

# WORKING WITH QUERIES, FORMS AND REPORTS

## CHAPTER-03

What are queries?

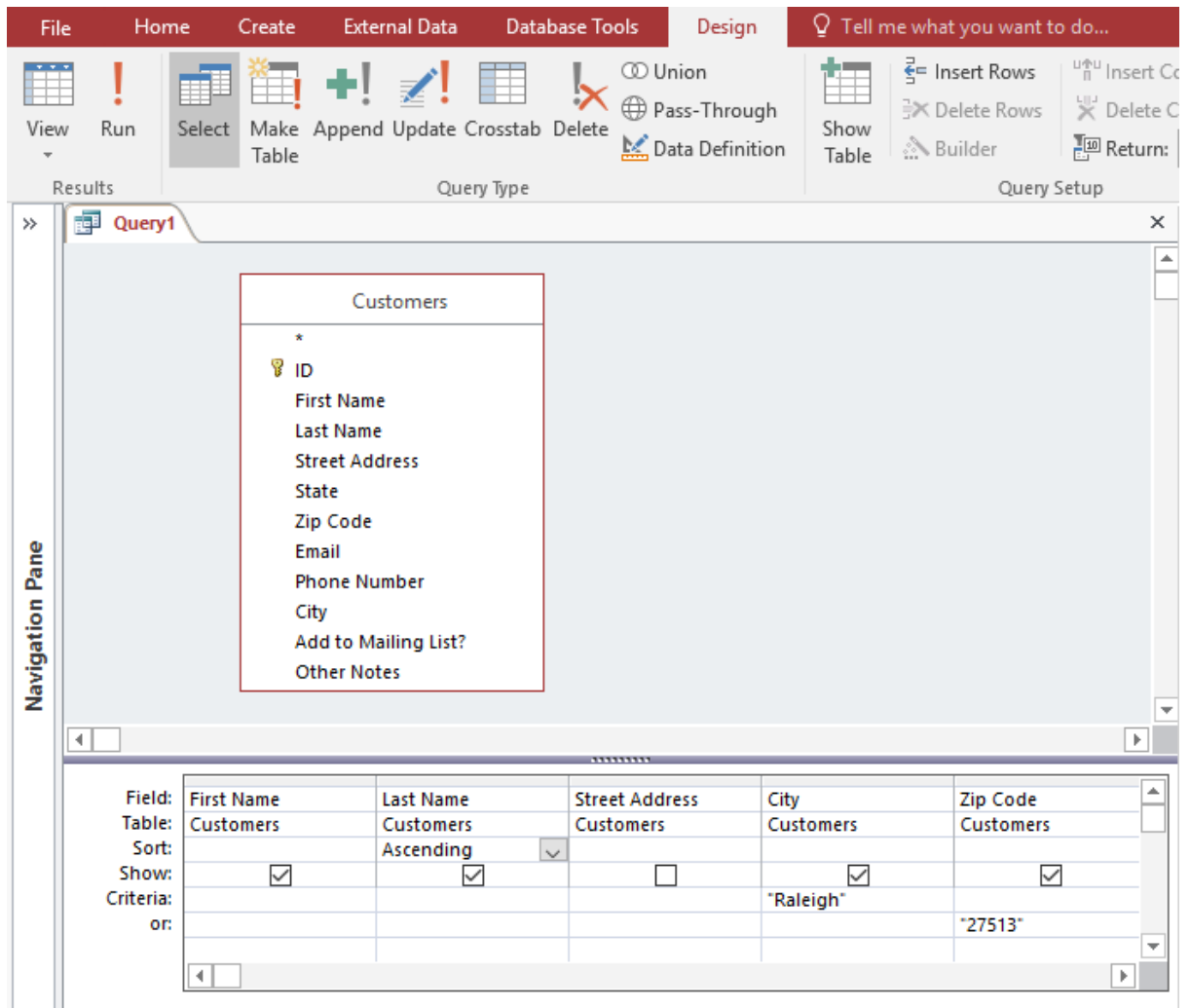
Queries are a way of **searching** for and **compiling** data from one or more tables. Running a query is like asking a **detailed question** of your database. When you build a query in Access, you are **defining specific search conditions** to find exactly the data you want.

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.

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 **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.

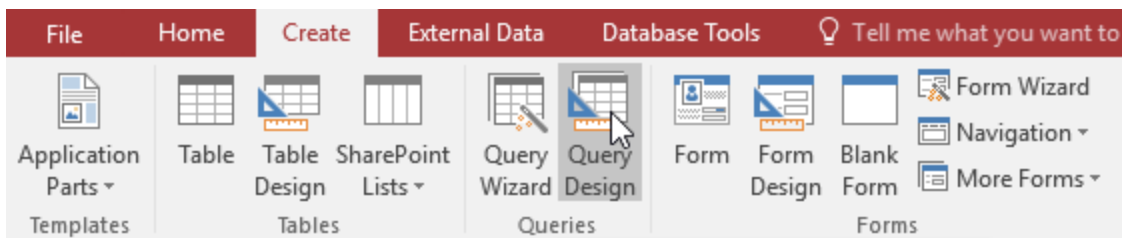
We want to find our customers who live in the city of **Raleigh**, so we'll search for "**Raleigh**" in the City field. Some customers who live in the suburbs live fairly

close by, and we'd like to invite them as well. We'll add their zip code, **27513**, as another criteria.

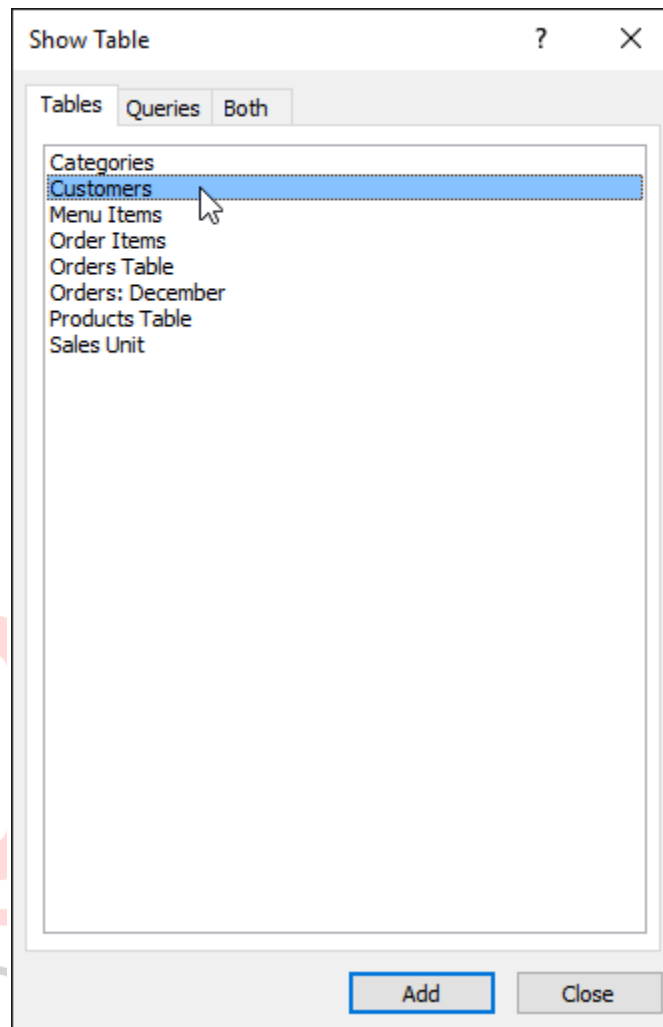
If you think this sounds a little like applying a filter, you're right. A one-table query is actually just an **advanced filter** applied to a table.

To create a simple one-table query:

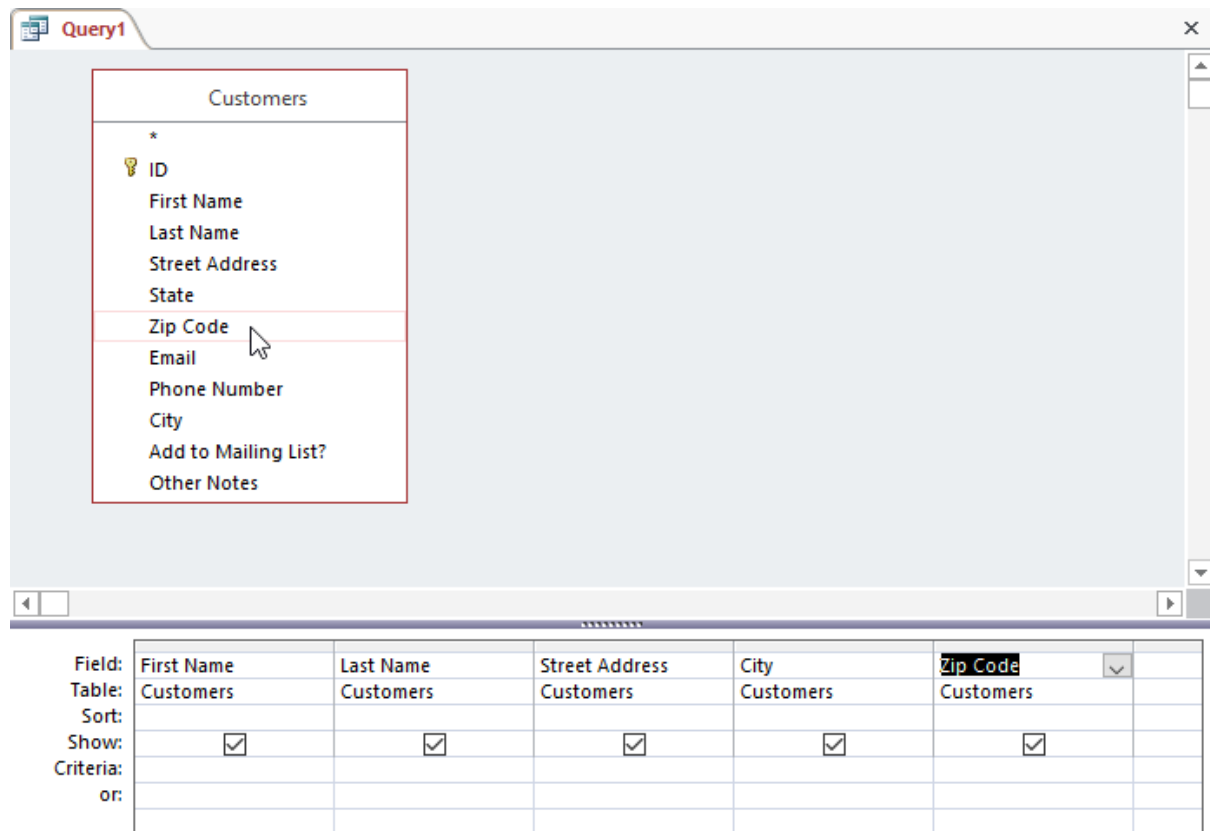
1. Select the **Create** tab on the Ribbon and locate the **Queries** group.
2. Click the **Query Design** command.



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 on our customers, so we'll select the **Customers** table.



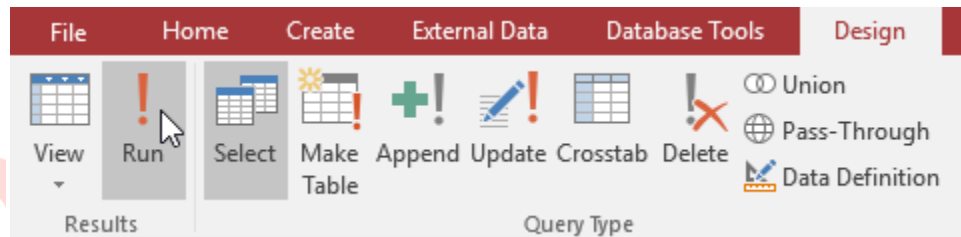
4. Click **Add**, then click **Close**.
5. The selected table will appear as a small window in the **Object Relationship pane**. In the table window, double-click the **field names** you want to include in your query. They will be added to the **design grid** in the bottom part of the screen. In our 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.



6. 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. Because we want to find customers who either live in Raleigh **or** in the 27513 zip code, we'll type "Raleigh" in the **City** field and "27513" into the **or:** row of the **Zip Code** field. The **quotation marks** will search these fields for an **exact match**.

Field:	City	Zip Code	
Table:	Customers	Customers	
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input 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	Jones	450 Denver Rd.	Cary	27513
Beatrix	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

Save As

Query Name:

Nearby Customers

OK Cancel

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**.

Challenge!

1. Open our [practice database](#).
2. **Create** a new query.
3. Select the **Customers** table to include in your query.
4. **Add** the following **fields** from the **Customers** table to your query:  
**First Name**  
**Last Name**  
**City**  
**Zip Code**
5. Set the following **criteria**:  
In the **City** field, type "**Durham**" to return only records with Durham in the City field.  
In the **Zip Code** field, type "**27514**" in the **or:** row to return records that are either in Durham or zip code 27514.
6. **Run** the query. If you entered the query correctly, your results will include customers who live in Durham **OR** in zip code 27514.
7. **Save** the query with the name **Customers who live in Durham**.

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**.

Throughout this tutorial, we will be using a sample database. If you would like to follow along, you'll need to download our [Access sample database](#). You will need to have Access installed on your computer in order to open the example.

Watch the video below to learn how to create a multi-table query (Part 1).

Watch the video below to learn more about joins and query criteria (Part 2).

### 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, follow these four steps:

**Pinpoint** exactly what you want to know. If you could ask our 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.

**Identify** every type of information you want included in your query results. Which fields contain this information?

**Locate** the fields you want to include in your query. Which tables are they contained in?



**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?

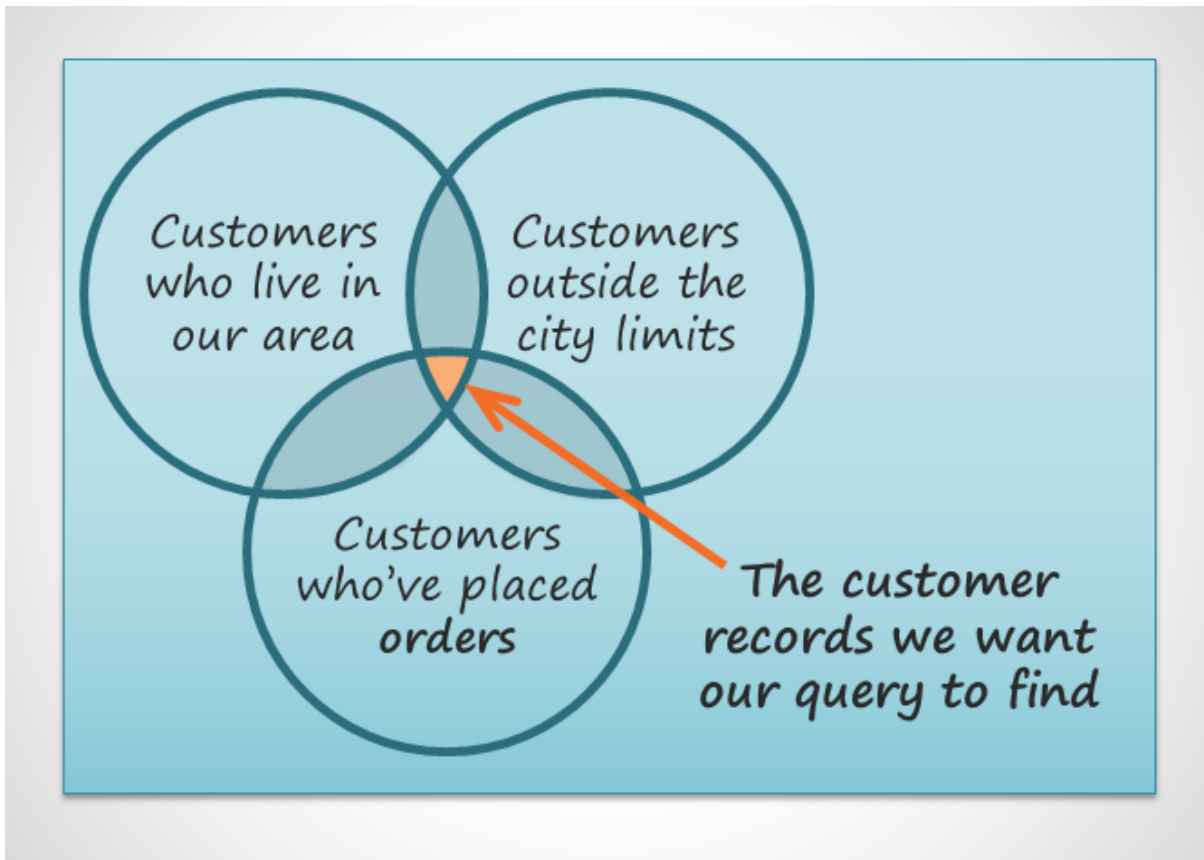
This process might seem abstract at first, but as we go through the process of planning our own multi-table query you should start to understand how planning your queries can make building them much easier.

### 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.

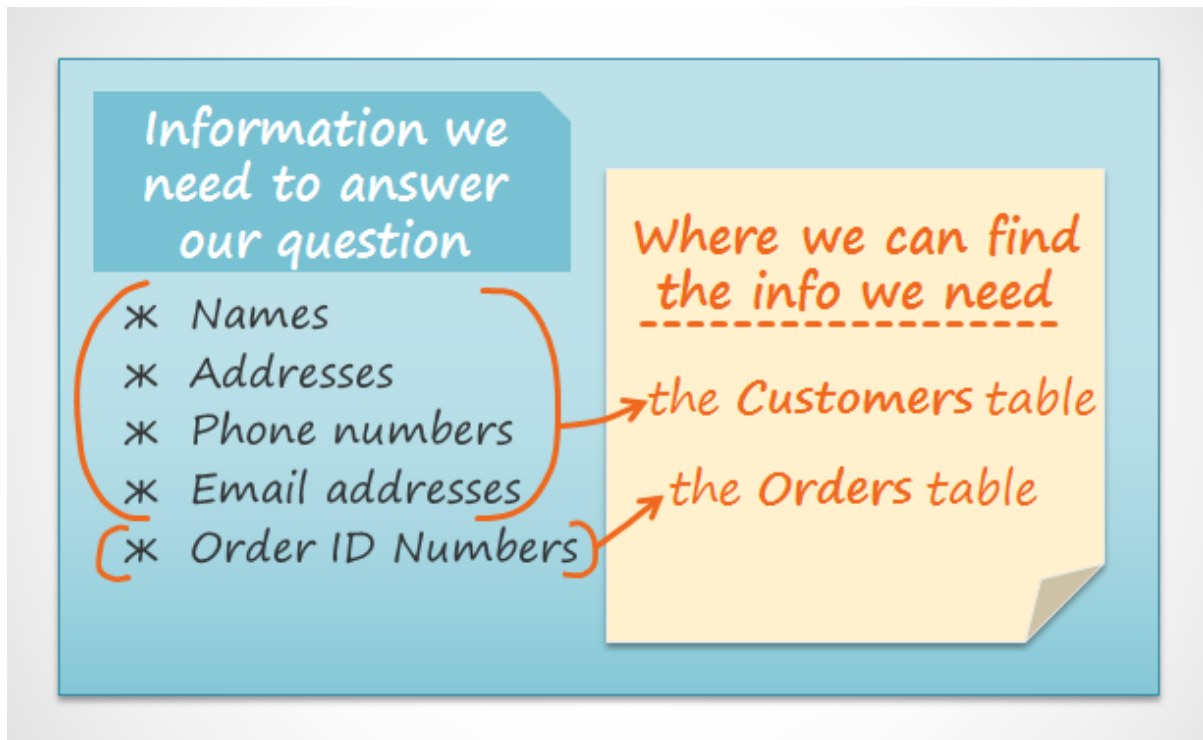
### Information we need to answer our question

- ✳ Names
- ✳ Addresses
- ✳ Phone numbers
- ✳ Email addresses
- ✳ Order ID Numbers

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 919 area code, 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.

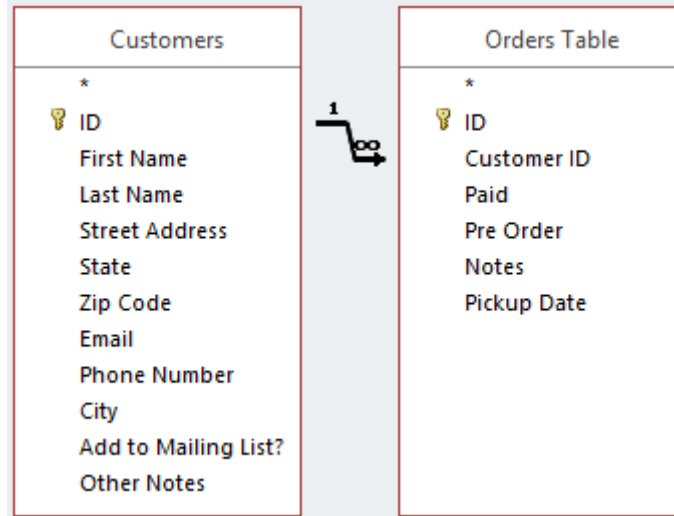
### Criteria the query should use to find customer records:

- \* No one living in our town, Raleigh
  - \* In the City field, type **Not in ("Raleigh")**
- \* Only customers with phone numbers that start with "919"  
(So we only get customers who live nearby)
  - \* In the Phone Number field, type **Like ("919\*")**

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, review our printable [Query Criteria Quick Reference Guide](#), which includes several of the most common criteria used in Access queries.

#### Joining tables in queries

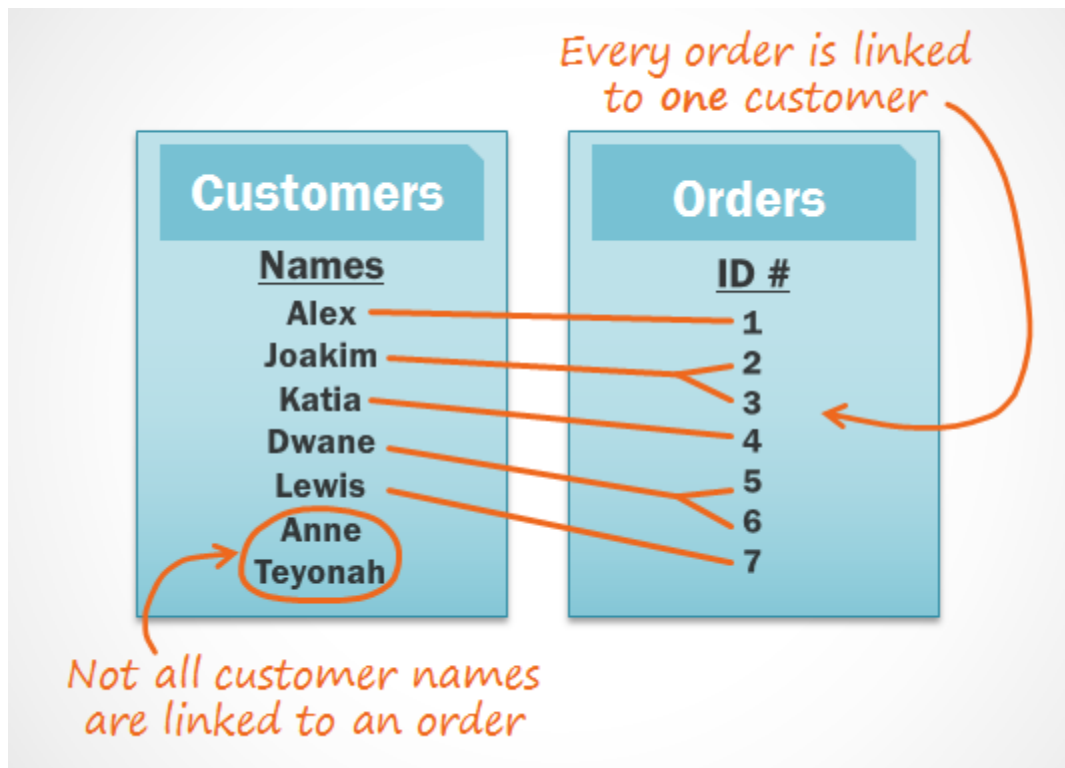
The final thing you need to consider when designing a query is the way you link, or **join**, the tables you're working with. When you add two tables to an Access query, this is what you'll see in the **Object Relationship pane**:



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.

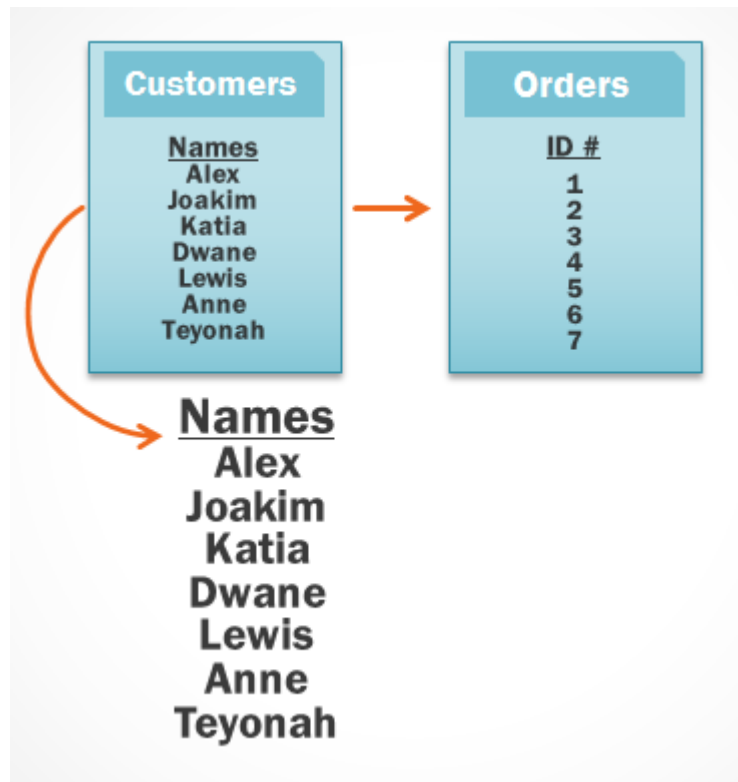
Your tables won't always be joined this way. Sometimes Access will join them **right to left**. In either case, you might need to **change the direction** of the join to make sure your query includes the correct information. The join direction can affect **which information** your query **retrieves**.

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.

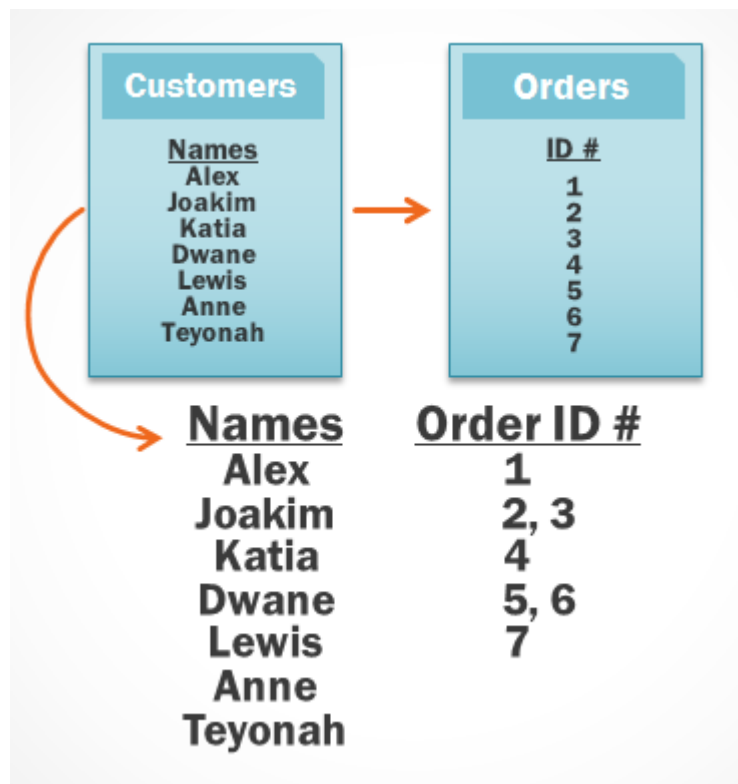


It then retrieves every record from the **right** table that has a relationship with a record Access has already taken from the left table.

EDUCATIONAL GROUP

*Changing your Tomorrow*





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:

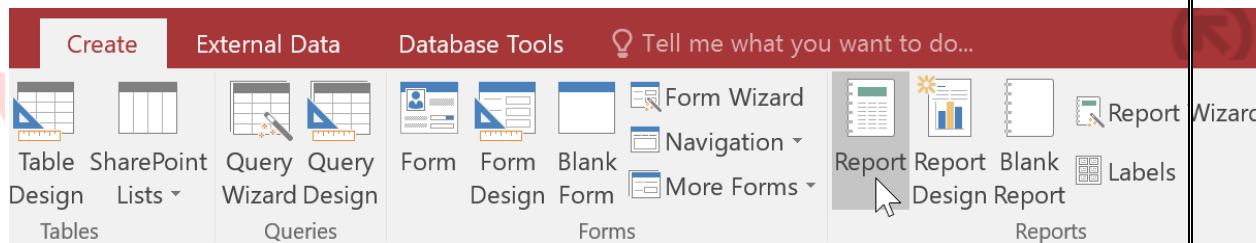
### ***To create a report:***

**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	SumOfQuantity
Cookies	Fudge Brownie	One Dozen	
Cookies	Fudge Chocolate	Single	
Cookies	Ginger Shortbread	One Dozen	
Cookies	Chocolate Chip	Single	
Cookies	Butterscotch	Single	
Cookies	Fudge Brownie	Single	
Cookies	Cranberry Walnut	One Dozen	
Cookies	White Chocolate Macademia Nut	Half-Dozen	
Cookies	Snickerdoodle	Single	

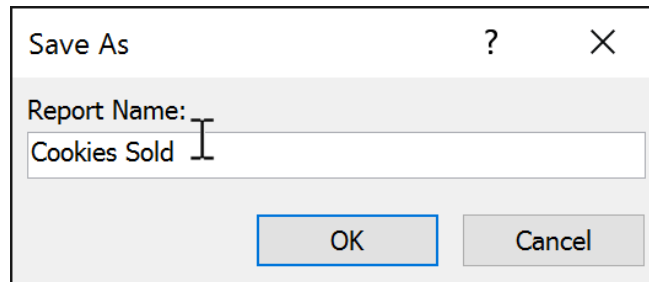
2. Select the **Create** tab on the Ribbon. Locate the **Reports** group, then click the **Report** command.



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]	[Pro
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**.



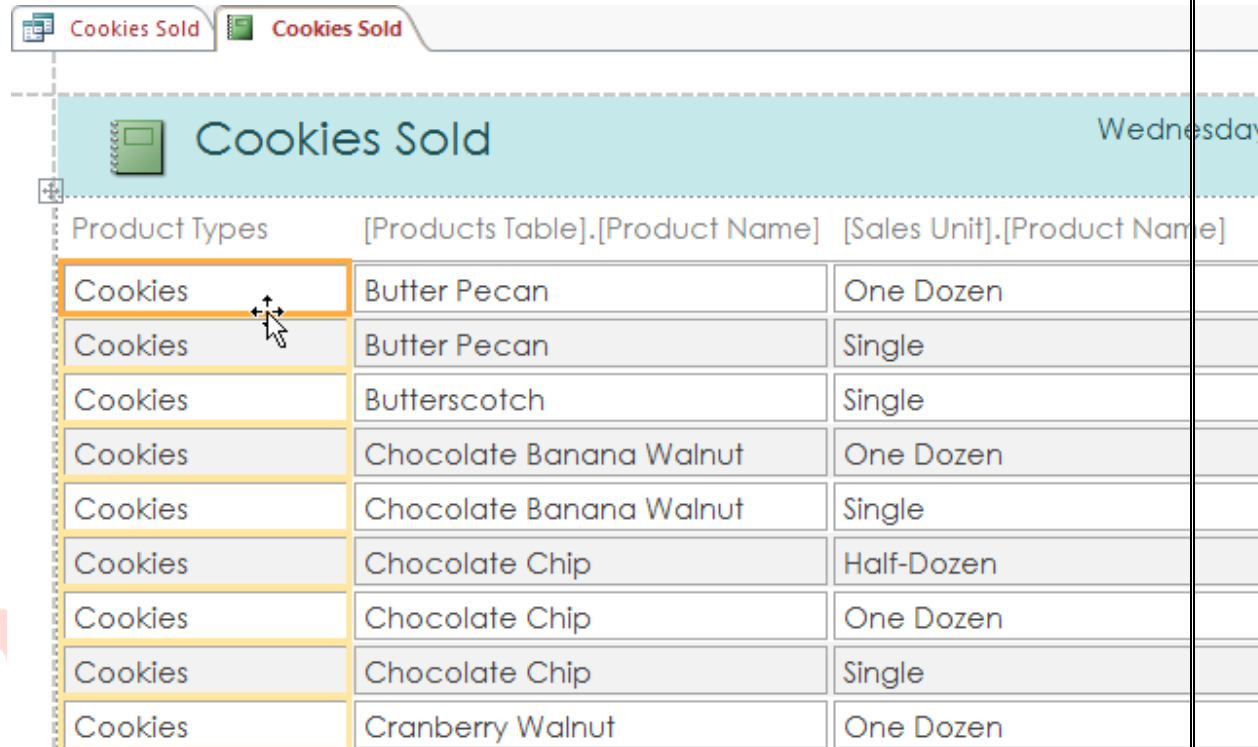
Just like tables and queries, reports can be **sorted** and **filtered**. Simply **right-click** the field you want to sort or filter, then select the desired option from the menu.

### ***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.

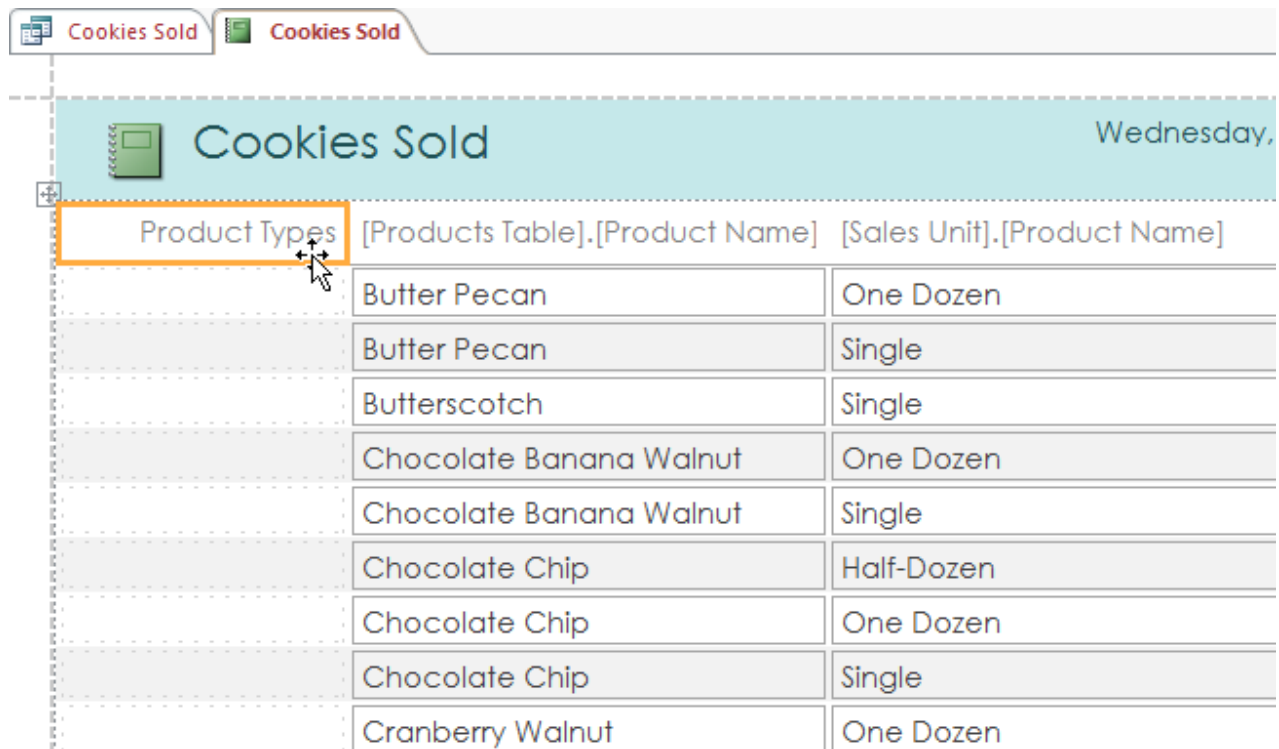


Product Types	[Products Table].[Product Name]	[Sales Unit].[Product Name]
Cookies	Butter Pecan	One Dozen
Cookies	Butter Pecan	Single
Cookies	Butterscotch	Single
Cookies	Chocolate Banana Walnut	One Dozen
Cookies	Chocolate Banana Walnut	Single
Cookies	Chocolate Chip	Half-Dozen
Cookies	Chocolate Chip	One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

EDUCATIONAL GROUP  
Changing your Tomorrow



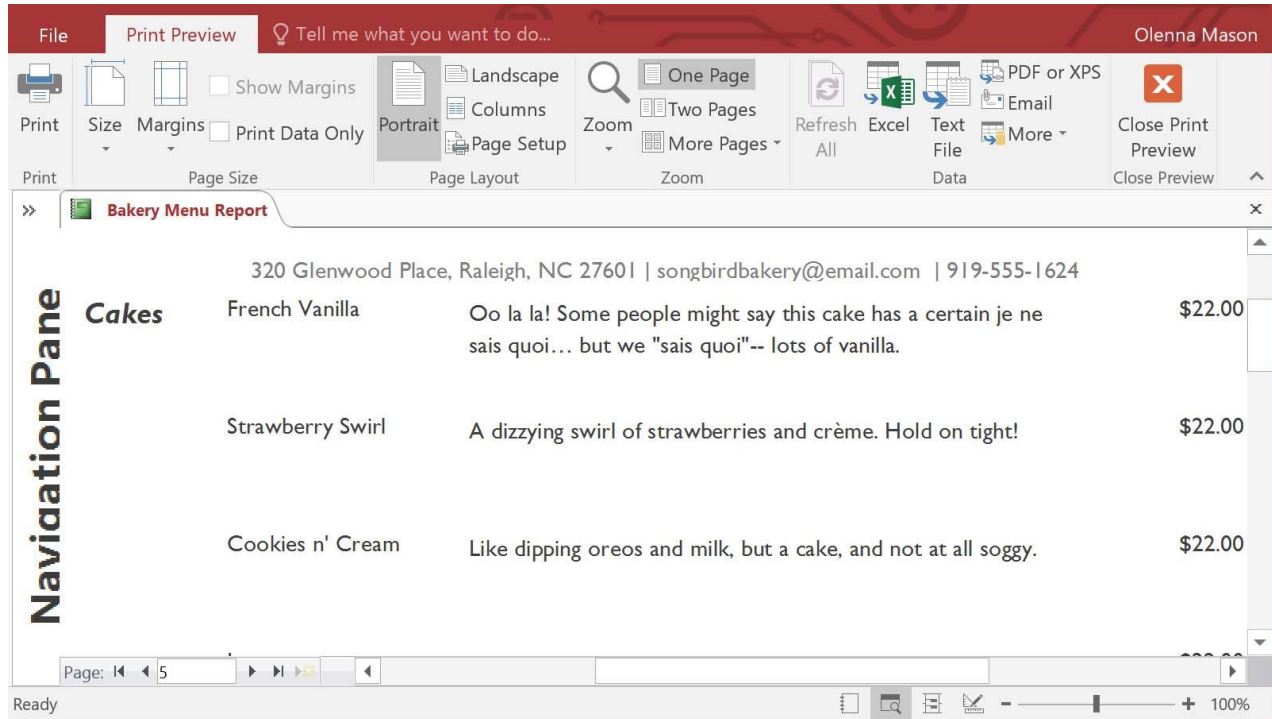
Product Types	[Products Table].[Product Name]	[Sales Unit].[Product Name]
	Butter Pecan	One Dozen
	Butter Pecan	Single
	Butterscotch	Single
	Chocolate Banana Walnut	One Dozen
	Chocolate Banana Walnut	Single
	Chocolate Chip	Half-Dozen
	Chocolate Chip	One Dozen
	Chocolate Chip	Single
	Cranberry Walnut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

### Printing and saving reports in Print Preview

While you can print reports using commands in **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.

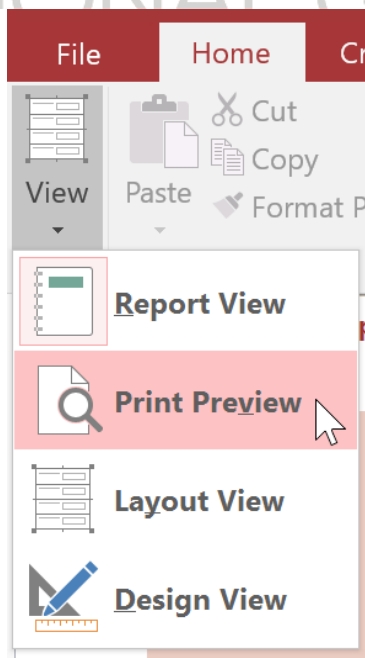
Watch the video below to learn more about printing reports.

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

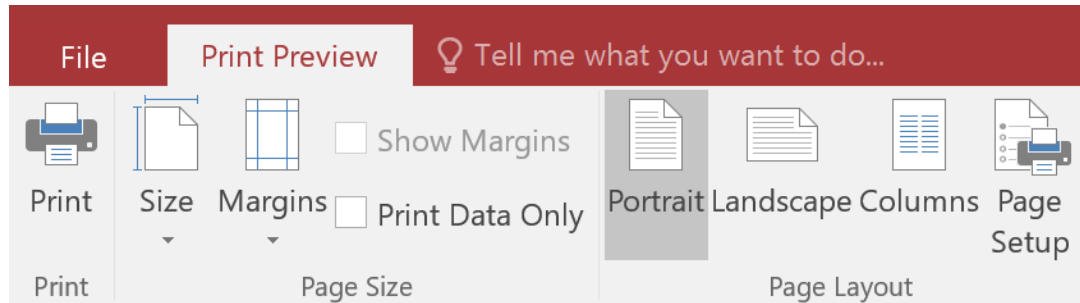


**To print a report:**

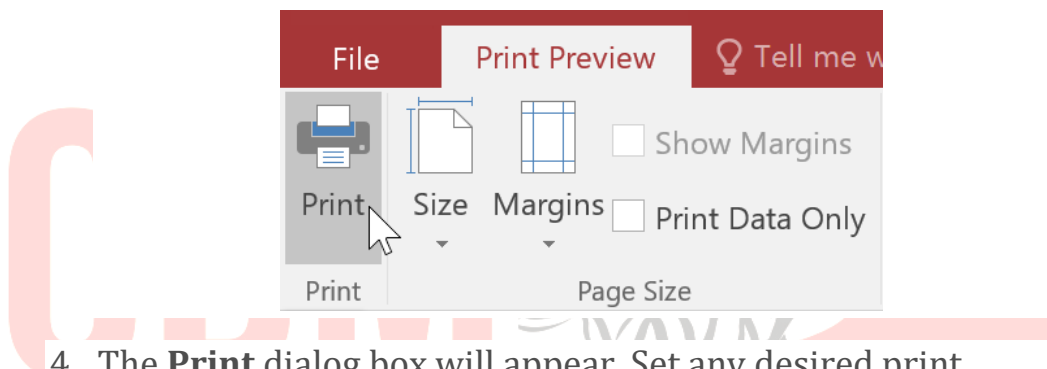
1. From the **Home** tab, click the **View** command, then select **Print Preview** from the drop-down list. Your report will be shown as it will appear on the printed page.



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**. The report will be printed.

### **Saving reports**

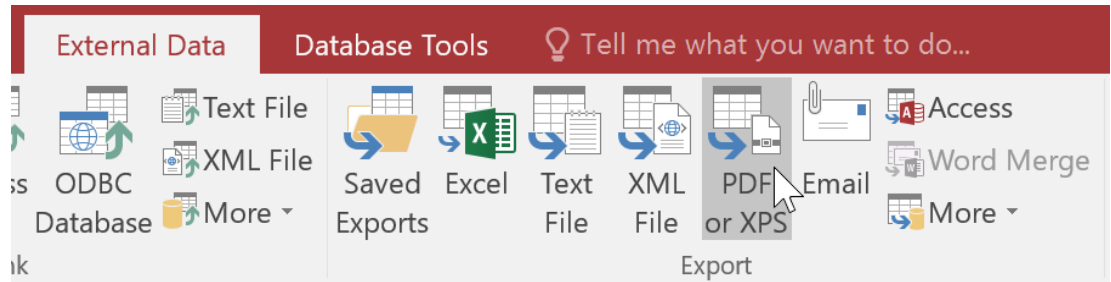
You can save reports in other formats so they'll be viewable outside of Access. This is called **exporting** a file, and it allows you to view and even modify reports in other formats and programs.

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

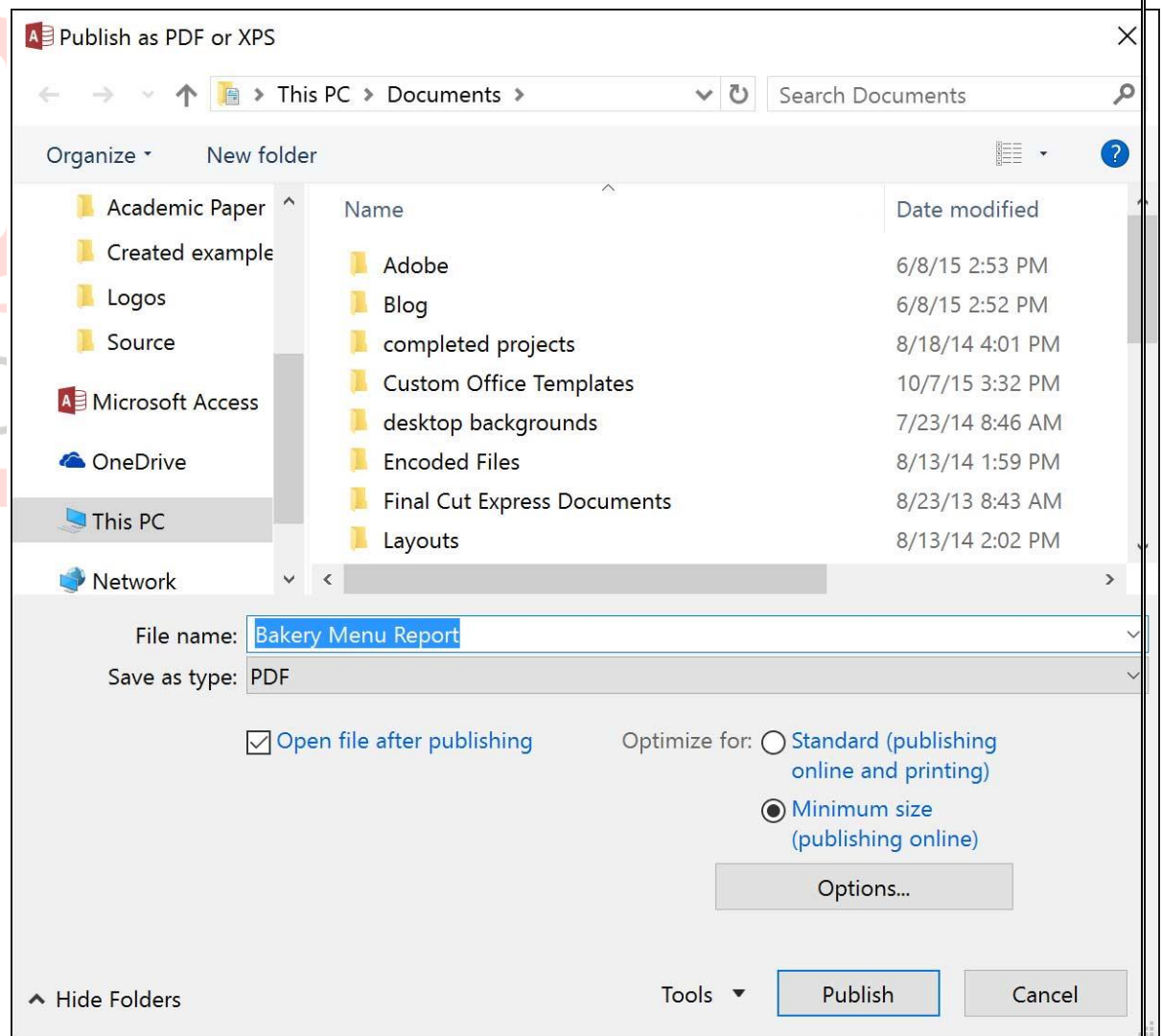
### **To export a report:**

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.

Export - Text File

Select the destination for the data you want to export

Specify the destination file name and format.

File name: \\psf\Home\Documents\Bakery Menu Report.txt Browse...

Specify export options.

**Export data with formatting and layout.**  
Select this option to preserve most formatting and layout information when exporting a table, query, form, or report.

**Open the destination file after the export operation is complete.**  
Select this option to view the results of the export operation. This option is available only when you export formatted data.

**Export only the selected records.**  
Select this option to export only the selected records. This option is only available when you export formatted data and have records selected.

OK Cancel

### Challenge!

1. Open our [practice database](#).
2. Open the **Customers Who Live Nearby** query and use it to **create** a report.
3. **Resize** the fields so all of the information is on the left side of the page break. Make sure the columns still have enough width to display all of the text.
4. **Move** the page number so it is to the left of the page break.

5. **Export** the report as a **PDF**.

