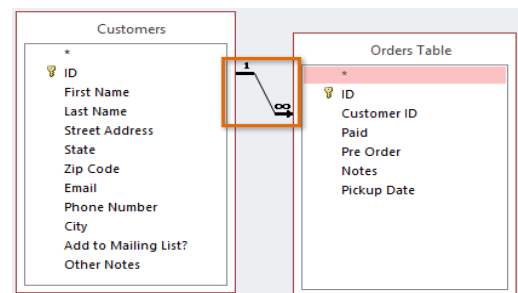
When you set criteria for a field in a query, you are basically applying a filter to it that tells the query to retrieve only information that matches your criteria. Review the list of fields we are including in this query. How and where can we set criteria that will best help us answer our question? We don't want customers who live in our town, Raleigh, so we want a criteria that will return all records **except for** those with **Raleigh** in the city field. We don't want customers who live too far away, either. All of the phone numbers in the area start with the area code 919, so we'll also include a criteria that will only return records whose entries from the **phone number field** begin with **919**. This should guarantee that we'll only send coupons to customers who live close enough to actually come back and use them. We won't set a criteria for the order ID field or any other fields because we want to see **all** of the orders made by people who meet the two criteria we just set.



To write queries, you'll need to be able to set criteria in a language that Access **understands**. As you can see in the image above, our criteria requiring phone numbers to begin with 919 must be typed like this: **Like ("919*")**. To learn how to write additional criteria, consult our printable **Query Criteria Quick Reference Guide**, which includes several of the most common criteria used in Access queries.

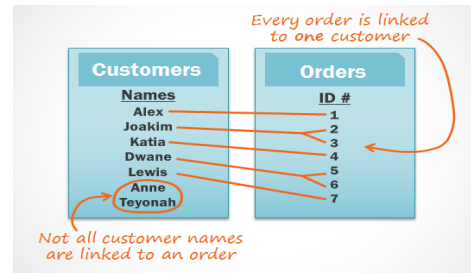
Joining tables in queries

The final thing you need to consider when designing a query is the way you link—or **join**—the tables you're working with. When you add two tables to an Access query, this is what you'll see in the **Object Relationship pane**:



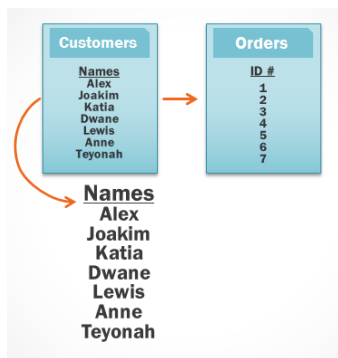
The line connecting the two tables is called the **join line**. See how the join line is actually an arrow? This is because it indicates the order in which the query looks at data from the two tables. In the image above, the arrow is pointing from **left to right**, which means the query will look at data in the **left** table first, then look at only the data in the **right** table that **relates** to the records it's already seen in the left table. Your tables won't always be joined this way. Sometimes Access will join them **right to left**. In either case, you might need to **change the direction** of the join to make sure your query includes the correct information. The join direction can affect **which information** your query **retrieves**.

To understand what this means, consider the query we're designing. For our query, we need to see customers who have placed orders, so we've included the **Customers** table and the **Orders** table. Let's take a look at some of the data contained in these tables.

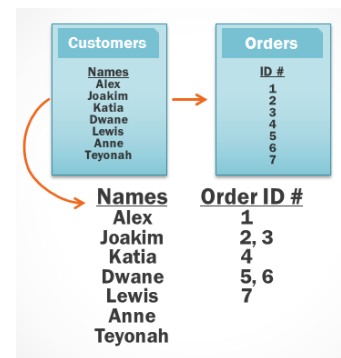


What do you notice when you look at these lists? First of all, every single order in the **Orders** table is linked to someone in the **Customers** table—the customer who placed that order. However, when you look at the Customers table, you'll see that the customers who've placed multiple orders are linked to more than one order, and those who've never placed an order are linked to no orders. As you can see, even when two tables are linked it's possible to have records in one table that have no relationship to any record in the other table.

So what happens when Access tries to run our query with the current join, **left to right**? It pulls every record from the table to the left: our Customers table.

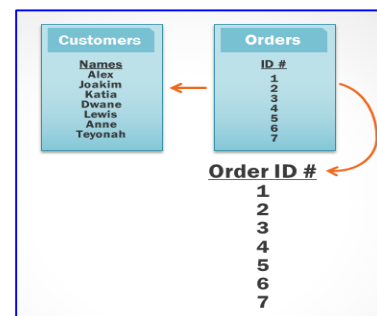


It then retrieves every record from the **right** table that has a relationship with a record Access has already taken from the left table.

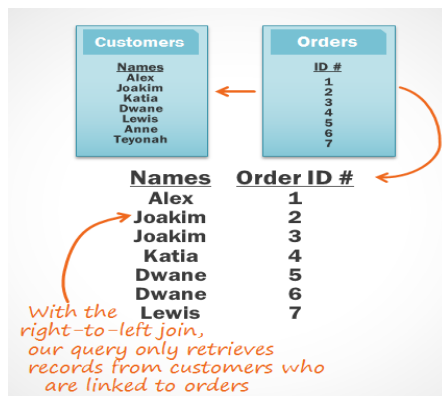


Because our join began with the **Customers** table, our query will include records for **all** of our customers, including those who've never placed orders. This is more information than we need. We **only** want to see records for **customers who have placed orders**.

Fortunately, we can fix this problem by changing the direction of the join line. If we join the tables from **right to left** instead, Access will first retrieve the orders from the **right** table, our **Orders** table:



Then Access will look at the left table and retrieve **only** the records of customers who are linked to an order on the right.



We now have exactly the information we want: **all** of the customers who have placed an order, and **only** those customers. As you can see, we had to join our tables in the **correct direction** to obtain the information we wanted.

Now that we understand which join direction we need to use, we're ready to build our query!

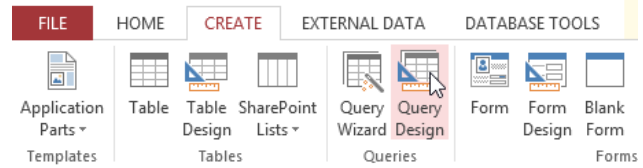
In our query, we needed to use the **right-to-left** join, but the correct join direction for the tables in your queries will depend on **what** information you want to see and **where** that information is stored. When you add tables to a query, Access will automatically join the tables for you, but it often doesn't join them in the correct direction. This is why it's important to **always review the joins** between your tables before you build a query.

Creating a multi-table query

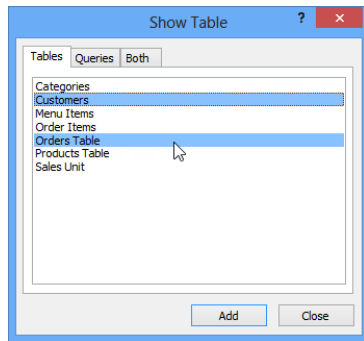
Now that we've planned our query, we're ready to design and run it. If you have created written plans for your query, be sure to reference them often throughout the query design process.

To create a multi-table query:

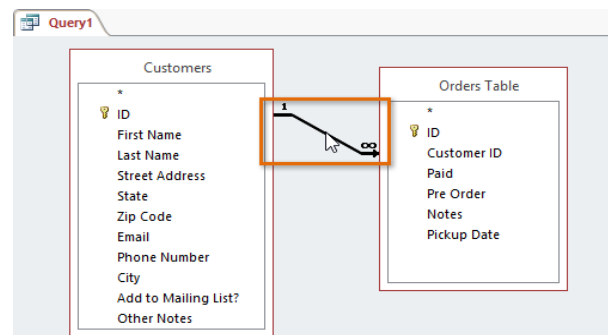
1. Select the **Query Design** command from the **Create** tab on the Ribbon.



2. In the dialog box that appears, select each table you want to include in your query and click **Add**. You can press and hold the **Ctrl** key on your keyboard to select more than one table. When we planned our query, we decided we needed information from the **Customers** and **Orders** table, so we'll add these.

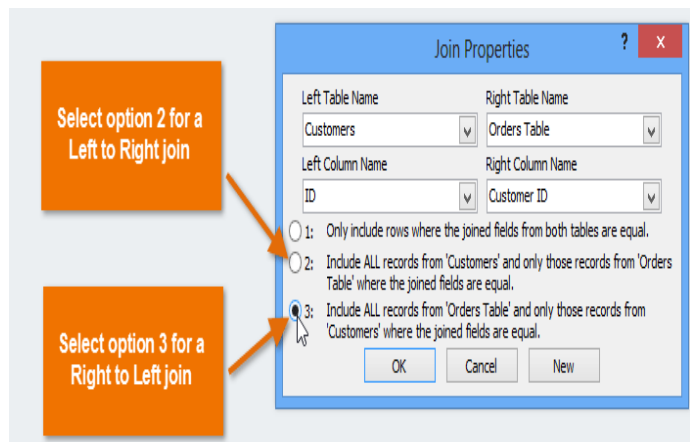


3. After you have added all of the tables you want, click **Close**.
4. The tables will appear in the **Object Relationship pane**, linked by a **join line**. Double-click the thin section of the join line between two tables to edit its **join direction**.



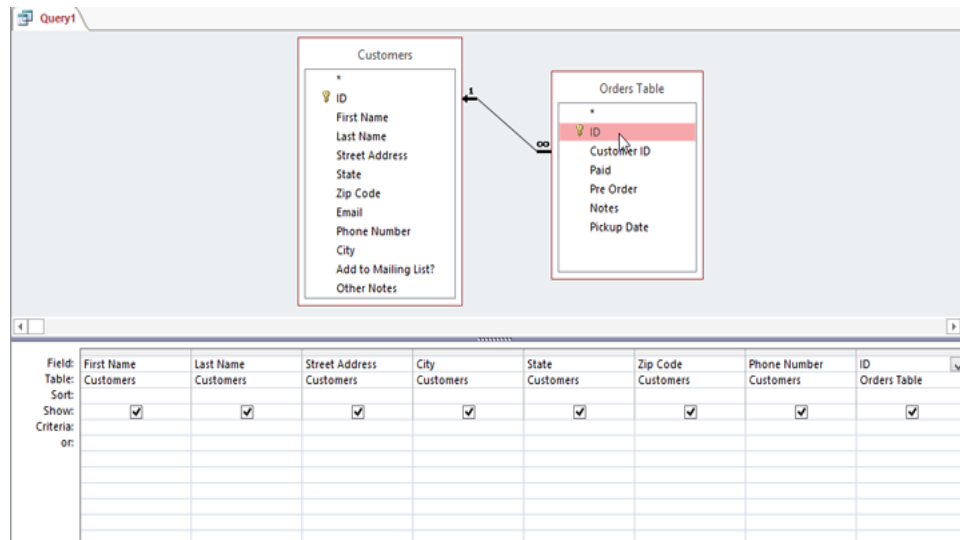
5. The **Join Properties** dialog box will appear. Select an option to choose the direction of your join.

- Choose option **2**: for a **left-to-right** join. In our query, the **left** table is the **Customers** table, so choosing this would mean all of the customers who met our location criteria—whether or not they had placed an order—would be included in our results. We don't want to choose this option for our query.
- Choose option **3**: for a **right-to-left** join. Because our **right** table is our **Orders** table, selecting this option will let us work with records for **all** of the orders and **only** the customers who've placed orders. We'll choose this option for our query because this is exactly the data we want to see.



6. In the table windows, double-click the **field names** you want to include in your query. They will be added to the **design grid** in the bottom part of the screen.

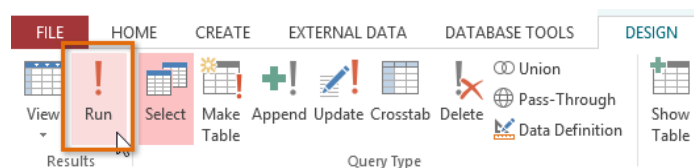
In our example, we'll include most of the fields from the **Customers** table: **First Name, Last Name, Street Address, City, State, Zip Code,** and **Phone Number**. We'll also include the **ID number** from the **Orders** table.



7. Set field **criteria** by entering the desired criteria in the criteria row of each field. We want to set two criteria:
- First, to find customers who do **not** live in Raleigh, we'll type **Not in ("Raleigh")** in the **City** field.
 - Second, to find customers who have a phone number beginning with the area code **919**, we'll type **Like ("919*")** in the **Phone Number** field.

Field:	City	State	Zip Code	Phone Number	ID
Table:	Customers	Customers	Customers	Customers	Orders Table
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	Not in ("Raleigh")			Like ("919*")	
or:					

8. After you have set your criteria, **run** the query by clicking the **Run** command on the **Design** tab.



9. The query results will be displayed in the query's **Datasheet view**, which looks like a table. If you want, **save** your query by clicking the **Save** command in the Quick Access toolbar. When prompted to name it, type the desired name and click **OK**.

City	State	Zip Code	Phone Number
Durham	NC	27714	919-555-8975
Chapel Hill	NC	27514	919-555-1096
Garner	NC	27529	919-555-1475
Cary	NC	27513	919-555-0089
Cary	NC	27513	919-555-8080
Chapel Hill	NC	27514	919-555-7025
Cary	NC	27513	919-555-3021
Durham	NC	27714	919-555-2085
Cary	NC	27513	919-555-4526
Durham	NC	27714	919-555-7070
Durham	NC	27714	919-555-0578
Chapel Hill	NC	27514	919-555-4789
Garner	NC	27529	919-555-0007
Hillsborough	NC	27278	919-555-0321

Save As

Query Name:
Customers Who've Ordered from Nearby Towns

OK Cancel

Now you know how to create a **multi-table** query. In the next lesson, we'll talk about more query design options that can make your query even more powerful.

LESSON 10 : MORE QUERY DESIGN OPTIONS

Introduction

Access offers several options that let you design and run queries that return exactly the information you're looking for. For instance, what if you need to find out **how many** of something exists within your database? Or what if you would like your query results to automatically be sorted a certain way? If you know how to use Access's query options, you can design almost any query you want.

In this lesson, you'll learn how to **modify** and **sort** your queries within Query Design view, as well as how to use the **Totals** function to create a query that can **perform calculations** with your data. You'll also learn about additional query-building options offered in Access.

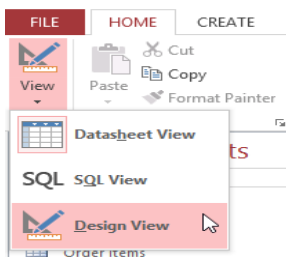
Modifying queries

Access offers several options for making your queries work better for you. In addition to **modifying** your query criteria and joins after you build your queries, you can choose to **sort** and **hide** fields in your query results.

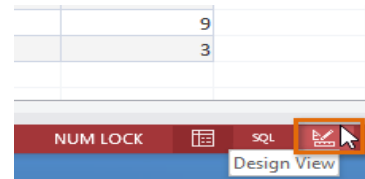
To modify your query:

When you open an existing query in Access, it is displayed in **Datasheet view**, meaning you will see your query results in a table. To modify your query, you must enter **Design view**, the view you used when creating it. There are two ways to switch to Design view:

- On the **Home** tab of the Ribbon, click the **View** command. Select **Design View** from the drop-down menu that appears.



- In the bottom-right corner of your Access window, locate the small **view icons**. Click the **Design View** icon, which is the icon farthest to the right.



Once in **Design view**, make the desired changes, then select the **Run** command to view your updated results.

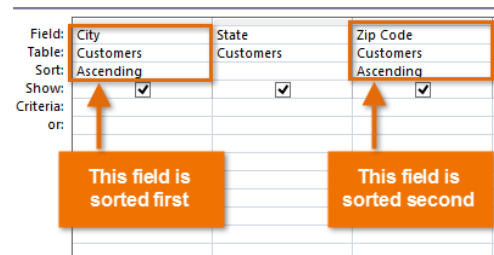
You may notice that Access also offers **SQL view**. You can ignore this. SQL view allows you to create advanced functions that you will not need to use for this tutorial or for most Access functions.

Sorting queries

Access allows you to apply multiple sorts at once while you're designing your query. This allows you to view your data exactly the way you want.

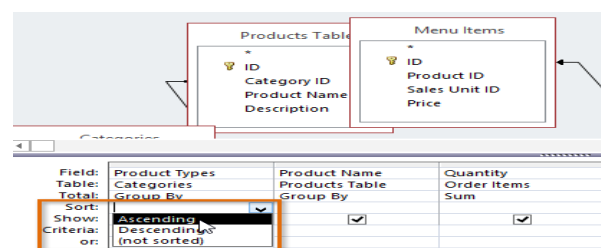
A sort that includes more than one sorted field is called a **multilevel sort**. A multilevel sort allows you to apply an initial sort, then further organize data with additional sorts. For instance, if you had a table full of customers and their addresses, you might choose to first sort the records by city, then alphabetically by last name.

When more than one sort is included in a query, Access reads the sorts from **left to right**. This means the leftmost sort will be applied first. In the example below, customers will be sorted first by the **City** where they live and then by the **Zip Code** within that city.



To apply a multilevel sort:

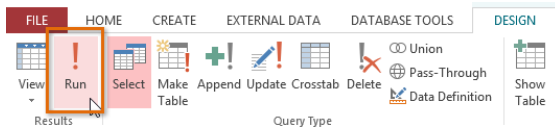
- Open the query and switch to **Design view**.
- Locate the field you want to sort first. In the **Sort:** row, click the drop-down arrow to select either an **Ascending** or **Descending** sort.



- Repeat the process in the other fields to add additional sorts. Remember, the sorts are applied from left to right, so any additional sorts must be applied to fields located **to the right** of your primary sort. If necessary, you can **rearrange** the fields by clicking the top of a field and dragging it to a new location.

Field:	Product Types	Product Name	Product Name	Quantity
Table:	Categories	Products Table	Sales Unit	Order Items
Total:	Group By	Group By	Group By	Sum
Sort:	Ascending			Descending
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				
or:				

- To apply the sort, click the **Run** command.



- Your query results will appear with the

Product Types	Product Name	SumOfQuantity
Cakes	Cheesecake	20
Cakes	Buche de Noel (Christmas Cake)- Winter	12
Cakes	Carrot Cake	9
Cakes	Black Forest	8
Cakes	Black Walnut	5
Cakes	Italian Rum	4
Cakes	Gingerbread - Winter	4
Cakes	Coconut	2
Cakes	French french vanilla	2
Cakes	German Chocolate	2
Cakes	Red Velvet	1
Cakes	Cookies n' Cream	1
Cookies	Fudge Brownie	7
Cookies	Fudge Chocolate	6
Cookies	Ginger Shortbread	6
Cookies	Chocolate Chip	5

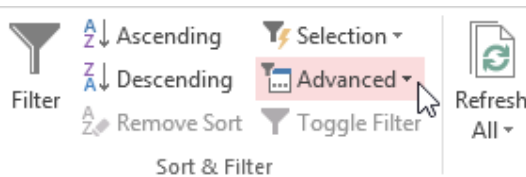
You can also apply multilevel sorts to tables that don't have

queries applied to them. On the **Home** tab on the Ribbon, select

the **Advanced** drop-down command in the **Sort & Filter** group. From the menu that appears, select **Advanced**

Filter/Sort and create the multilevel sort as you normally would. When you're finished, click the **Toggle Filter** command

to apply your sort.



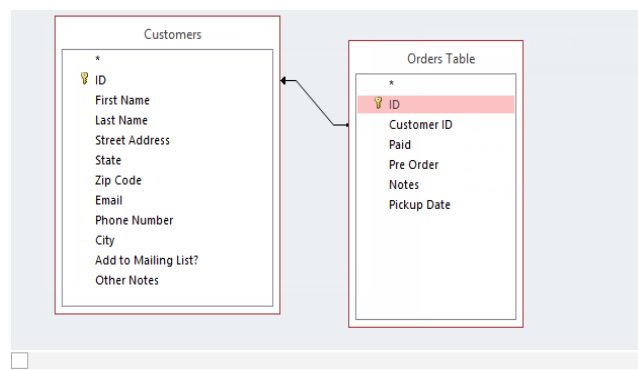
Hiding fields within queries

Sometimes you might have fields that contain important criteria, but you might not need to actually see the information from that field in the final results. For example, take one of the queries we built in our last lesson: a query to find the names and contact information of customers who had placed orders. We included Order ID numbers in our query because we wanted to make sure we only pulled customers who had placed orders.

However, we really didn't need to see this information in our final query results. In fact, if we were just looking for customer names and addresses, seeing the order number mixed in there might have been distracting. Fortunately, Access makes it easy to **hide** fields while still including any criteria they contain.

To hide a field within a query:

- Open the query and switch to **Design view**.
- Locate the field you want to hide.
- Click the **checkbox** in the **Show:** row to uncheck it.
- To see the updated query, select the **Run** command. The field will be hidden.



To **unhide** a hidden field, simply return to Design view and

click the checkbox in the field's **Show:** row again.

Field:	City	State	Zip Code	ID	Phone Number
Table:	Customers	Customers	Customers	Orders Table	Customers
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	Not In ("Raleigh")				Like ("919*")
or:					

More types of queries

By this point, you should understand how to create a simple one-table or multi-table query using multiple criteria. Additional queries offer you the ability to perform even more complex actions with your database. One of these is the **totals query**, which lets you perform calculations with your data.

Totals queries

Sometimes setting simple criteria won't give you the results you need, especially when you're working with numerical values. You may want to see your query results grouped or counted in some way. For example, let's say we want to find out **how many of each menu item at our bakery has been ordered**—how many Almond Croissants, Apple Pies, and so on. To do this, we could create a totals query to find the **sum** of the **quantities** for each item. First, the totals query will group all similar menu items from separate orders (for example, Almond Croissants). Then, the Sum function will add the values in the Quantity field to calculate the total number sold for that item.



The **Sum** function helped us find the desired information in this example, but in other situations you may need to use a different function to find the answer you need.

There are several functions you can choose from:

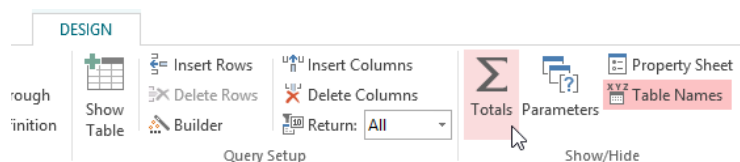
- Count:** Counts the total number of each item
- Sum:** Adds the values together
- Average:** Finds the average of the values
- Maximum:** Returns the highest value
- Minimum:** Returns the lowest value
- First:** Returns the first—or earliest—value
- Last:** Returns the last—or most recent—value

In our example above, we created a **subtotal** for each menu item in our query. If you wanted to create a **grand total** for all of the items, you would need to add a **totals row**. Review our lesson on **Modifying Tables** to learn how.

To create a totals query:

For our example, we want to find the total number we've sold of each of our menu items, so we'll use a query showing us all of the menu items we've sold. If you want to follow along in our database, open the **Menu Items Ordered** query.

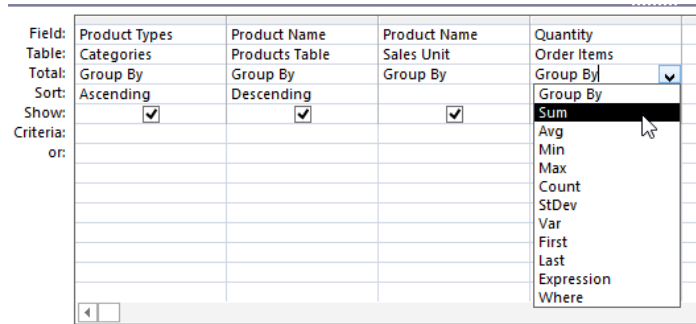
1. Create or open a query you want to use as a **totals query**.
2. From the **Design** tab, locate the **Show/Hide** group, then select the **Totals** command.



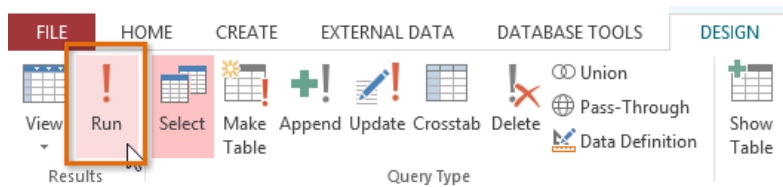
3. A row will be added to the table in the **design grid**, with all values in that row set to **Group By**. Select the cell in the **Total:** row of the field you want to perform a calculation on, then click the drop-down arrow that appears.

Field:	Product Types	Product Name	Product Name	Quantity
Table:	Categories	Products Table	Sales Unit	Order Items
Total:	Group By	Group By	Group By	Group By
Sort:	Ascending	Descending		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				
or:				

4. Select the calculation you want to be performed in that field. In our example, we want to **add** the quantities of products we've sold, so we'll select the **Sum** option.



5. When you are satisfied with your query design, select the **Run** command on the **Query Tools Design** tab to **run** the query.



6. The query results will be displayed in the query's **Datasheet view**, which looks like a table. If you want, save your query by clicking the **Save** command on the Quick Access toolbar.

Product Types -i	Products Table.Product Name -	Sales Unit.Product Name -	SumOfQuantity -i
Cakes	Cheesecake	Single	20
Cakes	Buche de Noel (Christmas Cake)- Winter	Single	12
Cakes	Carrot Cake	Single	9
Cakes	Black Forest	Single	8
Cakes	Black Walnut	Single	5
Cakes	Italian Rum	Single	4
Cakes	Gingerbread - Winter	Single	4
Cakes	Coconut	Single	2
Cakes	French french vanilla	Single	2
Cakes	German Chocolate	Single	2
Cakes	Red Velvet	Single	1
Cakes	Cookies n' Cream	Single	1
Cookies	Fudge Brownie	One Dozen	7
Cookies	Fudge Chocolate	Single	6
Cookies	Ginger Shortbread	One Dozen	6
Cookies	Chocolate Chip	Single	5

More query options

We offer mini-lessons on creating additional types of queries in the last lesson in this tutorial. Below is a list of the queries we currently cover.

- **Parameter query**
A **parameter query** allows you to create a query that can be updated easily to reflect a new criterion, or **search term**. When you open a parameter query, Access will prompt you for a search term and then show you query results that reflect that search.
- **Find duplicates query**
A **find duplicates query** lets you find all **duplicate records** in your database so you can **delete** them. Duplicate records can negatively affect the **integrity** of your database.

Other query-building resources

- Review our **Query Criteria Quick Reference Guide** for a list of criteria you can use in building queries. You can also download a **printable version** of the guide.

LESSON 11: CREATING REPORTS

Introduction

If you need to share information from your database with someone but don't want that person actually working with your database, consider creating a **report**. Reports allow you to organize and present your data in a reader-friendly, visually appealing format. Access makes it easy to create and customize a report using data from any query or table in your database.

In this lesson, you will learn how to **create**, **modify**, and **print** reports.

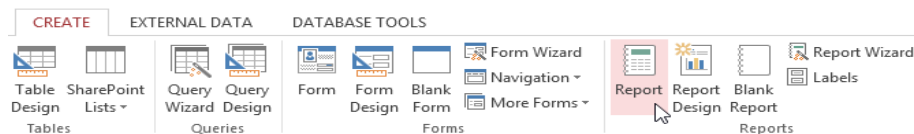
To create a report:

Reports give you the ability to present components of your database in an easy-to-read, printable format. Access lets you create reports from both **tables** and **queries**.

1. Open the table or query you want to use in your report. We want to print a list of cookies we've sold, so we'll open the **Cookies Sold** query.

Product Types	Products Table.Product Name	Sales Unit.Product Name	SumOfQuan
Cookies	Fudge Brownie	One Dozen	7
Cookies	Fudge Chocolate	Single	6
Cookies	Ginger Shortbread	One Dozen	6
Cookies	Chocolate Chip	Single	5
Cookies	Butterscotch	Single	3
Cookies	Fudge Brownie	Single	3
Cookies	Cranberry Walnut	One Dozen	3
Cookies	White Chocolate Macademia Nut	Half-Dozen	3
Cookies	Snickerdoodle	Single	3

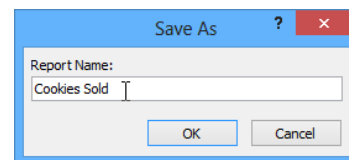
2. Select the **Create** tab on the Ribbon. Locate the **Reports** group, then click the **Report** command.



3. Access will create a new report based on your object.
4. It's likely that some of your data will be located on the other side of the **page break**. To fix this, **resize** your fields. Simply select a field, then **click** and **drag** its edge until the field is the desired size. **Repeat** with additional fields until all of your fields fit.

Product Types	[Products Table].[Product Name]	[Sales Unit].[Prodi
Cookies	Butter Pecan	One Dozen
Cookies	Butter Pecan	Single
Cookies	Butterscotch	Single
Cookies	Chocolate Banana Walnut	One Dozen
Cookies	Chocolate Banana Walnut	Single
Cookies	Chocolate Chip	Half-Dozen

5. To **save** your report, click the **Save** command on the **Quick Access toolbar**. When prompted, type a **name** for your report, then click **OK**.



Just like tables and queries, reports can be **sorted** and **filtered**.

Simply **right-click** the field you want to sort or filter, then select the desired option from the menu.

Deleting fields

You might find that your report contains some fields you don't really need to view. For instance, our report contains the **Zip Code** field, which isn't necessary in a list of orders. Fortunately, you can **delete** fields in reports without affecting the table or query where you grabbed your data.

To delete a field in a report:

1. Click any cell in the field you want to delete, then press the **Delete** key on your keyboard.

Product Types	[Products Table].[Product Name]	[Sales Unit].[Product Name]
Cookies	Butter Pecan	One Dozen
Cookies	Butter Pecan	Single
Cookies	Butterscotch	Single
Cookies	Chocolate Banana Walnut	One Dozen
Cookies	Chocolate Banana Walnut	Single
Cookies	Chocolate Chip	Half-Dozen
Cookies	Chocolate Chip	One Dozen
Cookies	Chocolate Chip	Single
Cookies	Cranberry Walnut	One Dozen

2. The field will be deleted.

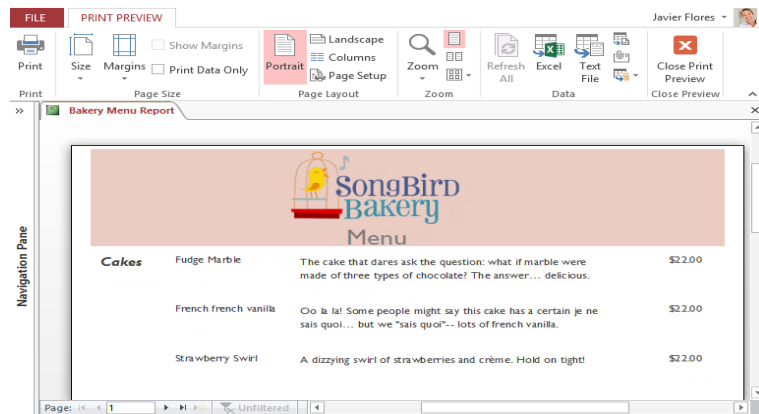
When you delete a field, be sure to delete its header as well. Simply select the header and press the **Delete** key.

Product Types	[Products Table].[Product Name]	[Sales Unit].[Product Name]
	Butter Pecan	One Dozen
	Butter Pecan	Single
	Butterscotch	Single
	Chocolate Banana Walnut	One Dozen
	Chocolate Banana Walnut	Single
	Chocolate Chip	Half-Dozen
	Chocolate Chip	One Dozen
	Chocolate Chip	Single
	Cranberry Walnut	One Dozen

Printing and saving reports in Print Preview

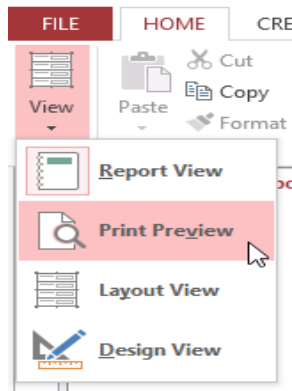
While you can print reports using commands in the **Backstage view**, you can also use **Print Preview**. Print Preview shows you how your report will appear on the printed page. It also allows you to **modify** the way your report is displayed, **print** it, and even **save** it as a different file type.

Click the buttons in the interactive below to learn about **Print Preview**.

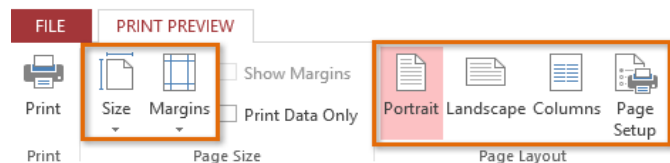


To print a report:

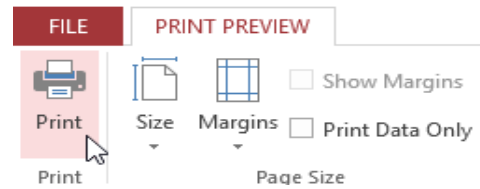
1. From the **Home** tab, click the **View** command, then select **Print Preview** from the drop-down list. Your report will be shown as it will appear on the printed page.



2. If necessary, modify the **page size**, **margin width**, and **page orientation** using the related commands on the Ribbon.



3. Click the **Print** command.
4. The **Print** dialog box will appear. Set any desired print options, then click **OK**.
5. The report will be printed.



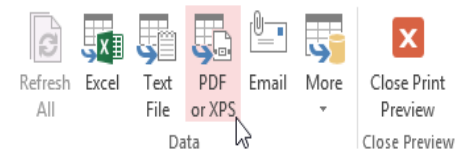
Saving reports

You can save reports in other formats so they'll be viewable outside of Access. This is called **exporting** a file, and it allows you to view and even modify reports in other formats and programs.

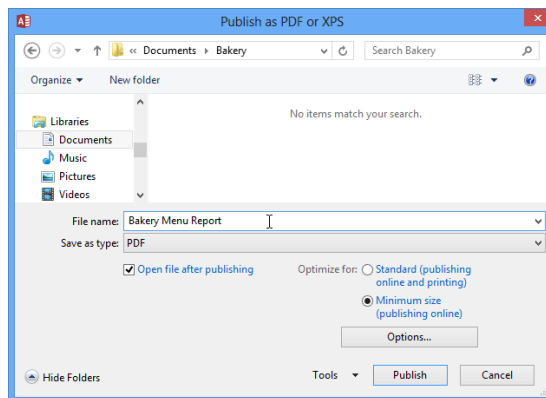
Access offers options to save your report as an **Excel file, text file, PDF, HTML document**, and more. Experiment with the different export options to find the one that best suits your needs.

To export a report:

1. From the **Home** tab, click the **View** command, then select **Print Preview** from the drop-down list.
2. Locate the **Data** group on the Ribbon.
3. Select one of the file type options, or click **More** to see options to save your report as a **Word** or **HTML** file.

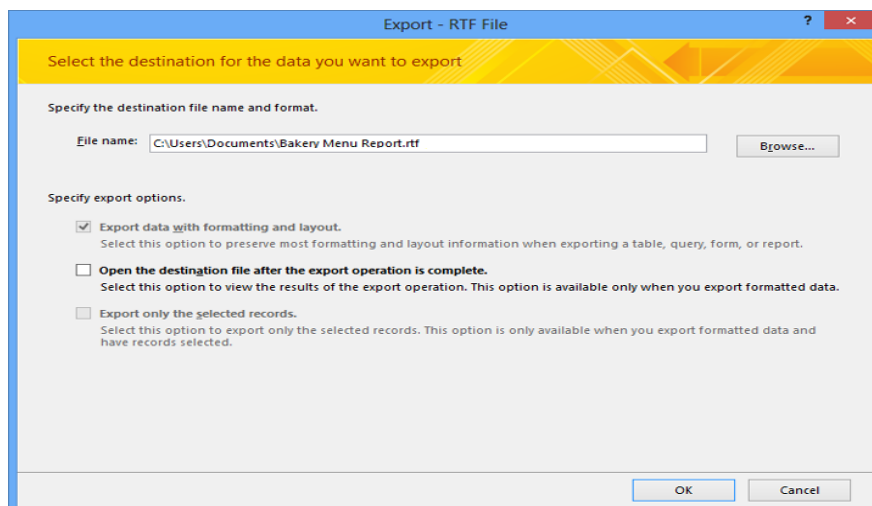


4. A dialog box will appear. Select the **location** where you want to save the report.
5. Enter a **file name** for the report, then click **Publish**.



6. A dialog box will appear to notify you that your file has been successfully saved. Click **Close** to return to your report.

Some export options will cause the **Export Wizard** to appear. Simply follow the instructions to export your report.



LESSON 12: ADVANCED REPORT OPTIONS

Introduction

Access offers several **advanced options** for creating and modifying reports. The **Report Wizard** is a tool that guides you through the process of creating complex reports. Once you've created a report—whether through the Report Wizard or the Report command—you can then **format** it to make it look exactly how you want.

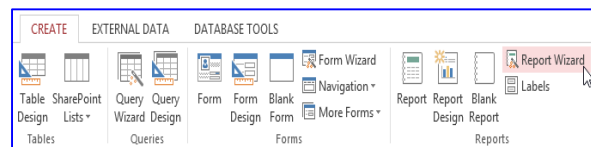
In this lesson, you'll learn how to use the **Report Wizard** to create complex reports. You'll also learn how to use Access's **formatting options** to **format text**, **change report colors** and **fonts**, and **add a logo**.

The Report Wizard

While using the **Report** command is a quick way to create reports from the current object, it's not as helpful if you want to create a report with data from multiple objects. The **Report Wizard** makes it easy to create reports using fields from multiple tables and queries. It even lets you choose how your data will be organized.

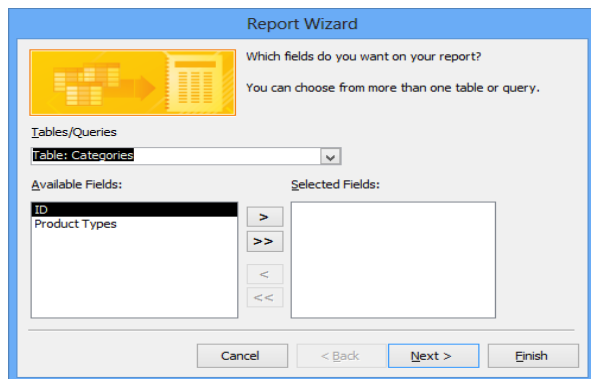
To create a report with the Report Wizard:

1. Select the **Create** tab and locate the **Reports** group. Click the **Report Wizard** command.
2. The **Report Wizard** will appear. In the procedures below, we'll discuss the different pages in the Report Wizard.

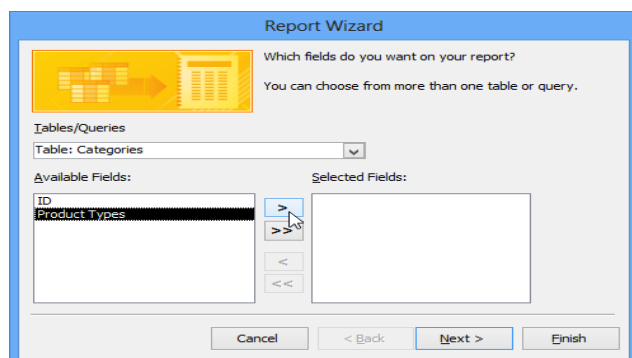
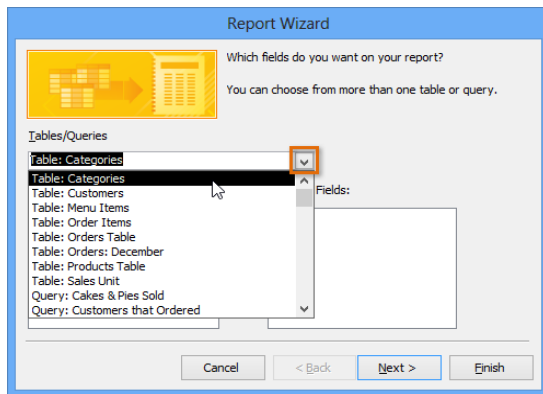


Step 1: Select the fields to include in your report

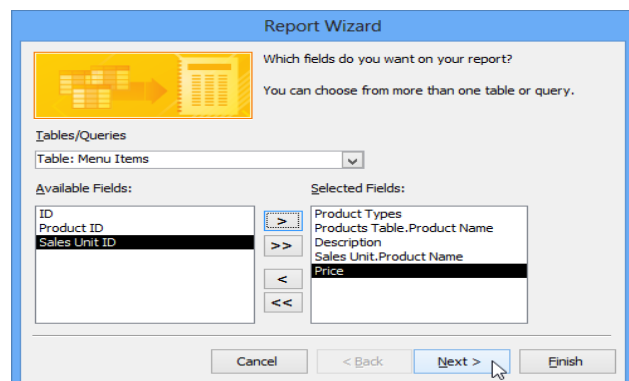
1. Click the **drop-down arrow** to select the table or query that contains the desired field(s).



2. Select a field from the list on the left, and click the **right arrow** to add it to the report.



3. You can add fields from more than one table or query by repeating the above steps. Once you've added the desired fields, click **Next**.

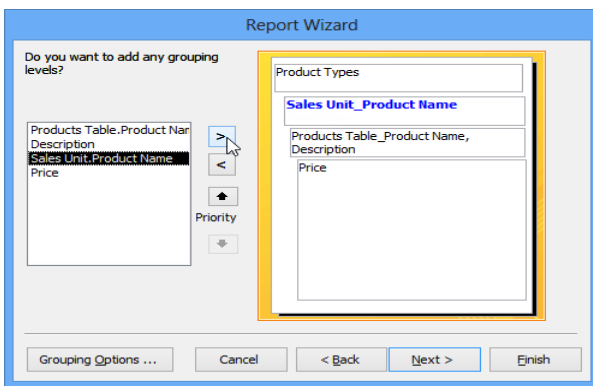
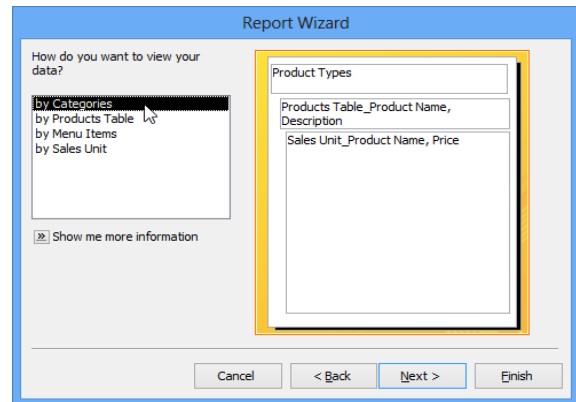


Step 2: Organize the report

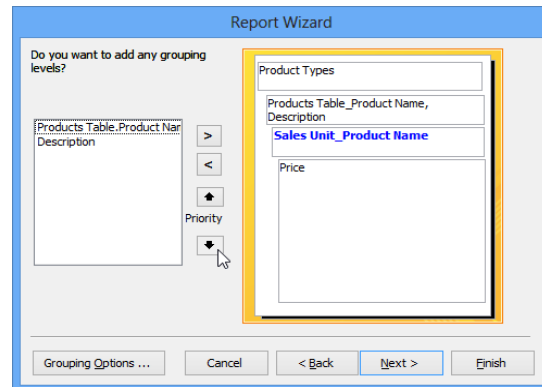
The Report Wizard will provide you with options that let you choose how to view and organize your data. These options **group** similar data within your fields and **organize** these fields into multiple levels, like in an outline or bulleted list.

If you are building a report from only one table or query, you can skip to **Step 3** below.

1. Access will offer a list of several organization options. Select an option from the list to preview it.
2. Click **Next** when you are satisfied with the basic organization of your data.
3. If you're not satisfied with the way your data is organized, you can now modify the grouping levels. Select a field from the list, and click the **right arrow** to add it as a new level.



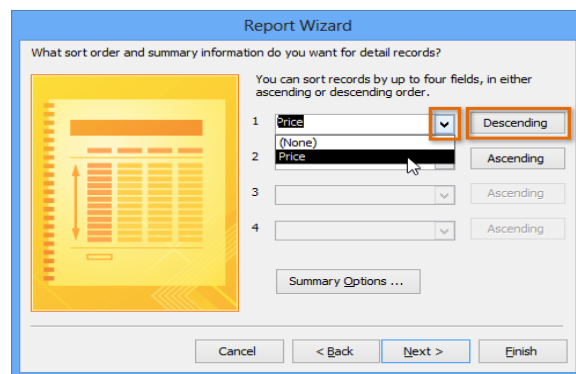
4. If necessary, modify the order of your grouped fields by selecting a field and clicking the **up** or **down Priority** arrow to move it up or down a level.



5. Once you are satisfied with the organization of your report, click **Next**.

Step 3: Sort your report data

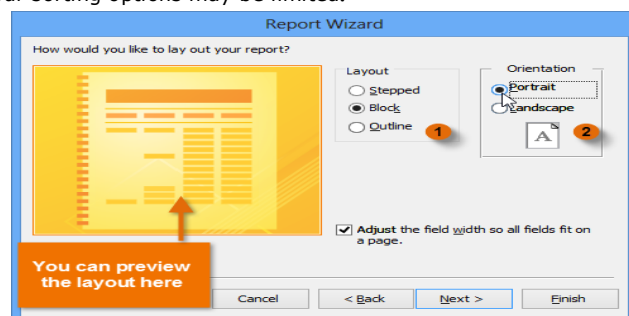
1. Click the top drop-down arrow, and select the name of the first field you want to sort.
2. Click the button on the right to change the sort to **ascending** or **descending**.
3. Add any additional sorts. You can sort up to **four fields**. The sort will be applied from top to bottom, meaning the sort at the top of the list will be the main sort.
4. When you are satisfied with the way your data is sorted, click **Next**.



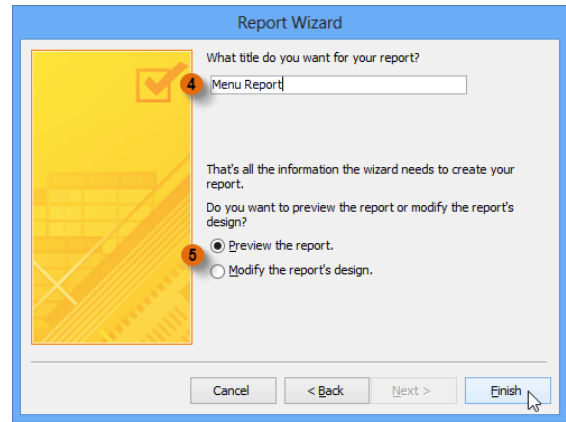
Depending on the grouping you have chosen for your data, your sorting options may be limited.

Step 4: Select a layout and title

1. Click the various layout options to see how they look, then **select** one to use in your report.
2. Select either a **portrait** (tall) or **landscape** (wide) orientation for your report.



- Once you are satisfied with your report layout, click **Next**.
- Select the text box, and type the **title** you would like for your report.
- Select whether you want to **preview** the report or **modify** its design, then click **Finish**.
- Your report will be created and saved.



You may have to adjust your field and row **size** and **location** to make sure your data looks the way you want it to. To do this, you'll need to switch to **Design view**. When you're done, switch back to **Report view** to see your changes.

Formatting reports

One of the strengths of reports is that you can modify their appearance to make them look how you want. You can add **headers** and **footers**, apply new **colors**, and even add a **logo**. All of these things can help you create visually appealing reports.

Modifying report text

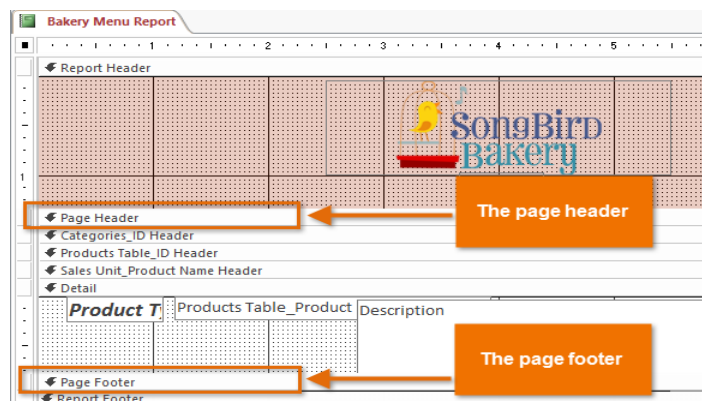
The bulk of the information in your report comes straight from the query or table you built it from, which means you can't edit it within the report. However, you can change, add, or delete label text, headers, and footers to make your report clearer and easier to read. For example, in our report we decided we didn't need the field headings to understand our data, so we simply **deleted** them.

Just like other Office programs, Access allows you to modify the text color and font, add shapes, and more. If you're not sure how to perform basic text and shape formatting, visit the **Formatting Text** and **Shapes** lessons from our **Word 2013** tutorial.

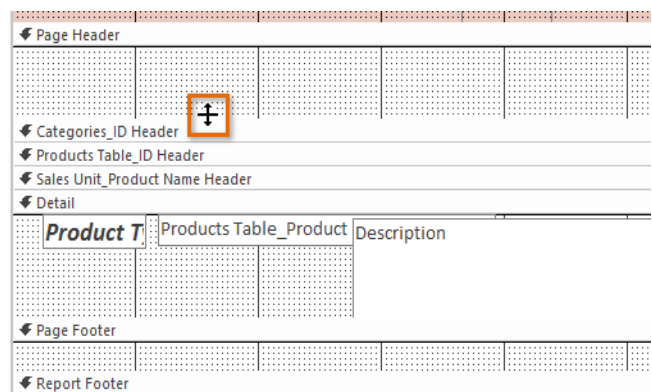
Cakes	Strawberry Swirl	A dizzying swirl of strawberries and crème. Hold on tight!	\$22.00
Cakes	Cookies n' Cream	Like dipping oreos and milk, but a cake, and not at all soggy.	\$22.00
Cakes	Lemon	A simple classic-- sweet and sour.	\$22.00
Cakes	German Chocolate	"Chocolate" in German is "schokolade." You don't have to be able to say that to enjoy this cake.	\$22.00

Modifying the page header and footer

To view and modify the **header** and **footer** that appear on each page of your report, select the **View** command on the Ribbon and switch to **Design view**. The header and footer are located in the white space beneath the **Page Header** and **Page Footer** bars.

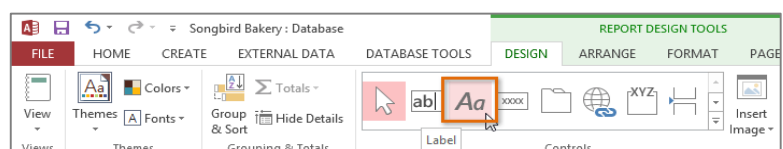


Depending on your report's design, sometimes you may find that there is no white space in the page header and footer, as in the image above. If this is the case, you must **resize** the header and footer before you can add anything to them. Simply click and drag the bottom border of the header or footer to make it larger.

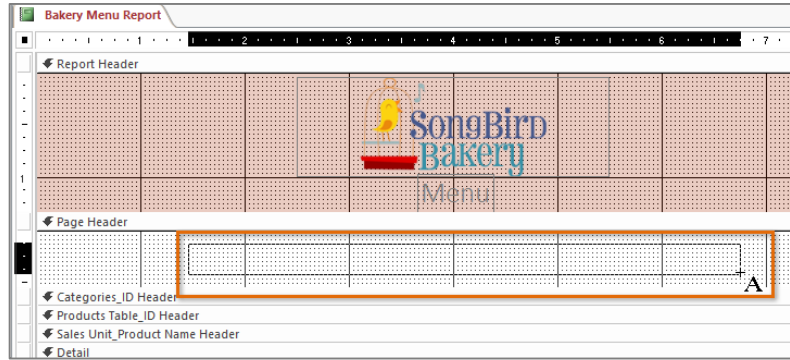


To add text to a header or footer:

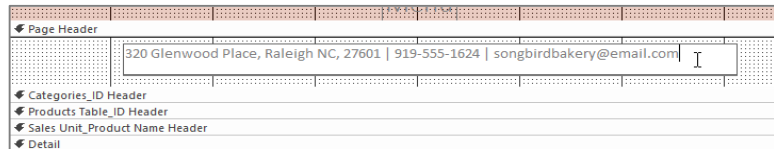
- Select the **Design** tab, locate the **Controls** group, and click the **Label** command.



2. Click and drag the mouse inside the white area to create your label. Release the mouse when it is the desired size.

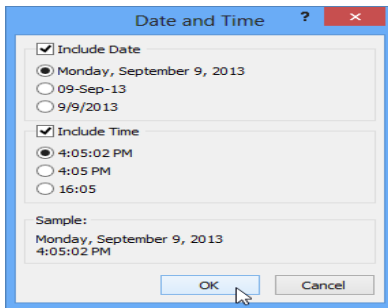
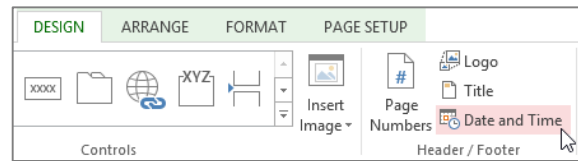


3. Click the text box, and type the desired text.



To add the date and time to a header or footer:

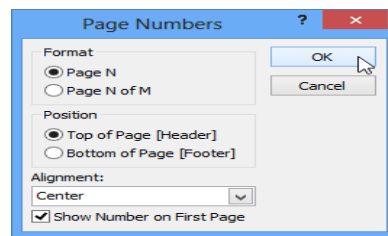
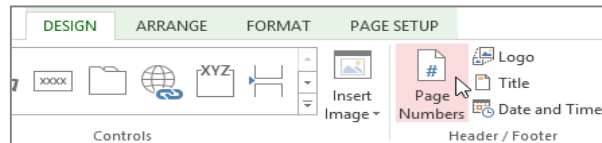
1. Select the **Design** tab, locate the **Header/Footer** group, and click the **Date and Time** command.
2. A dialog box will appear. Select the desired formatting options. A preview of the text that will be included in your report will appear.
3. When you are satisfied with the appearance of the date and time, click **OK**.



By default, the date and time appear in the **header**. If you would like to move them to the **footer** instead, simply click the date and time boxes and drag them to the desired location.

To add page numbers to a header or footer:

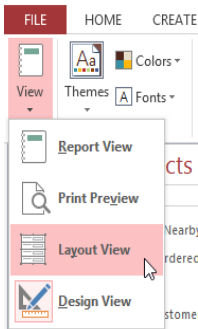
1. Select the **Design** tab, then locate the **Header/Footer** group.
2. Click the **Page Numbers** command.
3. The **Page Numbers** dialog box will appear. Select the **format** of the page numbers.
 - o Select **Page N** to display the number of only the current page.
 - o Select **Page N of M** to display the number of the current page and the number of total pages.
4. Under **Position**, choose **Top of Page** or **Bottom of Page** to control where the page numbers appear.
5. Click the drop-down arrow to select the **alignment** of the page numbers.
6. When you are satisfied with the settings, click **OK**.



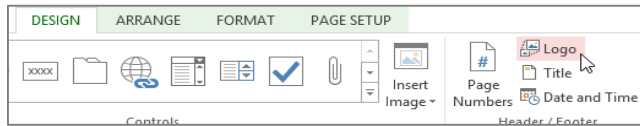
Modifying your report's appearance

To add a logo:

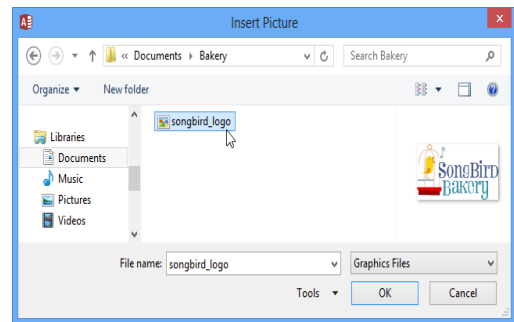
1. From the **Design** tab, click the **View** command, then select **Layout View** from the drop-down list.



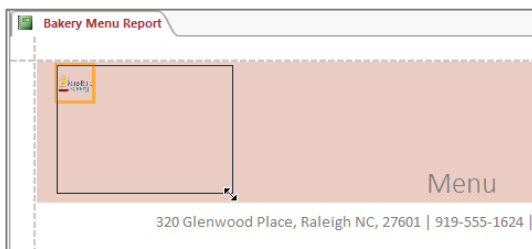
2. Locate the **Header/Footer** group, then click the **Logo** command.



3. A dialog box will appear. Locate and select the desired file, then click **OK** to add it to your report.



4. A small version of the image will appear in the header. Click and drag the image border to resize it.



5. If necessary, move your logo to the desired location by clicking and dragging it.

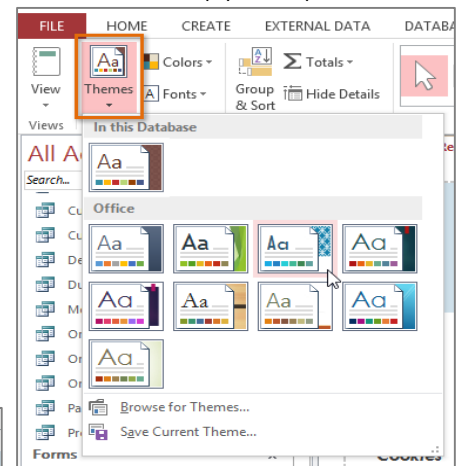


Themes and fonts

A **theme** is a set of **colors** and **fonts** that applies to the **entire database** to give it a consistent, professional look. By default, databases use the Office theme. When you change the theme, all of the theme fonts and colors in your database change to match the new theme. Designing and modifying reports using theme elements can help you keep the appearance of your reports consistent.

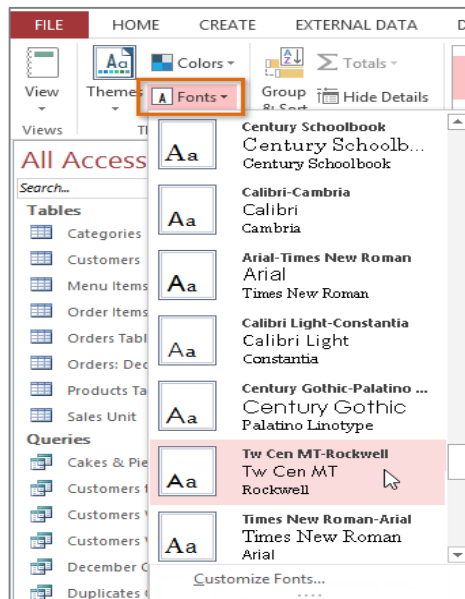
To change the theme:

1. Select the **Design** tab, locate the **Themes** group, and click the **Themes** command.
2. A drop-down menu will appear. Select the desired theme.
3. The theme will be applied to your entire database.



To change the theme fonts:

1. Select the **Design** tab, locate the **Themes** group, and click the **Fonts** command.
2. A drop-down menu will appear. Select a set of **theme fonts**.



3. The fonts will be applied to your entire database.



LESSON 13: MODIFYING TABLES

Introduction

After working with your database, you might find that you need to make some changes to the tables that store your data. Access makes it easy to modify your tables to suit your database's needs.

In this lesson, you'll learn how to **create** and **rearrange** table fields. You'll also learn how to ensure your table data is correctly and consistently formatted by setting **validation rules**, **character limits**, and **data types** in your fields. Finally, we will direct you to additional options for performing simple math functions within your tables.

Modifying tables

In addition to making basic modifications to your tables, like **adding** and **moving** fields, you can make more advanced modifications that let you set **rules** for your data. All of these changes can help make your tables even more useful.

Adding and rearranging fields

Access makes it easy to rearrange existing fields and add new ones. When you add a new field, you can even set the **data type**, which dictates which **type** of data can be entered into that field.

There are several types of fields you can add to a table:

- **Short Text:** This is the default option and is best for most text in Access. You should also choose it for numbers you don't plan to do math with, like postal codes and phone numbers.
- **Number:** This is best for numbers you might want to do calculations with, like quantities of an item ordered or sold.
- **Currency:** This automatically formats numbers in the currency used in your region.
- **Date & Time:** This allows you to choose a date from a pop-out calendar.
- **Yes/No:** This inserts a checkbox into your field.
- **Rich Text:** This allows you to add formatting to text, like **bold** and *italics*.
- **Long Text:** This is ideal for large amounts of text, like product descriptions.
- **Attachment:** This allows you to attach files, like images.
- **Hyperlink:** This creates a link to a URL or email address.

To add a new field to an existing table:

1. Open the desired table, then click the header with the text **Click to Add**. If you already have several fields, you may need to scroll all the way to the right to see this option.
2. A drop-down menu will appear. Select the desired **data type** for the new field.

Phone Number	Click to Add
919-555-2314	
919-555-4534	
919-555-4564	
919-555-7653	
919-555-8658	
919-555-5112	
919-555-5460	
919-555-5753	
919-555-9745	
919-555-8975	

Phone Number	Click to Add
919-555-2314	
919-555-4534	
919-555-4564	
919-555-7653	
919-555-8658	
919-555-5112	
919-555-5460	
919-555-5753	
919-555-9745	
919-555-8975	
919-555-2332	
919-555-1123	
919-555-3432	
919-555-5467	
919-555-7755	

3. Type a name for your field, then press the **Enter** key.

Phone Number	Add to Mailing List	Click to Add
919-555-2314	<input type="checkbox"/>	
919-555-4534	<input type="checkbox"/>	
919-555-4564	<input type="checkbox"/>	
919-555-7653	<input type="checkbox"/>	
919-555-8658	<input type="checkbox"/>	
919-555-5112	<input type="checkbox"/>	
919-555-5460	<input type="checkbox"/>	
919-555-5753	<input type="checkbox"/>	
919-555-9745	<input type="checkbox"/>	
919-555-8975	<input type="checkbox"/>	
919-555-2332	<input type="checkbox"/>	
919-555-1123	<input type="checkbox"/>	
919-555-3432	<input type="checkbox"/>	

To move a field:

1. Locate the field you want to move, then hover your mouse over the **bottom border** of the field header. The cursor will become a four-sided arrow.
2. Click and drag the field to its new location.

Email	Phone Number	Add to Mailing List	Click to Add
beck@email.com	919-555-2314	<input checked="" type="checkbox"/>	
lugeo@email.com	919-555-4534	<input type="checkbox"/>	
texj@email.com	919-555-4564	<input type="checkbox"/>	
newkb@email.com	919-555-7653	<input type="checkbox"/>	
lo@email.com	919-555-8658	<input type="checkbox"/>	
denqui@email.com	919-555-5112	<input type="checkbox"/>	
dhoda@email.com	919-555-5460	<input type="checkbox"/>	
hall@email.com	919-555-5753	<input type="checkbox"/>	
mshill@email.com	919-555-9745	<input type="checkbox"/>	
rogki@email.com	919-555-8975	<input checked="" type="checkbox"/>	

3. Release the mouse. The field will appear in the new location.

Email	Add to Mailing List	Phone Number	Click to Add
beck@email.com	<input checked="" type="checkbox"/>	919-555-2314	
lugeo@email.com	<input type="checkbox"/>	919-555-4534	
texj@email.com	<input type="checkbox"/>	919-555-4564	
newkb@email.com	<input type="checkbox"/>	919-555-7653	
lo@email.com	<input type="checkbox"/>	919-555-8658	
denqui@email.com	<input type="checkbox"/>	919-555-5112	
dhoda@email.com	<input type="checkbox"/>	919-555-5460	
hall@email.com	<input type="checkbox"/>	919-555-5753	
mshill@email.com	<input type="checkbox"/>	919-555-9745	
rogki@email.com	<input checked="" type="checkbox"/>	919-555-8975	

Email	Add to Mailing List	Phone Number	Click to Add
beck@email.com	<input checked="" type="checkbox"/>	919-555-2314	
lugeo@email.com	<input type="checkbox"/>	919-555-4534	
texj@email.com	<input type="checkbox"/>	919-555-4564	
newkb@email.com	<input type="checkbox"/>	919-555-7653	
lo@email.com	<input type="checkbox"/>	919-555-8658	
denqui@email.com	<input type="checkbox"/>	919-555-5112	
dhoda@email.com	<input type="checkbox"/>	919-555-5460	
hall@email.com	<input type="checkbox"/>	919-555-5753	
mshill@email.com	<input type="checkbox"/>	919-555-9745	
rogki@email.com	<input checked="" type="checkbox"/>	919-555-8975	

Advanced field options

On the previous page, you learned about setting the **data type** for new fields. When you set field data type, you are really setting a **rule** for that field. Databases often include rules because they help ensure users enter the correct type of data.

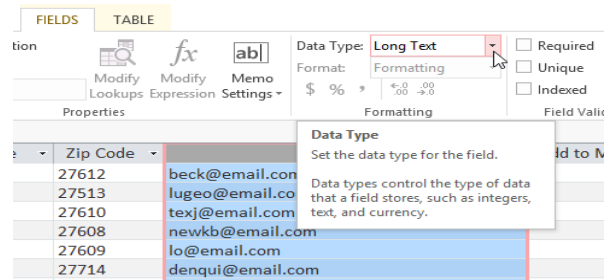
Why is this important? Computers aren't as smart as humans about certain things. While you might recognize that **two** and **2** or **NC** and **North Carolina** are the same thing, Access will not and therefore won't group these things together. Making sure to enter your data in a standard format will help you better organize, count, and understand it.

Rules can also determine which options you have for working with your data. For example, you can only do math with data entered in **number** or **currency** fields, and you can only format text entered into **text** fields.

There are three main types of rules you can set for a field: **data type**, **character limit**, and **validation rules**.

To change the data type for existing fields:

1. Select the field whose data type you want to change.
2. Select the **Fields** tab, then locate the **Formatting** group. Click the **Data Type** drop-down arrow.
3. Select the desired data type.



4. The field data type will be changed. Depending on the data type you chose, you may notice changes to your information. For instance, because we set the data type for the **Email** field to **Hyperlink** all of the email addresses in the field are now clickable links.

Zip Code	Email	Add to Mailing List
27612	beck@email.com	<input type="checkbox"/>
27513	lugeo@email.com	<input type="checkbox"/>
27610	texj@email.com	<input type="checkbox"/>
27608	newkb@email.com	<input type="checkbox"/>
27609	lo@email.com	<input type="checkbox"/>
27714	denqui@email.com	<input type="checkbox"/>

State	Zip Code	Email
NC	27612	beck@email.com
NC	27513	lugeo@email.com
NC	27610	texj@email.com
NC	27608	newkb@email.com
NC	27609	lo@email.com
NC	27714	denqui@email.com
NC	27513	dhoda@email.com
NC	27612	hall@email.com
NC	27606	mshill@email.com
NC	27714	rogki@email.com

You shouldn't change field data type unless you are certain your field data is in the correct format for the new data type. Changing a field containing only text to the **Number** type, for instance, will delete all of your field data. This process is often irreversible.

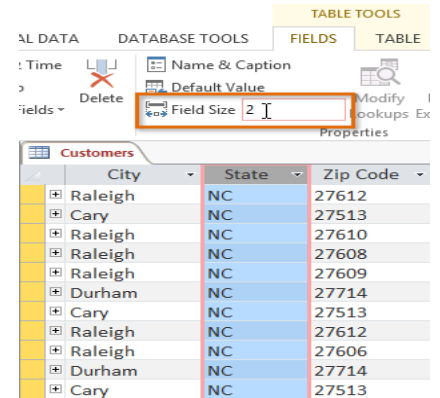
Field character limits

Setting the **character limit** for a field sets a rule about how many characters—letters, numbers, punctuation, and even spaces—can be entered into that field. This can help to keep the data in your records concise and even force users to enter data a certain way.

In the example below, a user is entering records that include addresses. If you set the character limit in the **State** field to **2**, users can only enter **two characters** of information. This means they must enter postal abbreviations for the states instead of the full name—here, NC instead of North Carolina. Note that you can only set a character limit for fields defined as text.

To set a character limit for a field:

1. Select the desired field.
2. Click the **Fields** tab, then locate the **Properties** group.
3. In the **Field Size** box, type the maximum number of characters you want to allow in your field.
4. **Save** your table. The character limit for the field will be set.



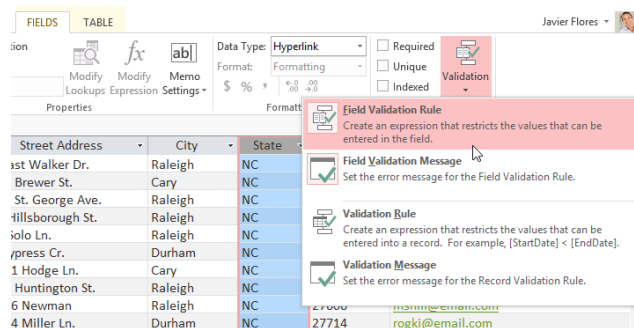
Validation rules

A **validation rule** is a rule that dictates which information can be entered into a field. When a validation rule is in place, it is impossible for a user to enter data that violates the rule. For example, if we were asking users to input a state name into a table with contact information, we might create a rule that limits the valid responses to U.S. state postal codes. This would prevent users from typing something that wasn't actually a real state postal code.

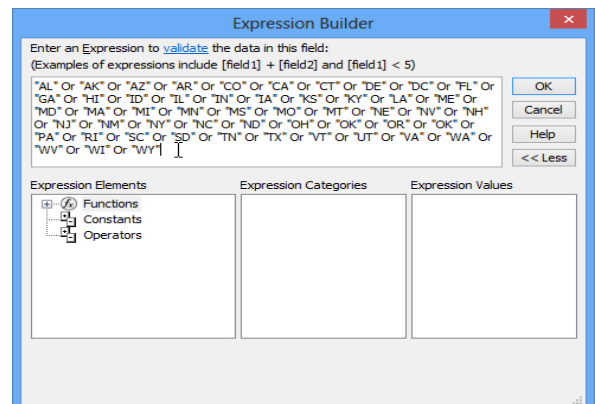
In the example below, we will apply that rule to our **Customers** table. It's a fairly simple validation rule—we'll just name all of the valid responses a user could enter, which will mean users can't type anything else into the record. However, it's possible to create validation rules that are much more complex. For detailed information on how to write validation rules, review this tutorial from Microsoft on **creating validation rules**.

To create a validation rule:

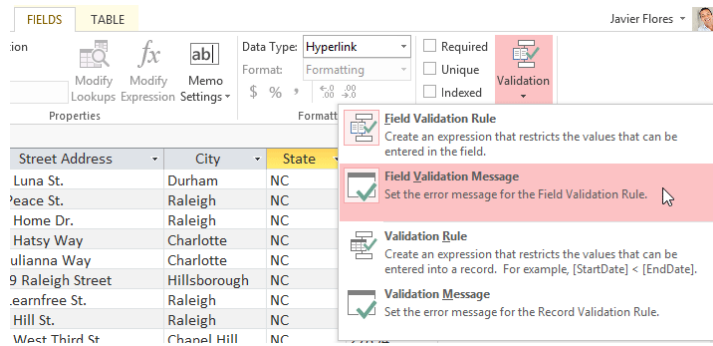
1. Select the field you want to add a validation rule to. In our example, we'll set a rule for the **State** field.
2. Select the **Fields** tab, then locate the **Field Validation** group. Click the **Validation** drop-down, then select **Field Validation Rule**.



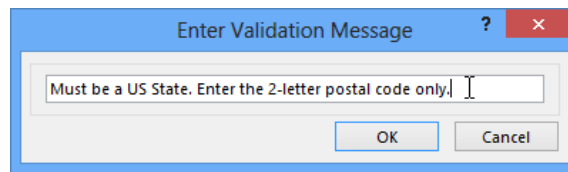
3. The **Expression Builder** dialog box will appear. Click the text box and type your validation rule. In our example, we want to limit data in the **State** field to actual state postal codes. We'll type each of the valid responses in quotation marks and separate them with the word **Or**, which lets Access know that this field can accept the response "AL" **Or** "AK" **Or** "AZ" or any of the other terms we've entered.
4. Once you're satisfied with the validation rule, click **OK**. The dialog box will close.



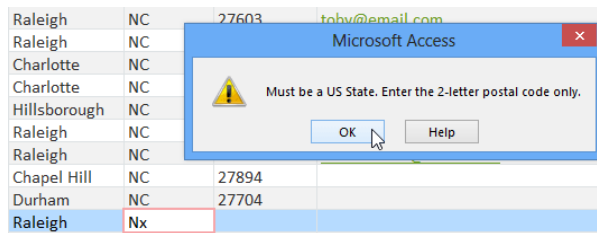
- Click the **Validation** drop-down command again. This time, select **Field Validation Message**.



- A dialog box will appear. Type the phrase you want to appear in an **error message** when users try to enter data that **violates** the validation rule. Your message should let them know what data is permitted.



- When you're satisfied with the error message, click **OK**.
- The validation rule is now included in the field. Users will be unable to enter data that violates the rule.



Simple validation rules can be written exactly like **query criteria**. The only difference is that query criteria search for data, while an identical validation rule either **permits** or **rejects** data. To see examples of query criteria, review our **Query Criteria Quick Reference Guide**.

More table options

Calculated fields and totals rows

Adding **calculated fields** and **totals rows** to your table lets you perform **calculations** using your table data. A calculated field calculates data within one record, while a totals row performs a calculation on an entire field of data. Whenever you see a **subtotal** for one record, you are looking at a calculated field. Similarly, whenever you see a **grand total** at the bottom of a table, you're looking at a totals row.

Product	Quantity	"Unit"	Price	Subtotal
German Chocolate	1	Single	\$22.00	\$22.00
German Chocolate	1		\$22.00	\$22.00
Chocolate Banana Walnut	1		\$19.00	\$19.00
Fudge Chocolate	3		\$22.00	\$66.00
Fudge Chocolate	2		\$22.00	\$44.00
Chocolate Chip	2		\$1.50	\$3.00
Chocolate Chip	3	Single	\$1.50	\$4.50
Chocolate Chip	1	Half-Dozen	\$7.50	\$7.50
Chocolate Chip	2	One Dozen	\$14.00	\$28.00
White Chocolate Macademia Nut	3	Half-Dozen	\$10.50	\$31.50
White Chocolate Macademia Nut	2	One Dozen	\$19.00	\$38.00
Fudge Chocolate	1	Single	\$22.00	\$22.00
Chocolate Banana Walnut	1		\$2.00	\$2.00
Chocolate Amaretto	2		\$24.00	\$48.00
Chocolate Amaretto	1		\$24.00	\$24.00
Total				\$371.50

To learn how to create **calculated fields** and **totals rows**, review our lesson on **Creating Calculated Fields and Totals Rows**.

LESSON 14: CREATING FORMS

Introduction

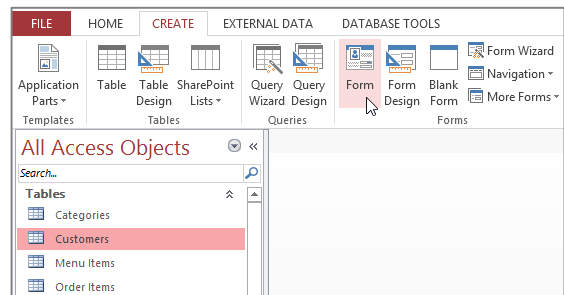
Creating **forms** for your database can make entering data more convenient. When you create a form, you can design it in a way that works with your database and that makes sense to you.

In this lesson, you will learn how to **create** and **modify** forms. You'll also learn how to use form options like **design controls** and **form properties** to make sure your form works exactly the way you want.

To create a form:

Access makes it easy to create a **form** from any table in your database. Any form you create from a table will let you **view the data** that's already in that table and **add new data**. Once you've created a form, you can modify it by adding additional fields and **design controls** like combo boxes.

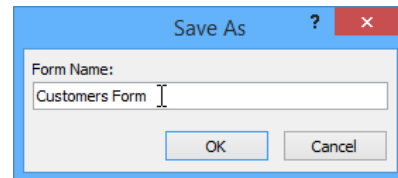
1. In the Navigation pane, select the table you want to use to create a form. You do not need to open the table.
2. Select the **Create** tab, locate the **Forms** group, and click the **Form** command.
3. Your form will be created and opened in **Layout view**.



The screenshot shows the 'Customers' form in Layout view. The form contains several text boxes for data entry: ID (1), First Name (Tracey), Last Name (Beckham), Street Address (7 East Walker Dr.), State (NC), and Zip Code (27612). There are also checkboxes for 'Add to Mailing List?' and 'Field1'. At the bottom, there is a table-like view showing a list of records with columns for ID, Paid, Pre Order, and Notes.

ID	Paid	Pre Order	Notes
38	Yes	No	
(New)	No	No	

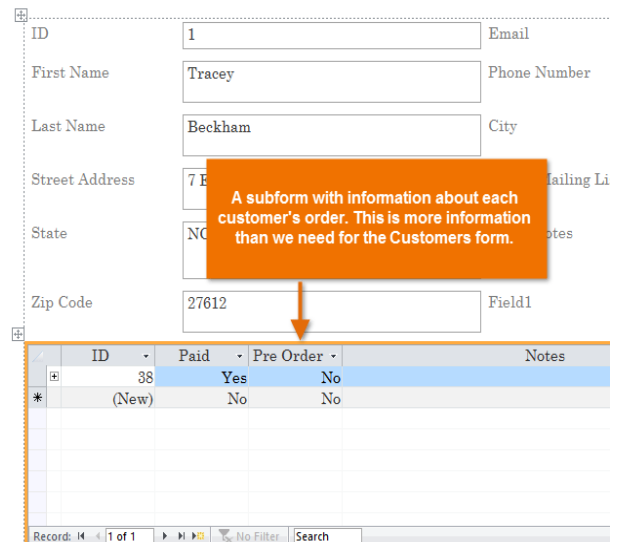
4. To **save** the form, click the **Save** command on the **Quick Access toolbar**. When prompted, type a **name** for the form, then click **OK**.



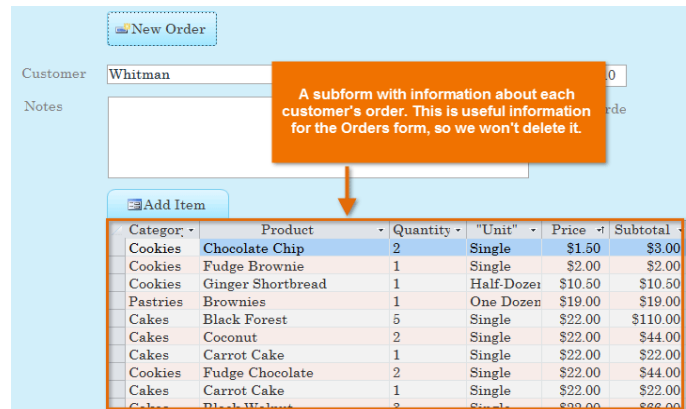
About subforms

If you created a form from a table whose records are linked to another table, your form probably includes a **subform**. A subform is a **datasheet form** that displays linked records in a table-like format. For instance, the subform included in the **Customers** form we just created displays linked customer **orders**.

We probably don't need to include this subform because we just want to use the Customers form to enter and review contact information. If you find that you don't need a subform, you can easily **delete** it. To do this, simply click it and press the **Delete** key on your keyboard.



However, subforms are often quite helpful. Depending on the content and source of your form, you might find that the subform contains useful information, like in the example below. In our **Orders** form, the subform contains the name, quantity, and price of each item contained in the order, which is all useful information.

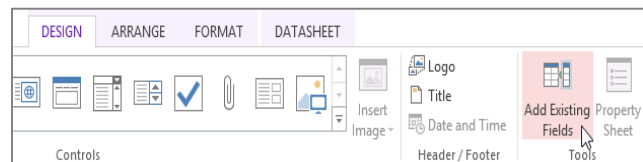


Adding additional fields to a form

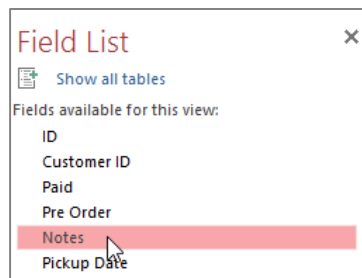
When you use the **Form** command on an existing table, all of the fields from the table are included in the form. However, if you later add additional fields to the table, these fields will **not** automatically show up in existing forms. In situations like this, you can **add** additional fields to a form.

To add a field to a form:

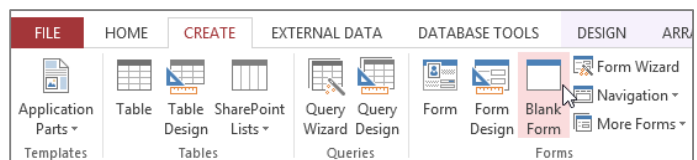
1. Select the **Design** tab, then locate the **Tools** group on the right side of the Ribbon.
2. Click the **Add Existing Fields** command.
3. The **Field List** pane will appear. Double-click the desired field(s).



4. The field will be added.



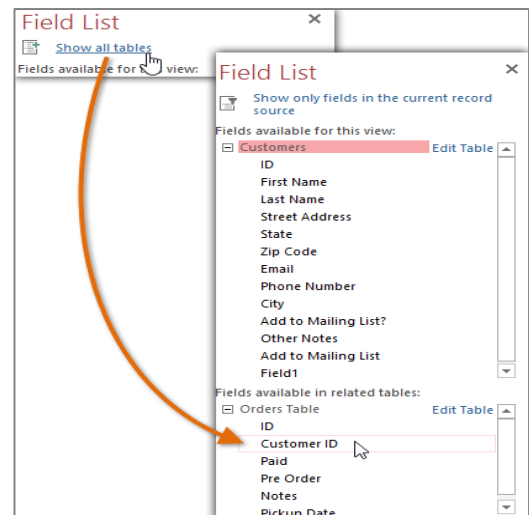
You can also use the above procedure to add fields to a totally blank form. Simply **create a form** by clicking the **Blank Form** command on the **Create** tab, then follow the above steps to add the desired fields.



To add a field from a different table:

You can also add fields from **different tables** in your database to the form.

1. From the **Field List** pane, click **Show All Tables**.
2. Click the plus sign **+** next to the table that contains the field you want to add, then double-click the desired field.
3. The new field will be added.



Adding design controls

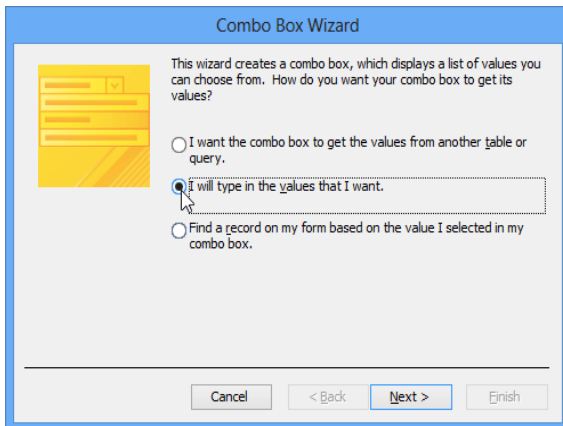
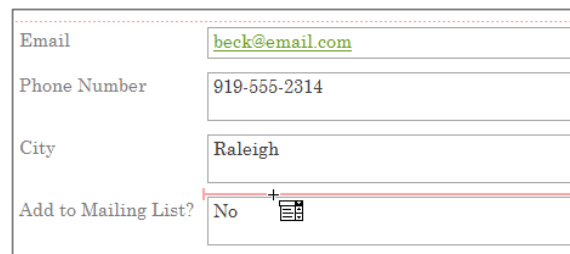
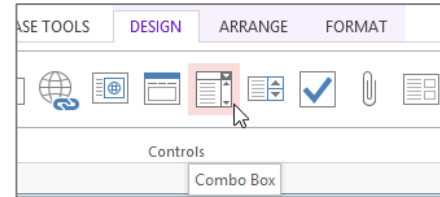
Design controls set restrictions on the fields in your forms. This helps you better control how the data is entered into your forms, which in turn helps keep the database consistent.

Combo boxes

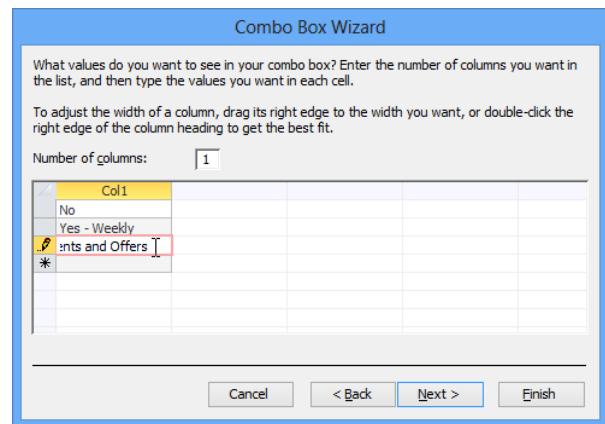
A **combo box** is a drop-down list you can use in your form in place of a field. Combo boxes **limit** the information users can enter by forcing them to select only the **options** you have specified. Combo boxes are useful for fields that have a limited number of possible valid responses. For instance, you might use a combo box to make sure people only enter a valid U.S. state while entering an address, or that they only choose products that already exist in your database while placing an order.

To create a combo box:

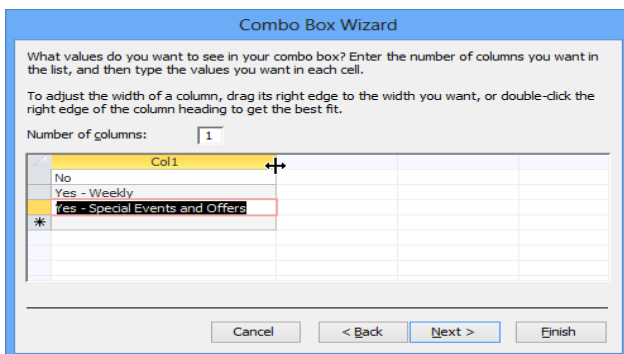
1. In **Form Layout** view, select the **Design** tab, then locate the **Controls** group.
2. Select the **Combo Box** command, which looks like a drop-down list.
3. Select the desired location for the combo box. A line will appear to indicate the location where your combo box will be created. In our example, we'll place it between the **City** field and the **Add to Mailing List?** fields.
4. The **Combo Box Wizard** dialog box will appear. Select **I will type in the values that I want**, then click **Next**.



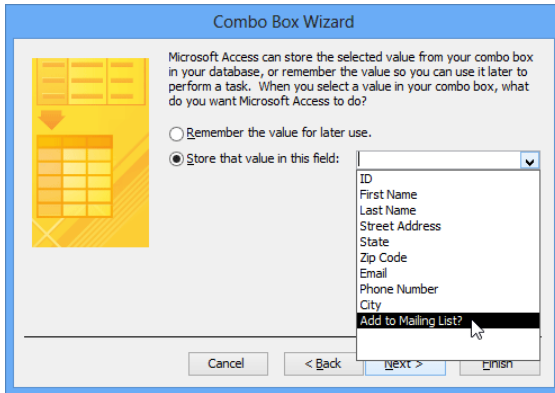
5. Type the choices you want to appear in your drop-down list. Each choice should be on its own row. In our example, we are creating a combo box for the **Add to Mailing List?** field in our form, so we will enter all of the possible valid responses for this field. Users will be able to select one of three choices from our finished combo box: **No**, **Yes - Weekly**, and **Yes - Special Events and Offers**.



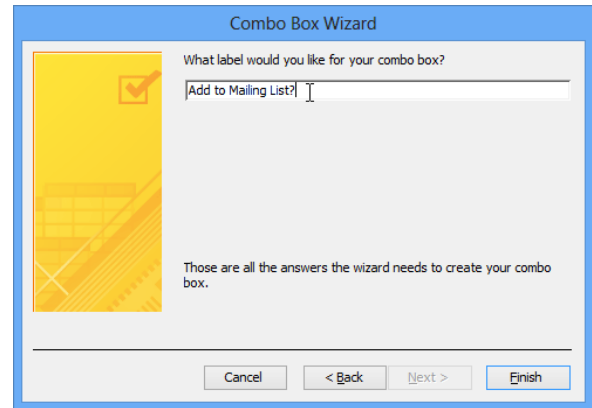
6. If necessary, **resize** the column so all of your text is visible. Once you are satisfied with your list, click **Next**.



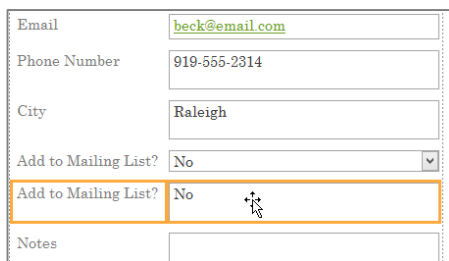
- Select **Store that value in this field**, then click the drop-down arrow and **select the field** where you want selections from your combo box to be recorded. After making your selection, click **Next**.



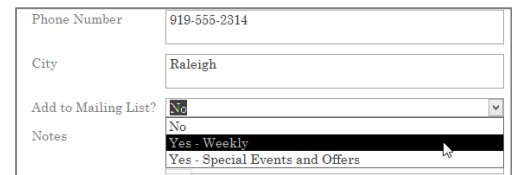
- Enter the **label**—or **name**—that will appear next to your combo box. Generally, it's a good idea to use the name of the field you chose in the previous step.



- Click **Finish**. Your combo box will appear on the form. If you created your combo box to **replace** an existing field, you should **delete** the first field. In our example, you might notice that we now have two fields with the same name. These two fields send information to the same place, so we don't need them both. We'll **delete** the one without the combo box.

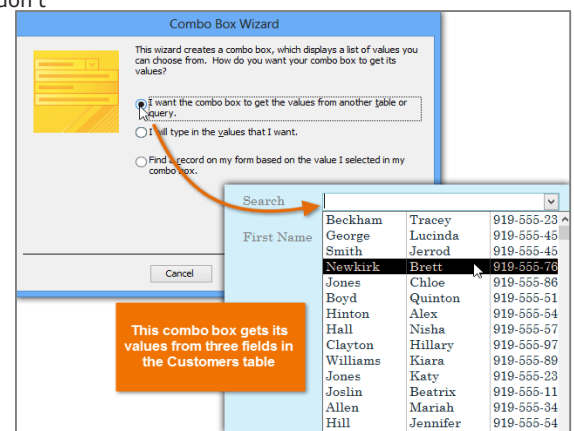


- Switch to **Form** view to **test** your combo box. Simply click the drop-down arrow and verify that the list contains the correct choices. The combo box can now be used to enter data.



If you want to include a drop-down list with a long list of options and don't want to type all of them out, create a combo box and choose the first option in the Combo Box Wizard, **I want to get the values from another table or query**. This will allow you to create a drop-down list from a table field.

Some users report that Access malfunctions while working with forms. If you have a problem performing any of these tasks in Layout view, try switching to Design view.

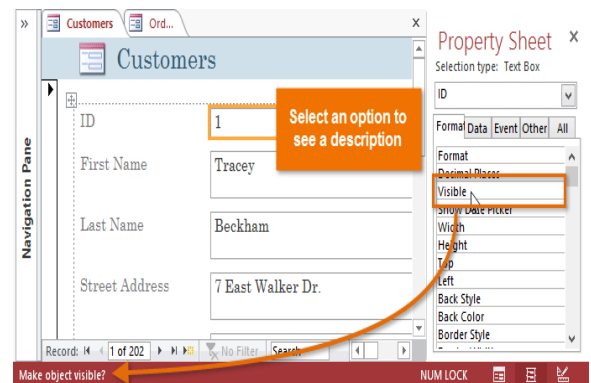


Customizing form settings with the Property Sheet

The **Property Sheet** is a pane containing **detailed information** on your form and each of its components. From the Property Sheet, you can make changes to every part of your form, both in terms of function and appearance.

The best way to familiarize yourself with the Property Sheet is to **open** it and **select** various options. When you select an option, Access will display a brief description of the option on the **bottom-left border** of the program window.

Pay close attention as you modify your form and its fields. It's possible to make subtle changes with the Property Sheet. Because there are so many options, it can sometimes be difficult to remember which one you used to modify each aspect of your form.

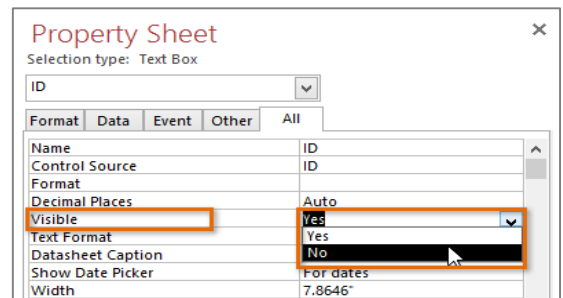
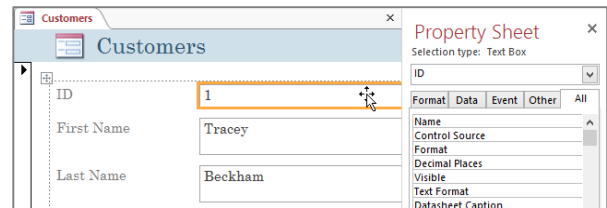


Modifying form settings

There are far too many options in the Property Sheet to discuss them all in detail. We'll review two useful ones here: **hiding** fields, and setting fields with **dates** to **automatically fill in the current date**. Practicing these procedures should also give you a sense of how to work with other Property Sheet settings.

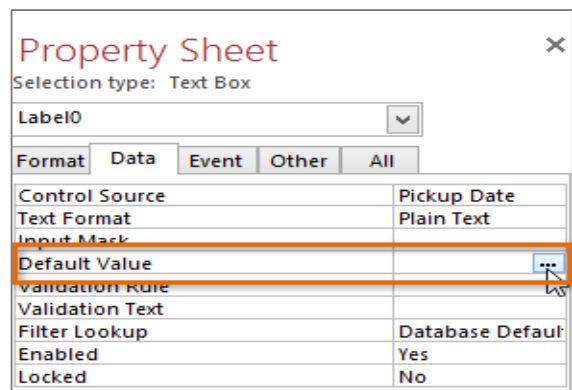
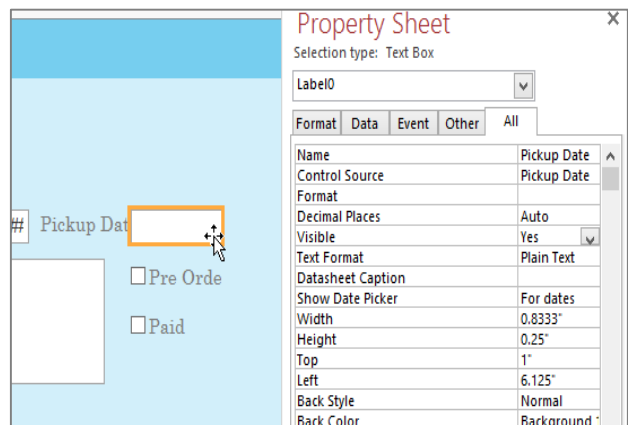
To hide a field:

1. In either **Layout** or **Design** view, select the **Design** tab, then locate the **Tools** group. Click the **Property Sheet** command.
2. The **Property Sheet** will appear in a pane on the right. On the form, **select** the field you want to hide. In our example, we'll hide the **Customer ID** field because we don't want any of our users to try to edit it.
3. In the **Property Sheet**, click the **All** tab, then locate the **Visible** option on the fifth row.
4. Click the drop-down arrow in the column to the right, then select **No**.
5. Switch to **Form** view to verify that the field is hidden.

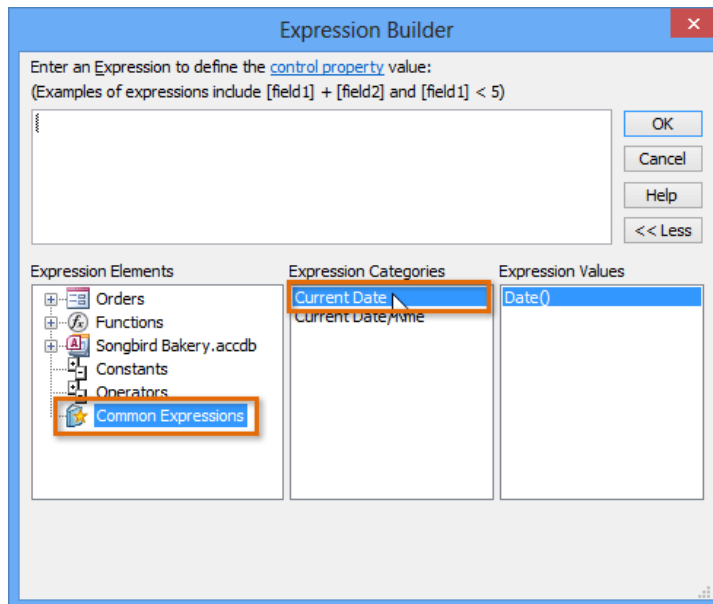


To set a field to auto fill with the current date:

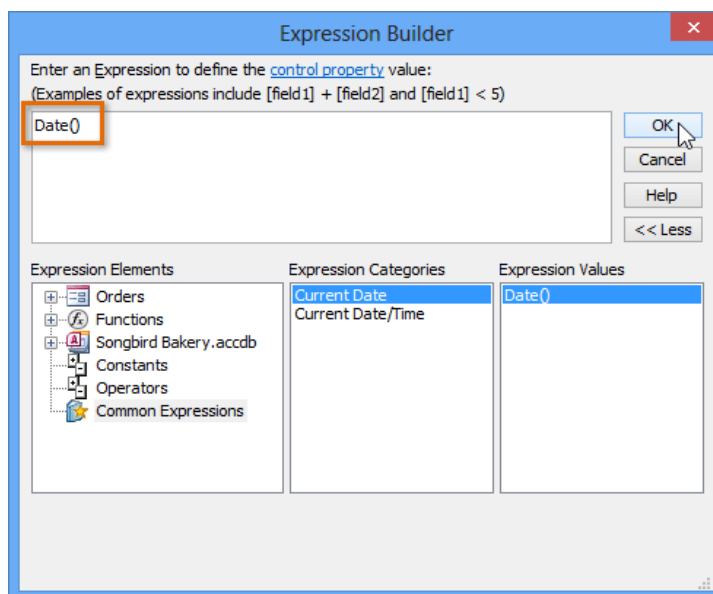
1. In either **Layout** or **Design** view, select the **Design** tab, then locate the **Tools** group. Click the **Property Sheet** command.
2. The **Property Sheet** will appear in a pane on the right. On the form, **select** the field you want to automatically fill in the current date. This **must** be a field with the **date** data type. For our example, we'll select the **Pickup Date** field on our **Orders** form.
3. In the **Property Sheet**, click the **Data** tab, then select the **Default Value** field in the fourth row. Click the **Expression Builder** button that appears in the column to the right.



- The **Expression Builder** dialog box will appear. In the **Expression Elements** list, click the words **Common Expressions**.
- In the **Expression Categories** list, double-click **Current Date**.



- The expression for **Current Date** will be added. Click **OK**.



- Switch to **Form** view to verify that the expression works. When you create a **new record** with that form, the date field you modified will automatically fill in the current date.

The screenshot shows a form titled 'New Order'. It has a 'Customer' dropdown menu, an 'Order #' field with '###' as a placeholder, and a 'Pickup Date' field filled with '9/10/2013'. Below these are 'Notes' and two checkboxes: 'Pre Orde' and 'Paid'.

LESSON 15: FORMATTING FORMS

Introduction

After creating a form, you might want to modify its appearance. **Formatting** your forms can help make your database look consistent and professional. Some formatting changes can even make your forms easier to use. With the formatting tools in Access, you can customize your forms to look exactly the way you want.

In this lesson, you will learn how to **add command buttons**, **modify form layouts**, **add logos** and **other images**, and change form **colors** and **fonts**.

Formatting forms

Access offers several options that let you make your forms look exactly the way you want. While some of these options—like **command buttons**—are unique to forms, others may be familiar to you.

Command buttons

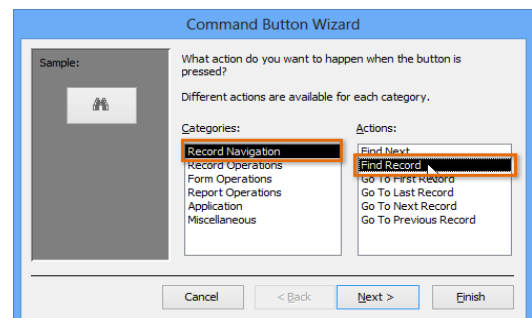
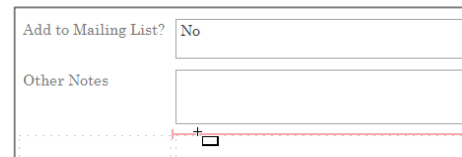
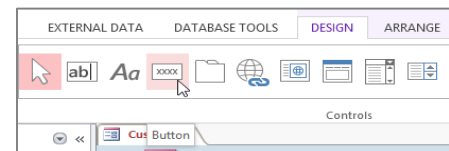
If you want to create a way for users of your form to quickly perform specific actions and tasks, consider adding **command buttons**. When you create a command button, you specify an action for it to carry out when clicked. By including commands for common tasks right in your form, you're making the form easier to use.

Access offers many different types of command buttons, but they can be divided into a few main categories:

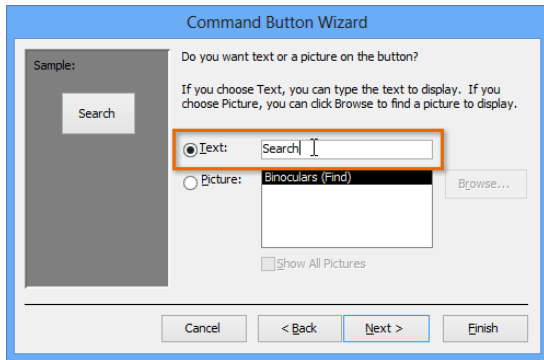
- **Record Navigation** command buttons, which allow users to move among the records in your database
- **Record Operation** command buttons, which let users do things like save and print a record
- **Form Operation** command buttons, which allow users to quickly open or close a form, print the current form, and perform other actions
- **Report Operation** command buttons, which offer users a quick way to do things like preview or mail a report from the current record

To add a command button to a form:

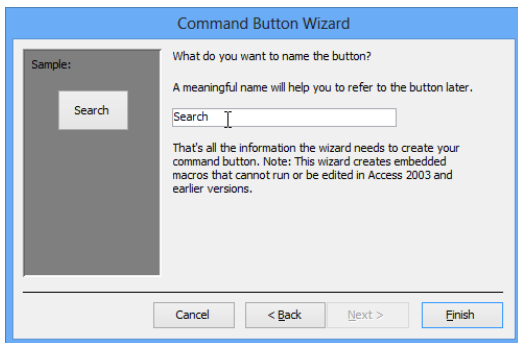
1. In **Form Layout** view, select the **Design** tab, then locate the **Controls** group.
2. Click the **Button** command.
3. Choose the desired location for the command button, then click the mouse.
4. The **Command Button Wizard** will appear. In the **Categories** pane, select the category of button you want to add. We want to find a way to move more quickly to specific records, so we'll choose the **Record Navigation** category.
5. The list in the **Actions** pane will update to reflect your chosen category. Select the action you want the button to perform, then click **Next**. In our example, we'll choose **Find Record**.
6. You can now decide whether you want your button to include **text** or a **picture**. A live preview of your button appears on the left.



- To include **text**, select the **Text** option, then type the desired word or phrase into the text box.

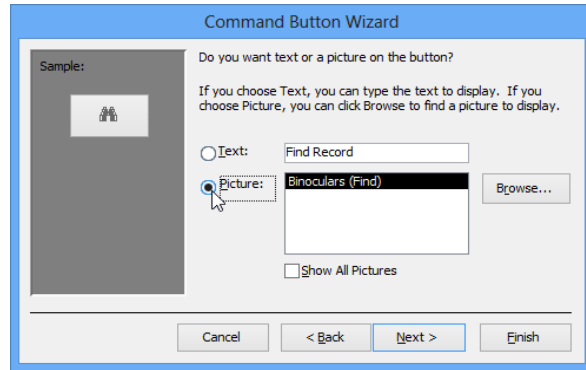


- When you are satisfied with the appearance of your command button, click **Next**.
- Type a **name** for the button. This name won't appear on the button, but knowing the name will help you quickly identify the button if you ever want to **modify** it with the **Property Sheet**. After adding the button name, click **Finish**.

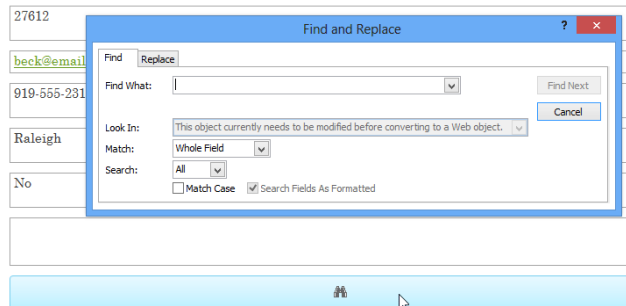


and **Replace** dialog box.

- To include a **picture**, select the **Picture** option. You can decide to keep the default picture for that command button or select another picture. Click **Show All Pictures** to choose from another command button icon or **Browse** to choose a picture from your computer.



- Switch to **Form** view to test the new button. Our **Search** button opens the **Find**



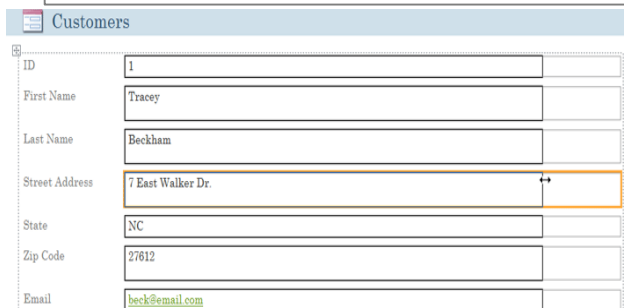
Some users report that Access malfunctions while formatting forms. If you have a problem performing any of these tasks in Layout view, try switching to Design view.

Modifying form layout

When you create a form, Access arranges the form components in a default layout where the fields are neatly stacked up on top of each other, all exactly the same width. While this layout is functional, you might find that it doesn't best fit your information. For instance, in the form below most of the fields are almost completely empty because the data stored there doesn't take up much room.

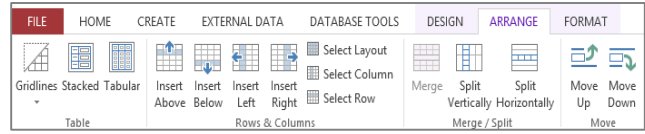


The form would fit the data better if we made the fields and command buttons smaller and even put some of them side by side. However, with the default layout, you won't be able to put two fields next to each other or resize one field or button without resizing all of them. This is because Access lines up form components in rows and columns. When you resize a field, you're really resizing the column that contains it.



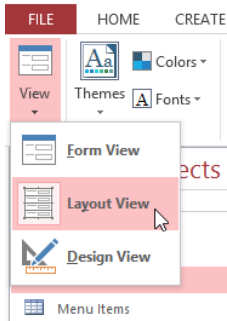
To resize and rearrange our fields the way we want, we'll have to **modify the form layout**. For instance, because the default layout for our form contains only two columns—one for the **field labels** and another for the **fields**—we would have to **create a new column** to put two fields side by side.

We can do this using the command on the **Arrange** tab, which contains all of the tools we'll need to customize a form's layout. If you've ever built and modified **tables** in Microsoft Word, you already know how to use most of these tools. If you're unfamiliar with tables, review our **Tables** lesson from our **Word 2013** tutorial.

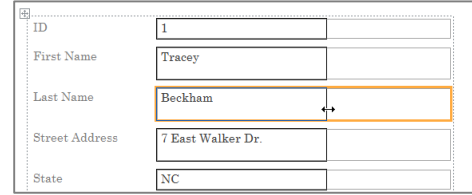


To resize form components:

1. Switch to **Layout view**.

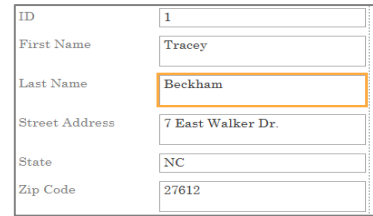


2. Select the field or button you want to resize, then **hover your mouse** over the edge. Your cursor will become a double-sided arrow.



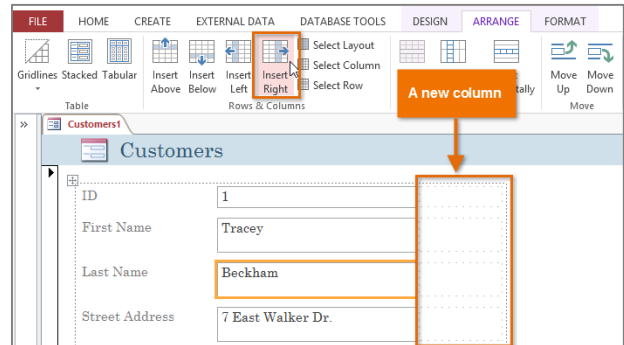
1. Click and drag the mouse to resize the selected object.

2. The field or button will be resized, as well as every other item aligned with it.

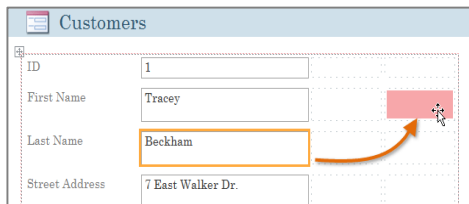


To move form components:

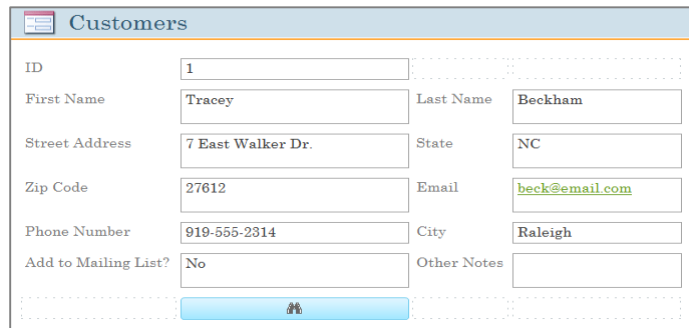
1. If necessary, **add columns or rows** to make room for the field or button you want to move by using the **Insert** commands in the **Rows & Columns** group. In our example, we want to move the **Last Name** field to the right of the **First Name** field, so we'll have to create two new columns to the right: one for the field label, and one for the field itself. To do this, we'll click the **Insert Right** command twice.



2. Click and drag the field or button to its new location. If you're moving a field, make sure to move the **field label** as well.



3. Repeat the steps above for any other fields or buttons you want to move.



If you want to make a field take up **more** or **less** space than one column, you can use the **Merge** and **Split** commands. The Merge command **combines** two or more cells, while the **Split** command **divides** a cell.

More formatting options

Adding **logos** and other images to your forms can greatly improve their appearance, as can applying **theme colors and fonts**. To learn how to add images and theme elements to your forms, review our **Advanced Report Options** lesson. The procedures are identical.

Modifying the colors and fonts of form components

To further customize the appearance of your forms, you can apply different **colors** and **fonts** to individual fields, buttons, labels, and other form components. Modifying form appearance this way is useful if you want to use a certain color or font scheme in a form but don't want these design elements to apply to your entire database.

For instance, in the form below we modified the **font** of our form **title**. We also applied a new **fill** and **border color** to the form **fields** and are doing the same with the **command buttons**.

You can make these changes using formatting techniques and tools similar to the ones you would use to modify shapes in Word and other Office programs. Simply select the object you want to modify while in **Layout** or **Design** view and use the formatting options on the **Format** tab to customize its appearance. For more information on making these types of modifications, review the **Shapes** lesson from our **Word 2013** tutorial.

LESSON 16: DESIGNING YOUR OWN DATABASE

Introduction

Now that you know how to use and modify existing databases, you might be interested in designing your own. Database design can be very complicated—so complicated, in fact, that people take extensive courses just to learn how to plan them. For this reason, we haven't focused on creating a database from scratch. However, we can help you get started.

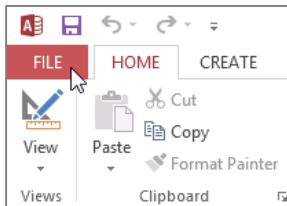
In this lesson, you will learn how to create a database from an existing **template**. You will also learn about other resources you can use to understand database design.

To create a database from a template:

Before deciding to build your own database, you may want to look at the **templates** included in Access to see if any of them match your needs. When you select a template, Access creates a **new database** based on that template. Once it's created, you can fill the database with your own information or modify it to suit your needs.

Note that many Access 2013 templates require you to save the database online with **Microsoft SharePoint**. Many businesses use SharePoint to share files at work.

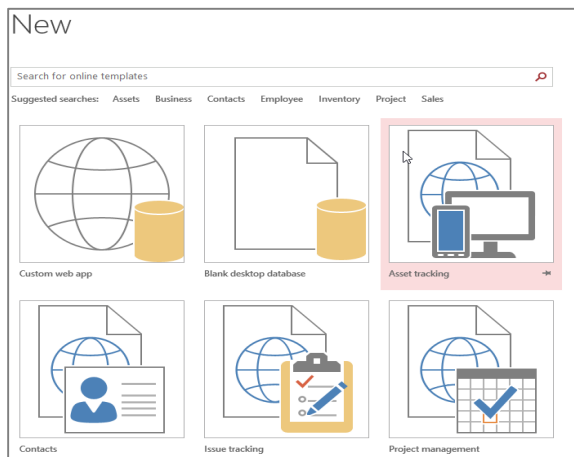
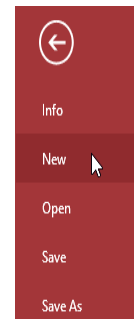
1. Select the **File** tab. This will take you to **Backstage view**.



2. Click **New**.

3. Several templates will appear below the **Blank desktop database** option. You can also click a **suggested search** to find templates or use the **search bar** to find something more specific.

4. Select a **template** to review it.



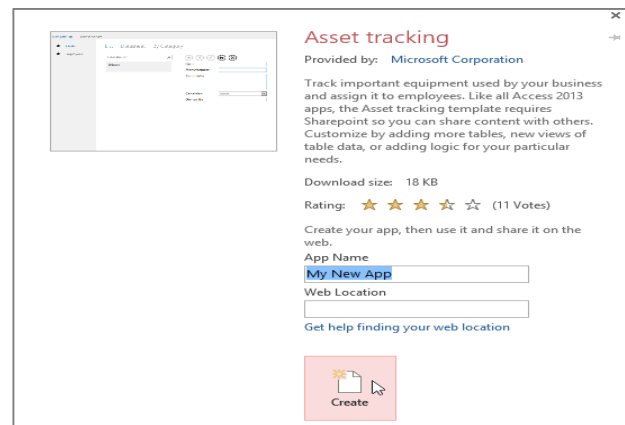
1. A **preview** of the template will appear, along with **additional information** on how the template can be used.
2. Click **Create** to use the selected template.
3. A new database will appear with the **selected template**.

It's important to note that not all templates are created by Microsoft. Many are created by third-party providers and even individual users, so some templates may work better than others.

More resources on database design

Unfortunately, it can be difficult to find free, high-quality resources on database design. On this page, you'll find links to online resources that can help you get started. You can also contact your local library for instructional books and resources.

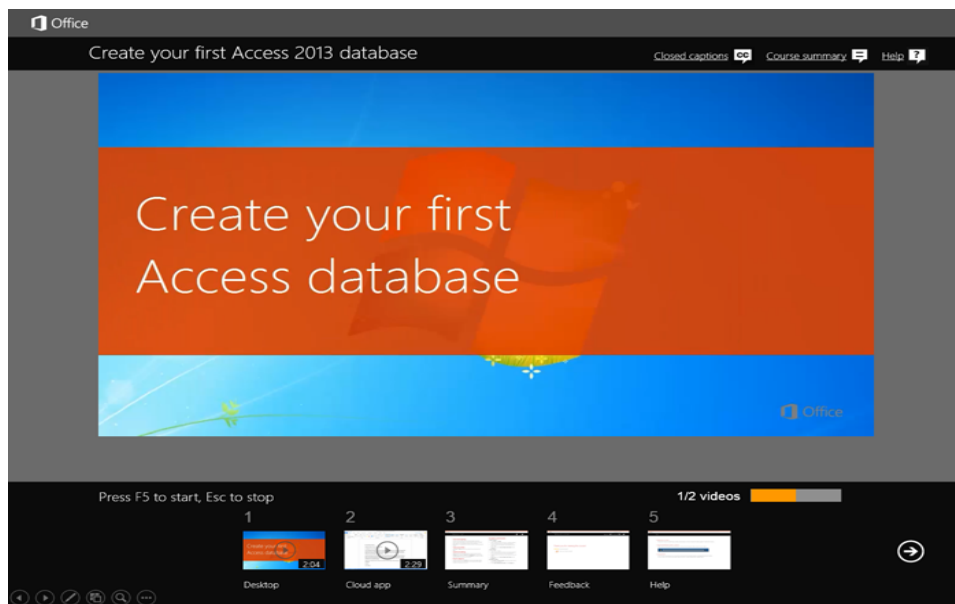
If you are willing to devote even more time and resources to learning database design, you can also search for **Access 2013 Certification Classes** online and in your community. However, as a rule, these classes are not free.



Free online resources

- **Create Your First Access 2013 Database**

This tutorial series from Microsoft offers basic guidance on database design and creation.



About.com: Building an Access Database from the Ground Up

About.com's tutorial guides you through the process of creating a database for a small business. This tutorial refers to an earlier version of Access, but a majority of the instructions are still relevant for Access 2013.

A screenshot of the About.com website. The page title is "Building an Access Database From the Ground Up" by Mike Chapple. The page features a navigation menu with "Databases", "Database Basics", "Learning SQL", and "Using Databases". Below the title, there are links for "See More About: database design", "access tutorials", "database tutorials", and "microsoft access". The main text introduces the tutorial series, mentioning "Patrick's Widgets" and "Microsoft Access database from the ground up". It states that the complete listing of tutorials is available and that the series is available as a free weekly e-mail course. The Office logo is visible on the right side of the page.

LESSON 17: CREATING CALCULATED FIELDS AND TOTALS ROWS

Introduction

Calculated fields and **totals rows** let you perform calculations with the data in your tables. Calculated fields perform calculations using data within one record, while totals rows perform a calculation on an entire field of data.

Calculated fields

When you create a **calculated field**, you are adding a new field in which every row contains a calculation involving other numerical fields in that row. To do this, you must enter a **mathematical expression**, which is made up of **field names in your table** and **mathematical symbols**. You don't need to know too much about math or expression building to create a useful calculated field. In fact, you can write robust expressions using only grade-school math. For instance, you could:

- Use **+** to find the **sum** of the contents of two fields or to add a constant value (such as **+ 2** or **+ 5**) to a field
- Use ***** to **multiply** the contents of two fields or to multiply fields by a constant value
- Use **-** to **subtract** one field from other or to subtract a constant value from a field

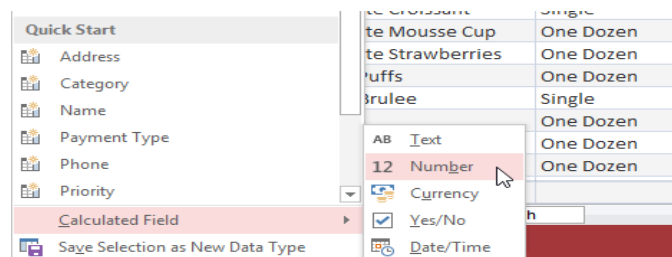
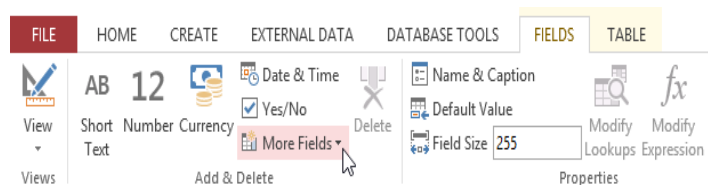
In our example, we will use a table containing the orders from one month. The table contains items listed by **sales unit**—single, half-dozen, dozen. One column lets us know the **number sold** of each sales unit. Another lets us know the actual **numerical value** of each of these units. For instance, in the top row you can see that **two dozen** fudge brownies have been sold and that one dozen equals **12** brownies.

Product Types	Product Name	Sales Unit	Value of Sales Unit	# of Sales Unit Sold
Cookies	Chocolate Banana Walnut	One Dozen	12	2

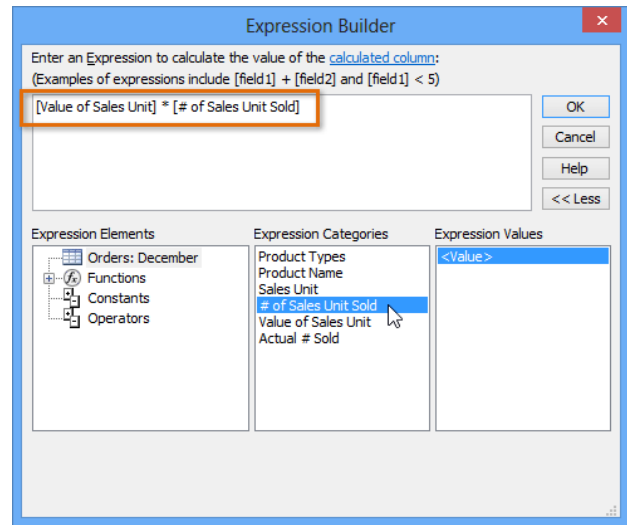
To find the **total number** of brownies that have been sold, we'll have to multiply the number of units sold by the numerical value of that unit—here, 2×12 , which equals 24. This was a simple problem, but performing that calculation for each row of the table would be tedious and time consuming. Instead, we can create a calculated field that shows the product of these two fields multiplied together on every row.

To create a calculated field:

1. Select the **Fields** tab, locate the **Add & Delete** group, and click the **More Fields** drop-down command.
2. Hover your mouse over **Calculated Field** and select the desired data type. We want our calculation to be a number, so we'll select **Number**.

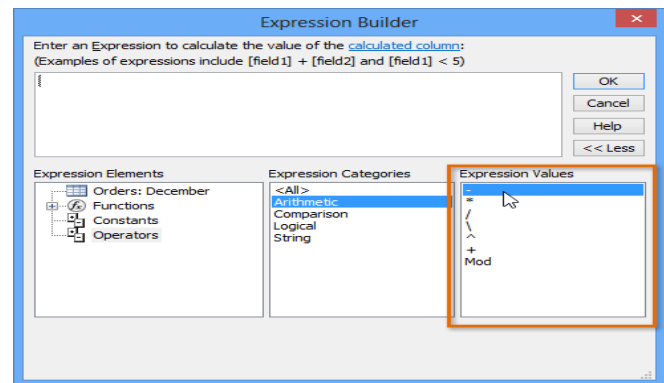


1. Build your expression. To select fields to include in your expression, double-click the field in the **Expression Categories** box. Remember to include mathematical operators like the + or - signs. Because we want to **multiply** our two fields, we'll put the multiplication symbol (*) between them.
2. Click **OK**. The calculated field will be added to your table. If you want, you can now sort or filter it.



Sales Unit	Value of Sales Unit	# of Sales Unit Sold	Actual # Sold
One Dozen	12	2	24
Single	1	4	4
Single	1	5	5
Single	1	3	3
Single	1	8	8
Single	1	5	5
One Dozen	12	1	12
Single	1	12	12
One Dozen	12	1	12
Single	1	1	1
Single	1	3	3
One Dozen	12	3	36
One Dozen	12	6	72
One Dozen	12	1	12
Single	1	8	8
Single	1	18	18
One Dozen	12	3	36

For more examples of mathematical expressions that can be used to create calculated fields, review the **Arithmetic Expressions** in the **Expression Builder** dialog box.



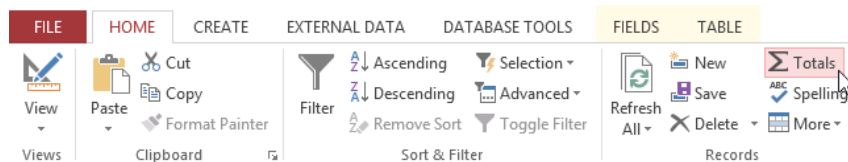
Totals rows

The **totals row** adds up an entire column of numbers, just like in a ledger or on a receipt. The resulting sum appears in a special row at the bottom of your table.

For our example, we'll add a totals row to our calculated field. This will show us the total number of items sold.

To create a totals row:

1. From the **Home** tab, locate the **Records** group, then click the **Totals** command.



2. Scroll down to the **last row** of your table.
3. Locate the desired field for the totals row, then select the **second empty cell** below the last record for that field. When a **drop-down arrow** appears, click it.

Pies	Pecan	Single	10	10
Pies	Pumpkin	Single	9	9
Pies	Sweet Potato	Single	3	3
*				
	Total			

Click To add total

- Select the function you want to perform on the field data. In our example, we'll choose **Sum** to add all of the values in the calculated field.

Pies	Pecan	Single	10	10
Pies	Pumpkin	Single	9	9
Pies	Sweet Potato	Single	3	3
*	Total			

None

Sum

Average

Count

Maximum

Minimum

Standard Deviation

Variance

- The totals row will appear.

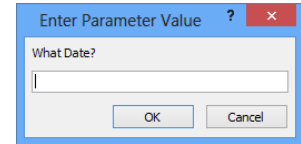
Pies	Peanut Butter Chocolate	Single	3	3
Pies	Pecan	Single	10	10
Pies	Pumpkin	Single	9	9
Pies	Sweet Potato	Single	3	3
*	Total			1289

LESSON 18: CREATING A PARAMETER QUERY

Introduction

A **parameter query** is one of the simplest and most useful queries you can create. Because parameter queries are so simple, they can be easily updated to reflect a new **search term**. When you open a parameter query, Access will prompt you for a search term and show you query results that reflect your search.

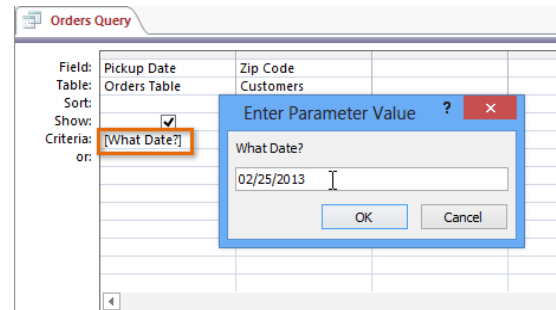
When you're running parameter queries, search terms act as **variable criteria**, which are query criteria that **change** each time you run the query. For instance, let's say we own a bakery and want to create a query that will quickly look up orders that were placed on a certain date. We could create a parameter query with variable criteria in the **Date** field. This way, each time we run the query a dialog box will appear to prompt us to enter the date we'd like our query to search for.



We'll enter the date we want, then Access will run the query using the date we entered as a search term.

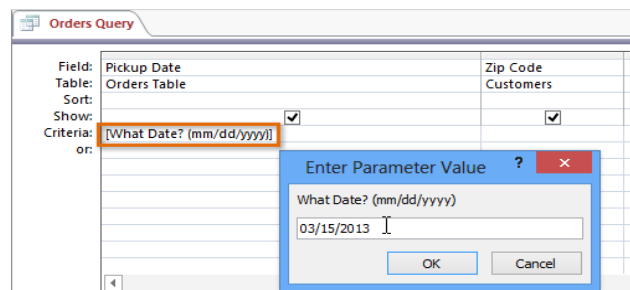
To create and run a parameter query:

1. Create a query as you normally would, modifying the table joins if necessary, selecting the fields to include in your query, and adding any non variable criteria to the appropriate fields in the **Criteria:** row.
2. Locate the field or fields where you want the variable criteria to appear, then select the **Criteria:** row.
3. Type the phrase you want to appear in the prompt that will pop up each time you run your query. Make sure to enclose the phrase in brackets []. For example, in our parameter query that searches for orders placed on a certain date, we might type our criteria like this: **[What Date?]**.
4. On the **Design** tab, click the **Run** command to **run** your query. A dialog box will appear with the specified prompt. Enter your search term and click **OK** to view your query results.



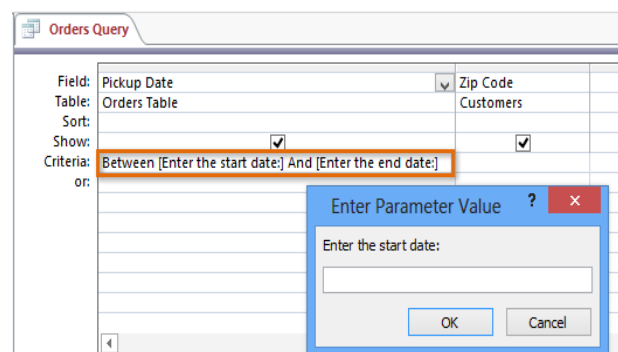
Tips for writing parameter queries

- Ideally, the prompt you create for your query should make it clear what **type** of information the search term should be, as well as the desired **format**. For example, to guarantee users enter a search for a date in the format used in our database, we could write the following in the **Criteria:** row of the **Pickup Date** field like this: **[What Date? (mm/dd/yyyy)]**.



- The simplest parameter query will give you an exact-match criteria, meaning the query will search for the **exact text** you enter in the prompt. However, you can turn any type of criteria into a variable criteria. Simply type your prompt text in brackets in the part of the criteria where you would normally put a search term.

For example, in a normal query we could find orders that were placed **between** two dates by using the criteria **Between x AND y** and replacing **x** and **y** with the first and second dates, respectively. To turn this into a parameter criteria, we would simply replace the **x** and **y** with the text we want to appear in the prompt. Our variable criteria might look like this: **Between [Enter the start date:] And [Enter the end date:]**. These two prompts will appear when you run the query.



LESSON 19: CREATING A FIND DUPLICATES QUERY

Introduction

A **find duplicates query** allows you to search for and identify **duplicate records** within a table or tables. A duplicate record is a record that refers to the **same thing** or **person** as another record.

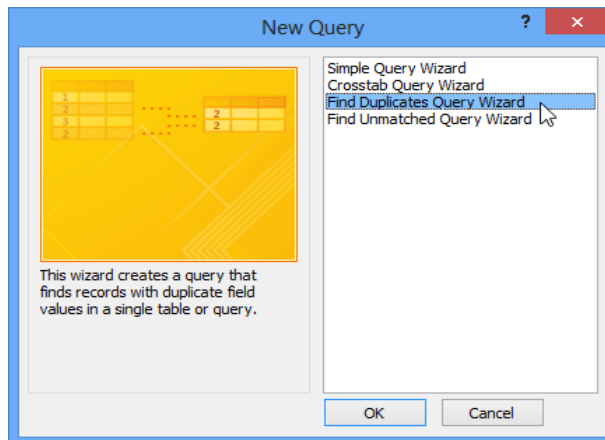
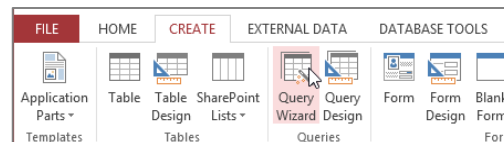
Not all records containing similar information are duplicates. For instance, records of two orders that were placed on different dates but that contained identical items would **not** be duplicate records. Likewise, not all duplicate records contain completely identical information. For example, two customer records could refer to the same person but include different addresses. The record with the out-of-date address would be the duplicate record.

Why is getting rid of duplicate records so important? Consider the example above. If we had multiple records for one customer, it would be difficult to view an order history for him because the information would be spread across several unlinked records. We might even deliver his order to the wrong address if the person entering the order information selects an outdated record. It's easy to see how having duplicate records can undermine the integrity and usefulness of your database.

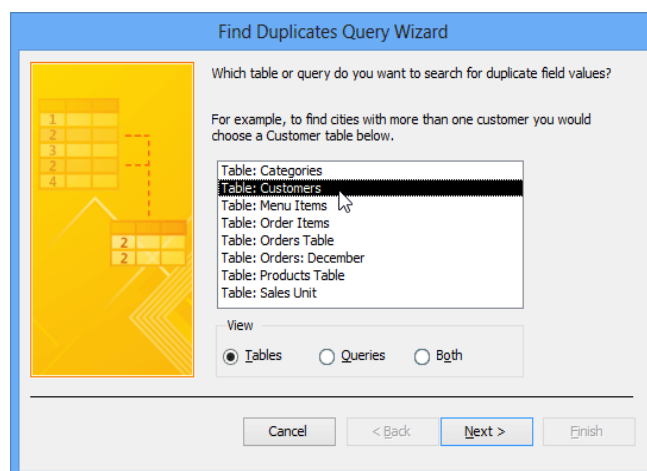
Fortunately, Access makes it easy to search for and locate potential duplicate records. Note that Access won't delete the records for you or help you figure out which one is current—you'll have to do those things for yourself. If you're familiar with the data in your database, though, getting rid of duplicate records will be a manageable task.

To create a find duplicates query:

1. Select the **Create** tab on the Ribbon, locate the **Queries** group, and click the **Query Wizard** command.
2. The **New Query** dialog box will appear. Select **Find Duplicates Query Wizard** from the list of queries, then click **OK**.



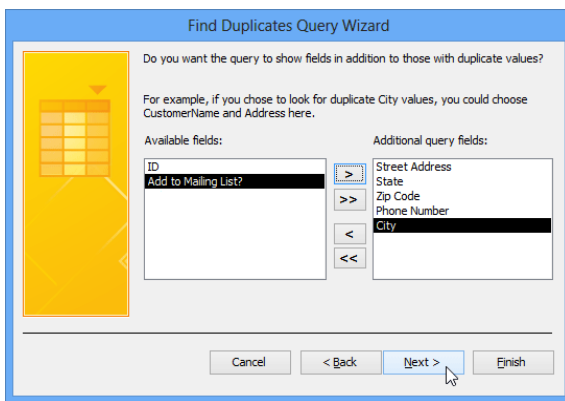
1. Select the table you want to search for duplicate records, then click **Next**. We're searching for duplicate customer records, so we'll select the **Customers** table.
2. Choose the fields you want to search for duplicate information by selecting them and then clicking the **right arrow button**. Only select fields that should not be identical in non duplicate records. For instance, because we're searching for duplicate customers we'll only select the **First Name** and **Last Name** fields because it's unlikely that multiple people with the exact same first and last names would place orders at our bakery.



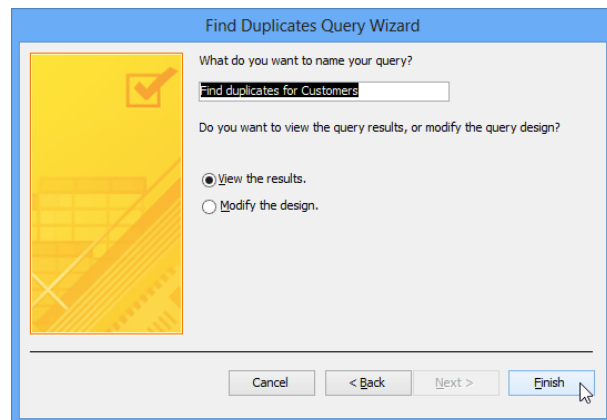
- When you've added the desired fields, click **Next**.



- Select additional fields to view in the query results. Choose fields that will help you distinguish between the duplicate records, and choose which one you want to keep. In our example, we'll add all of the fields relating to customer **addresses**, plus the **Phone Number** field because records with identical customer names might contain non identical information in this field. When you're satisfied, click **Next**.



- Access will suggest a name for your query, but you can type a different name if you want. When you're satisfied with the query name, click **Finish** to run your query.



- If Access found any duplicate records in your query, they will be displayed in the **query results**. Review the records and **delete** any outdated or incorrect records as needed.

First Name	Last Name	Street Address	State	Zip Code	Phone Number	City
David	Barrett	434 Hill St.	NC	27609	919-555-0662	Raleigh
David	Barrett	430 Hill St.	NC	27609	919-555-0662	Raleigh
Magda	Sremski	544 Wayne St.	NC	27612	919-555-4001	Raleigh
Magda	Sremski	98 Tyler St.	NC	27612	919-555-1024	Raleigh
*						

Tips for resolving duplicate records

- **Save** your duplicate records queries, and run them often.
- **Investigate** potential duplicate records by looking at linked data in other tables. You can do this by searching for these records' **record ID numbers** in related tables. Is one record linked to mostly old orders while another contains recent ones? The latter is likely to be the current one.
- Once you decide which record to delete, make sure you won't be losing any information you might need. In our example, before we deleted our duplicate record we found all of the orders linked to that record's **ID number** and replaced them with the ID number of the record we decided to keep.

Simplified

E- BOOK

MS Access 2013



(An I.T Skill Advancement Training Programme, Initiated by SITED®-India)

An ISO 9001:2015 Certified Organization

Legal: *No part of this e- book publication may be reproduced, stored in retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, and recording otherwise, without the prior permission of the abovementioned Organization. Every possible effort has been made in bringing out the text in this e-book correctly and completely to fulfill the aspirations of students. The Organization does not take any warranty with respect to the accuracy of the e- book and hence cannot be held liable in any way for any loss or damages whatsoever. This book shall be used for non commercial I.T Skill Advancement awareness programme, not for commercial purposes publicly.*

This is an independent work, compiled solely for information and guidance for students studying under Organization's I.T & Skill Advancement Training literacy awareness Programmes. The information's have been compiled from various sources. The Organization does not assume any responsibility for performance of any software, or any part thereof, described in the book. Product Names mentioned are used for identification/IT literacy awareness purposes only and may be trademarks of their respective companies. All trademarks referred to in the book are acknowledged as properties of their respective owners. The Centre Head & students should, in their own interest, confirm the availability of abovementioned books titles features or softwares from their respective authorized Companies or Owners or dealers or authors.

Centre Stamp

Chapter at a Glance

Chapter-1	Introduction to Databases	----	Page No. 1 to 2
Chapter-2	Introduction to Objects	----	Page No. 3 to 5
Chapter-3	Getting Started With Access	----	Page No. 6 to 9
Chapter-4	Managing Databases and Objects	----	Page No. 10 to 12
Chapter-5	Working With Tables	----	Page No. 13 to 19
Chapter-6	Working With Forms	----	Page No. 20 to 22
Chapter-7	Sorting And Filtering Records	----	Page No. 21 to 26
Chapter-8	Designing A Simple Query	----	Page No. 27 to 28
Chapter-9	Designing A Multi-Table Query	----	Page No. 29 to 33
Chapter-10	More Query Design Options	----	Page No. 34 to 37
Chapter-11	Creating Reports	----	Page No. 38 to 40
Chapter-12	Advanced Report Options	----	Page No. 41 to 46
Chapter-13	Modifying Tables	----	Page No. 47 to 50
Chapter-14	Creating Forms	----	Page No. 51 to 56
Chapter-15	Formatting Forms	----	Page No. 57 to 60
Chapter-16	Designing Your Own Database	----	Page No. 61 to 62
Chapter-17	Creating Calculated Fields And Totals Rows	----	Page No. 63 to 65
Chapter-18	Creating A Parameter Query	----	Page No. 66 to 66
Chapter-19	Creating A Find Duplicates Query	----	Page No. 67 to 68