

Step by Step guide to building a database, assignment details are at the end.

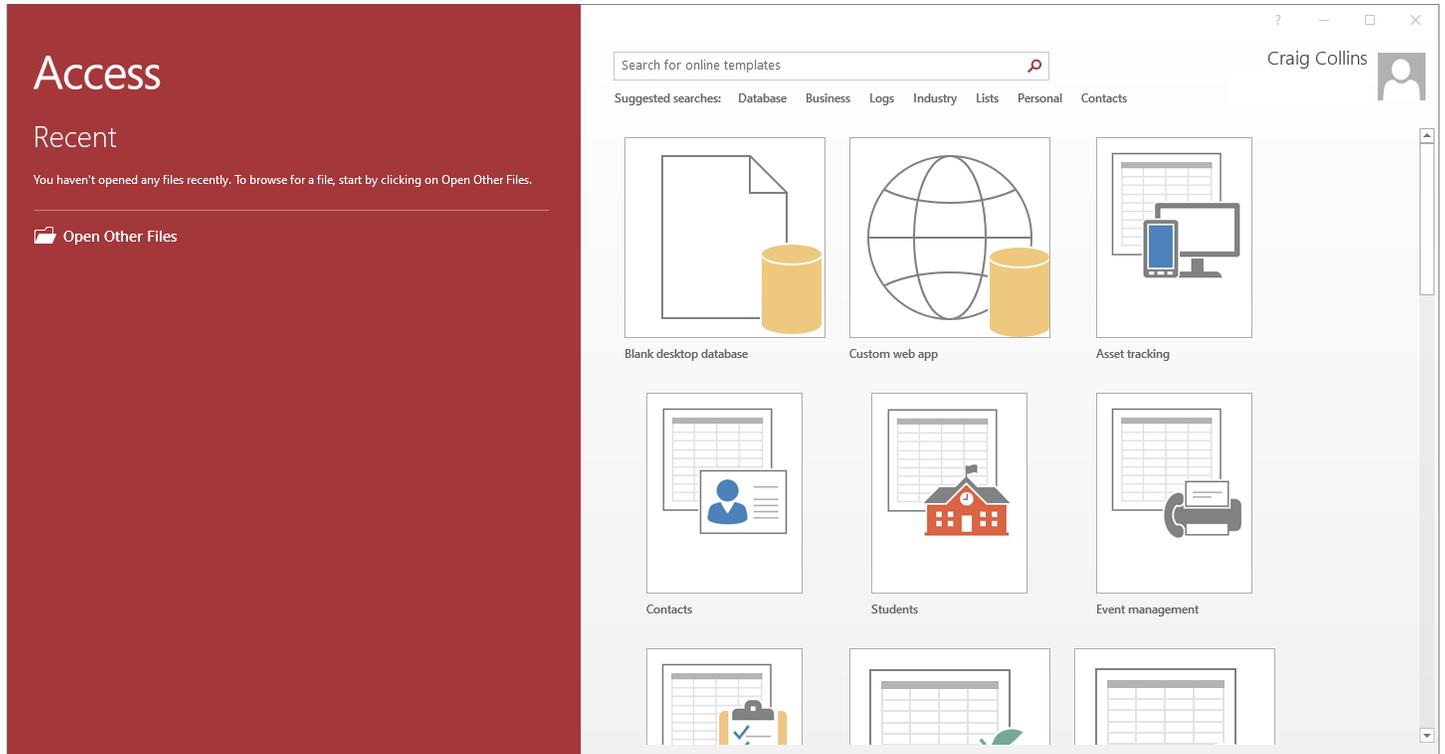
Access is different from other Microsoft Office tools, if you open a Word document, you see the document.

If you open an Access Database, you see the objects that can go in a database, the tables, forms, queries, and reports. You select which object you want to deal with, and often you have two different views... the normal view or the design view.

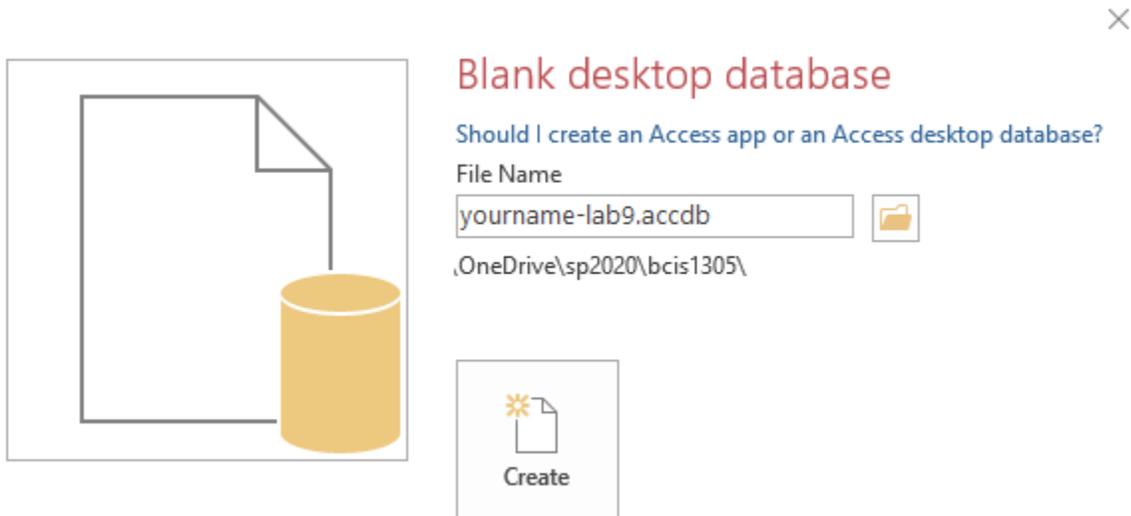
The following is sort of a step by step guide to building a database. Tables are below. Forms are on page 9.

Relationships are on page 13. Queries are on page 16. Modifying queries are on page 18. Forms are on page 19.

We'll start with a Blank desktop database. Note: your screens may differ.

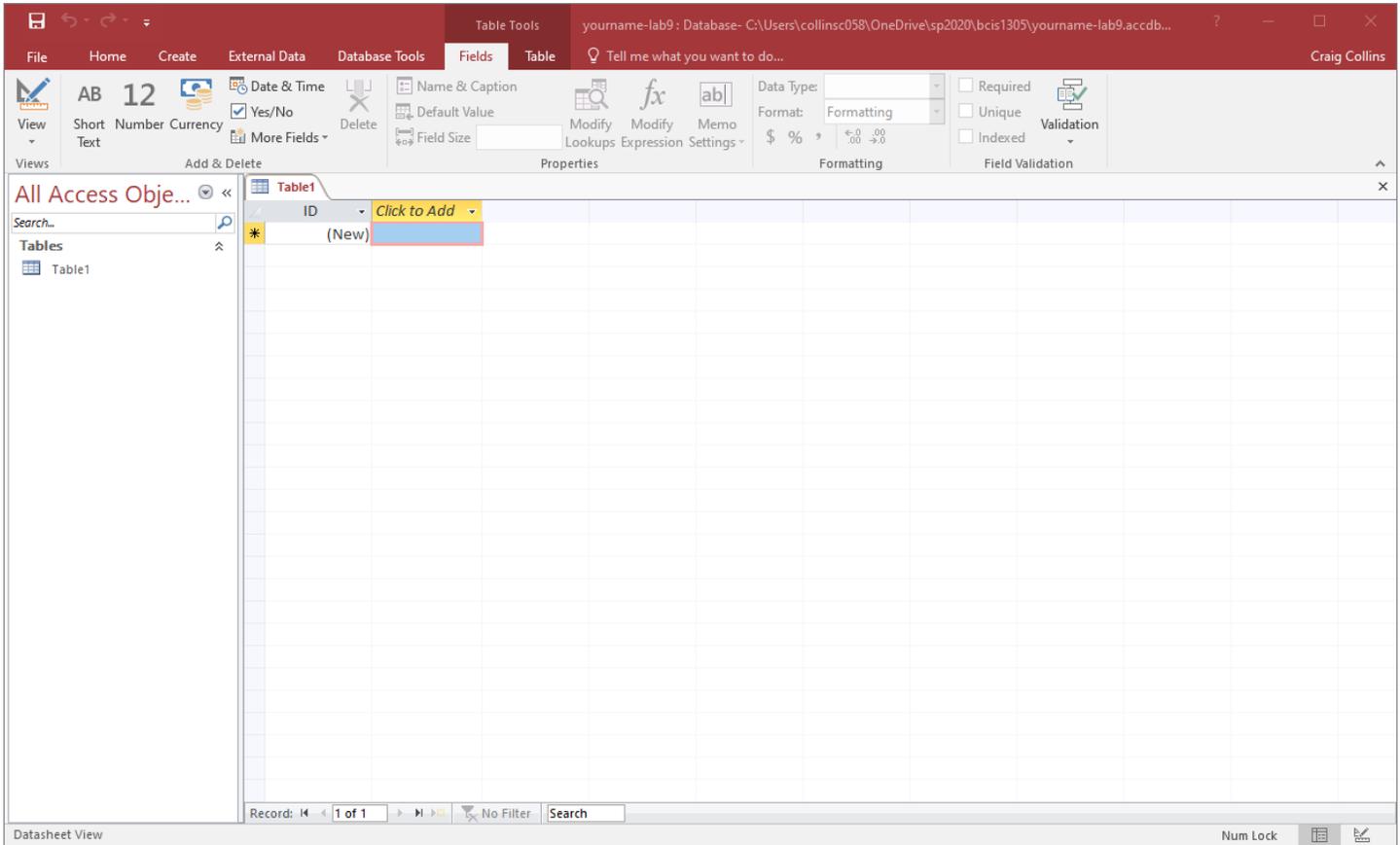


Unlike other Office products, you have to Save As FIRST. Browse to where you wish to save your file, then give it a meaningful name.

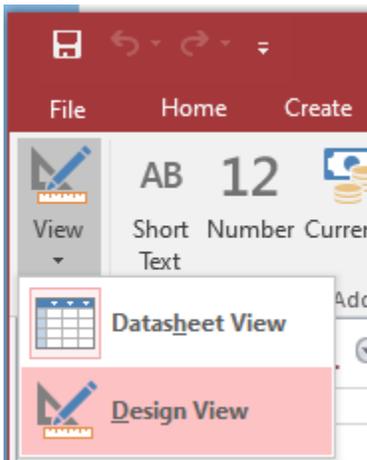


Tables

The tendency of Access is to try and set things up for you, but databases need you to actually do a lot of the heavy lifting on occasion, as the default settings may not always work for different types of databases. The perfect example is simply starting a new table.



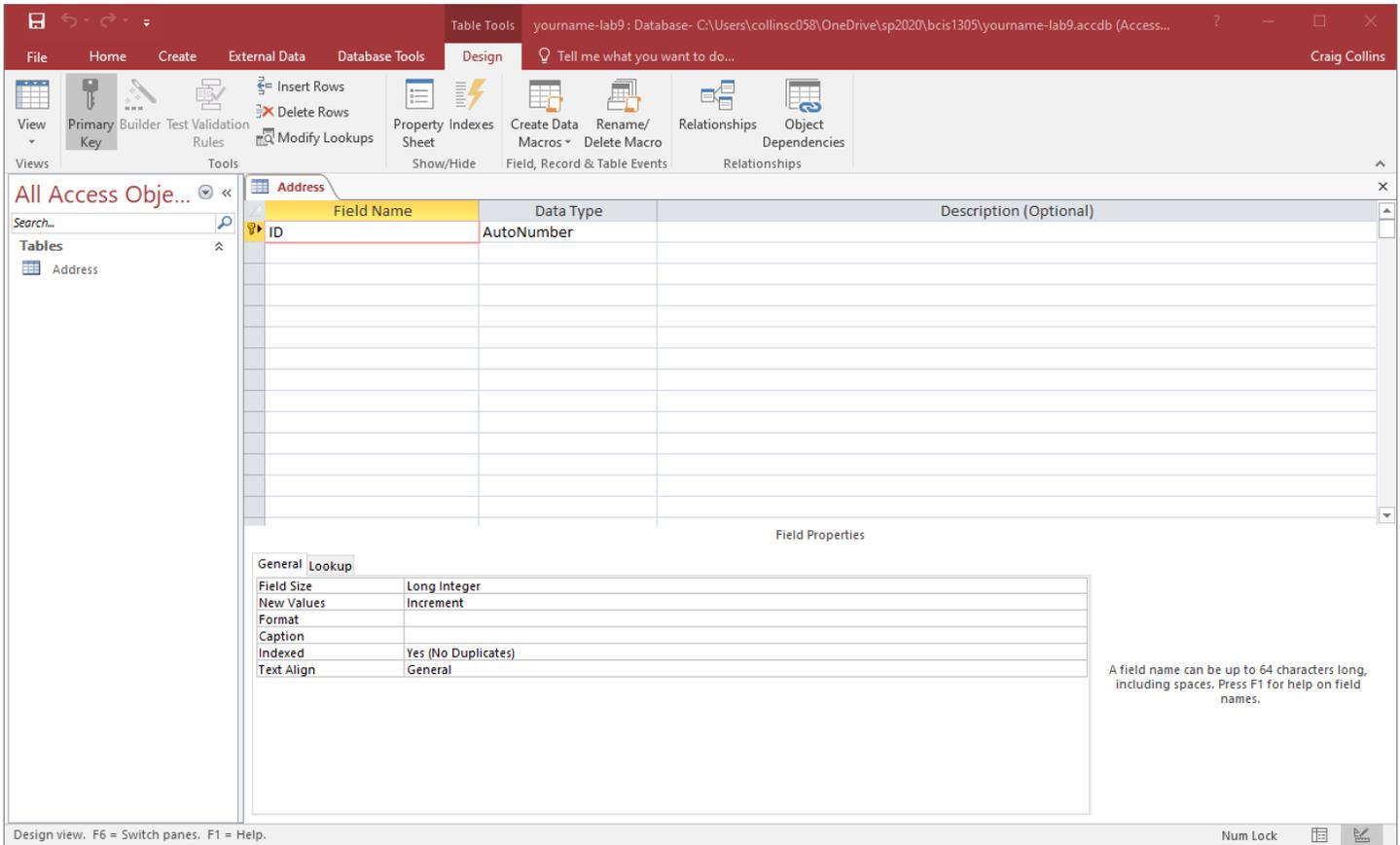
While fine for other applications, the new table adds an ID field which we do not want, and we really should be in Design view, not the normal Datasheet view. Click the arrow below the View icon, and select Design View



Databases want you to Save objects when the structures like objects are changed... Name the table: Address

Note, later when actually adding items to this table we will not need to save as we go, Databases *expect* you to add items. You only Save when changing an object's structure.

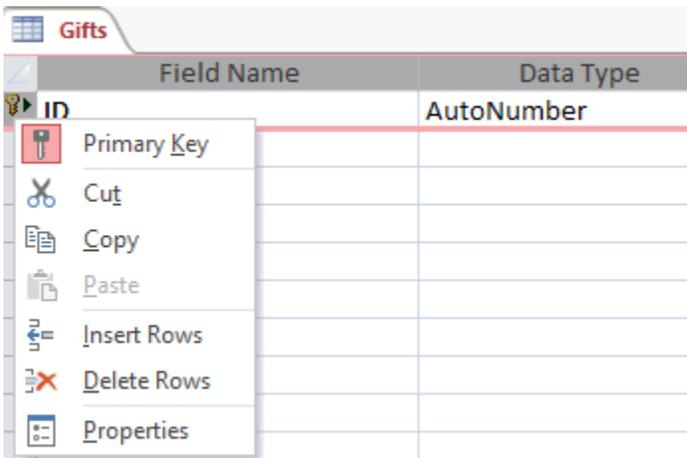
Here is the Design view.



Notice that the default setting for the first field is to make it the primary key field, and to make the name of the field ID, and that the field is expecting the data type AutoNumber.

While fine for other applications, this is not what I want now.

Right click the key icon, and select Delete Rows



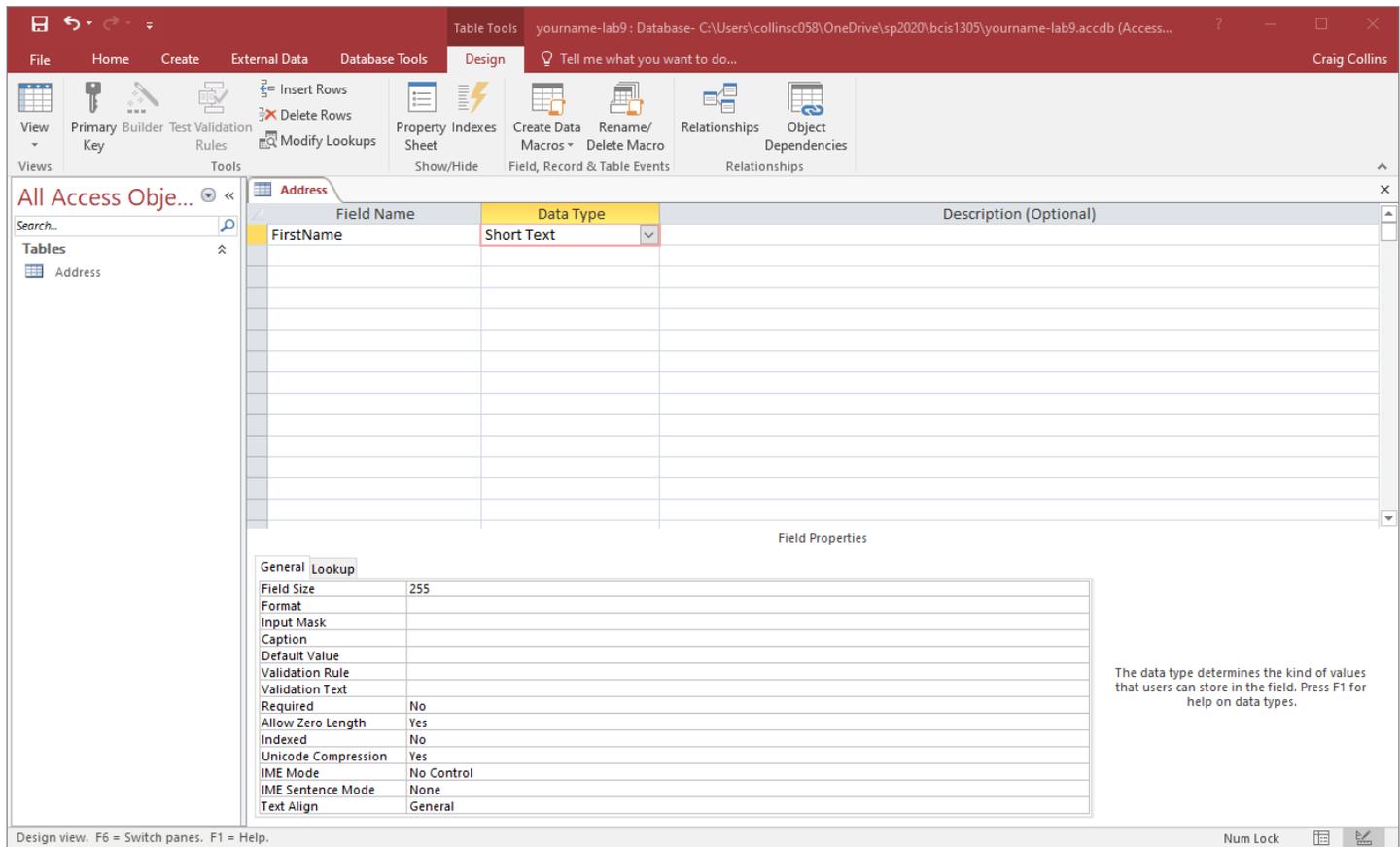
If challenged, tell Access you really want to delete.

Now we'll build this the way we want.

We are not looking at the table, but rather the Data Dictionary, where we describe what is going into the table. So all that we are doing now will modify the FIRST column of the table. Recall, in the final table columns are fields, and when we add records, those become the row. This will make more sense soon.

Select below the Field name in first row... for Field Name type FirstName, then press tab.

For data type, select Short text.



Now, **right click** in the box to the left of FirstName, and make this field the key field, the Primary key.



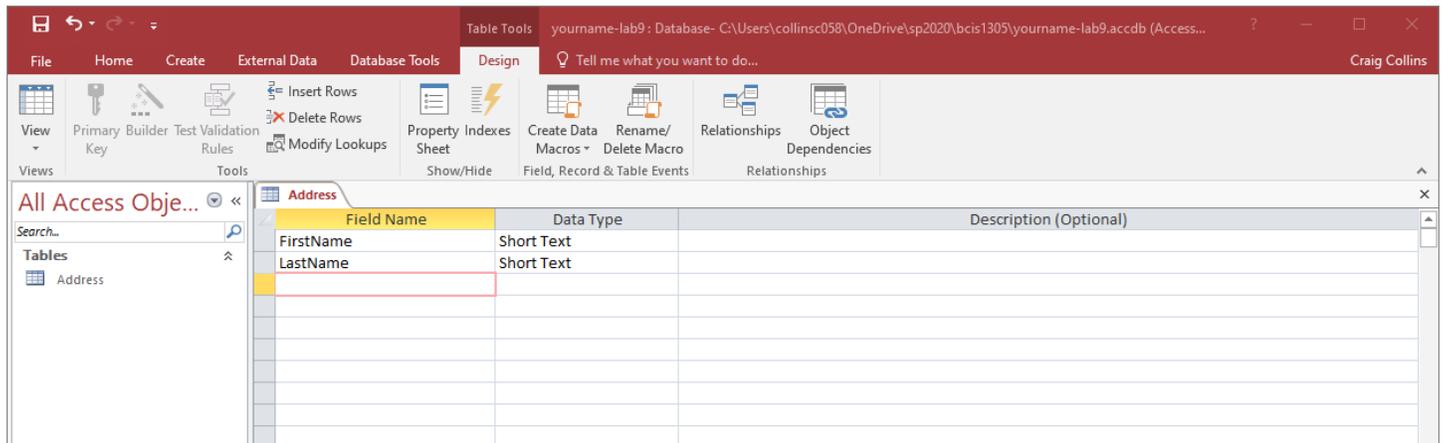
Recall, the primary key is the unique identifier. When I grew up, my Father's name was Robert Claude Collins. My Name was Robert Craig Collins. He went by Robert, I went by Craig, but on all paperwork forms I was Robert C Collins, and so was my father. We had the same address, the same phone number. So, how did the Government keep us separate? Our Social Security numbers were different and unique.

I do not want to assign ID numbers to my friends for this silly example, so I decided to make the first name the primary key, the key field.

There is a downside to choosing the first name as the primary key, and that is each person I add will have to have a different first name... so if I had two friends named John, I might have to adjust the other John when entering his name, maybe to Johnny.

Now it is time to add the other fields I want to capture in my address book, LastName, Address, City, State, Zip, and Phone.

Select the box under Field Name on second row, and type LastName, press tab and select Short Text. Our field name is pretty descriptive, so press tab twice to get to the third Field we need.

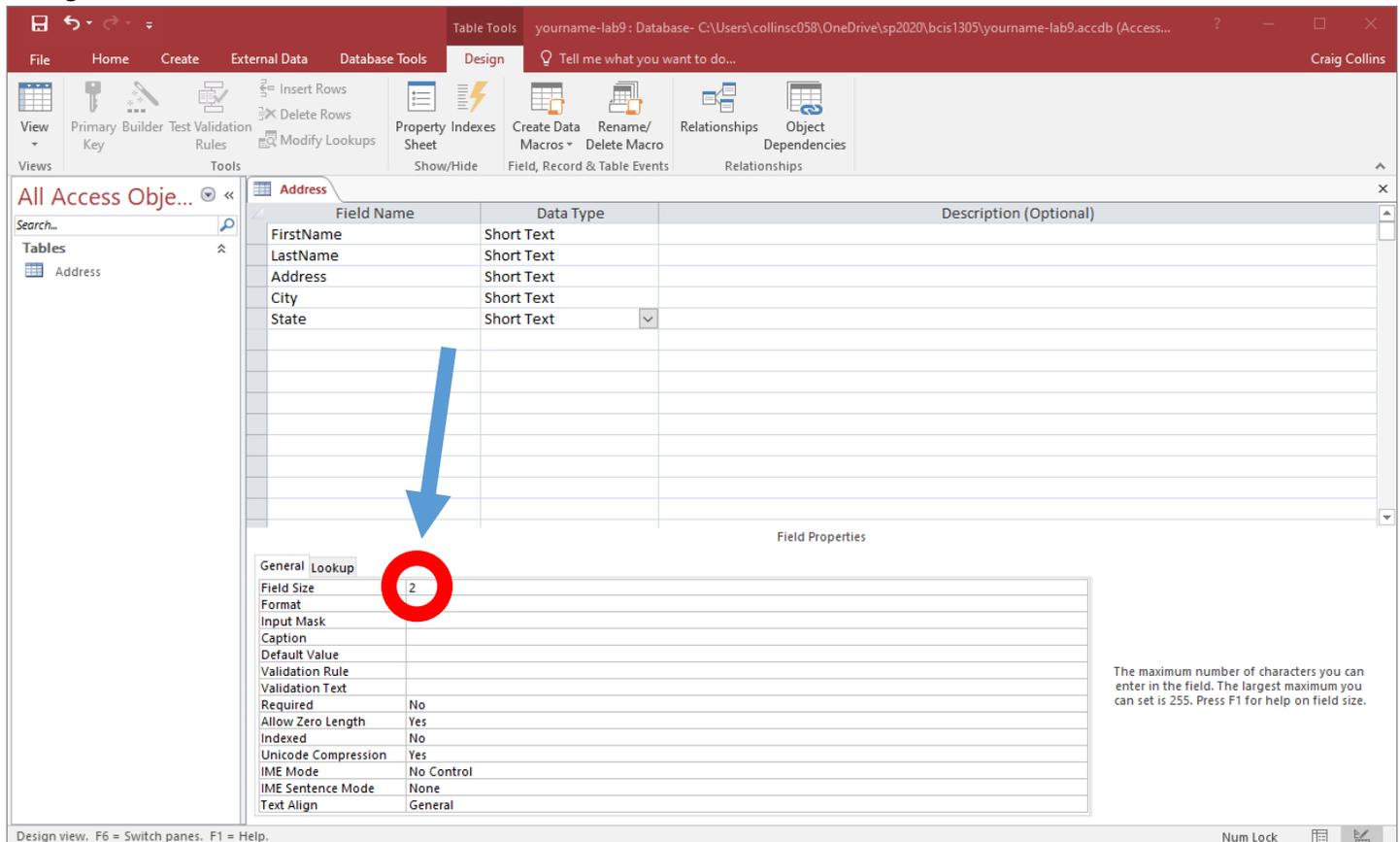


Type in Address, and tab. You might be tempted to change this from Short Text to Number, but you don't do math on an address, leave it at Short Text. Follow the steps above to add City as Short Text. Next line will be State, and Short Text.

But now we will drop to the bottom of the page to start adding some validation.

We want to prevent bad information from getting into a database. You can create Rules to prevent you adding a negative number to a price for example, or use an input mask to allow only 10 numbers to be put in a phone number. But we'll start simply, we will limit the field size to 2. I'll be some of you might spell Connecticut or Massachusetts differently each time you tried... so we will try to avoid this issue by using only a 2 letter abbreviation.

Change field size from 255 to 2.



Now we'll try in an input mask. Back to the top of the page, click the box under State, and add Zip. Again, you might be tempted to make this a number, but again, you don't do math on a zip code. Make it Short Text.

Now, find the row below that says Input Mask, and click it. You should see [...] appear. The ellipsis means a dialog box will open to help us finish. Click the [...]

General	Lookup
Field Size	255
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	

We are about to change the structure, so Yes, Save now.

Select Zip Code, and then click Next.

Input Mask Wizard

Which input mask matches how you want data to look?

To see how a selected mask works, use the Try It box.

To change the Input Mask list, click the Edit List button.

Input Mask:	Data Look:
Phone Number	(206) 555-1212
Social Security Number	831-86-7180
Zip Code	98052-6399
Extension	63215
Password	*****
Long Time	1:12:00 PM

Try It:

You'll see the following:

Input Mask Wizard

Do you want to change the input mask?

Input Mask Name: Zip Code

Input Mask:

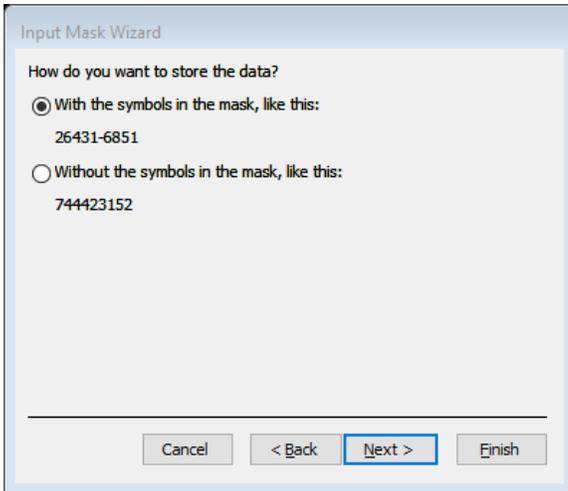
What placeholder character do you want the field to display?

Placeholders are replaced as you enter data into the field.

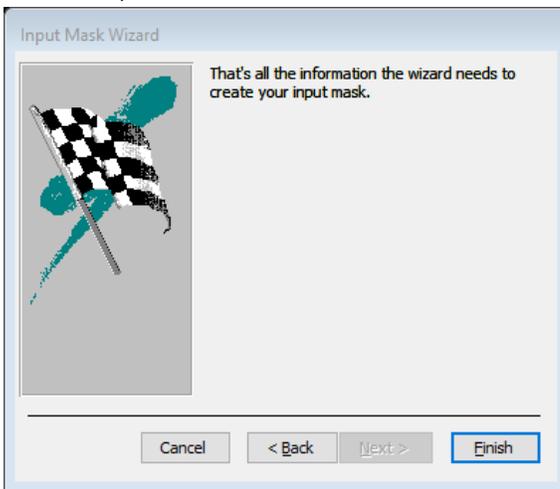
Placeholder character:

Try It:

A 0 is required, 9 means it is optional. You probably know your 5 digit zip code, but maybe not the last 4, so this means you won't have to add the last 4. Click Next. I like the symbols, so I will select the top option:



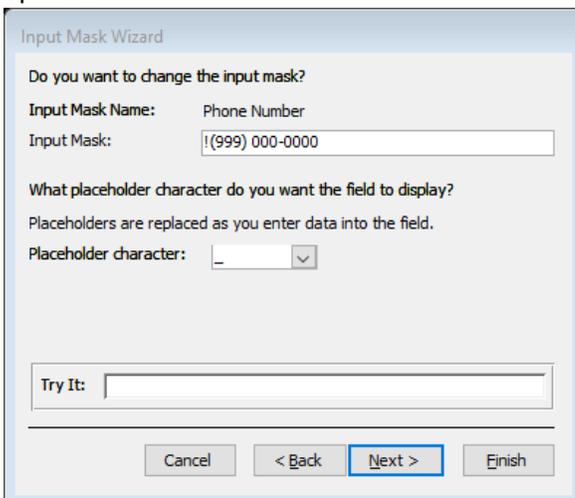
Click Next, then click Finish.



It adds some garbage, 00000\ -9999;0;_ but we know what the Wizard just added for us, even if we don't quite understand how it is represented.

Back to the top of the page, and below Zip add Phone as Short Text.

Repeat the input process shown before, but select Phone Number instead of Zip Code. I like that the area code is optional.



Next

With the symbols is my choice.

Input Mask Wizard

How do you want to store the data?

With the symbols in the mask, like this:
(814) 287-2621

Without the symbols in the mask, like this:
0373242304

Cancel < Back Next > Finish

Next and Save.

Now, close the Address Table, by clicking the [x] on the Address row, to the right. Save the changes.

Design view. F6 = Switch panes. F1 = Help.

Field Name	Data Type	Description (Optional)
FirstName	Short Text	
LastName	Short Text	
Address	Short Text	
City	Short Text	
State	Short Text	
Zip	Short Text	
Phone	Short Text	

Field Properties

Property	Value
Field Size	255
Format	
Input Mask	!(999) !000!-0000!0_
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	No
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Text Align	General

Now you can repeat this whole process to make a new table, called Gifts.

Click the **Create** tab, and select Table Design.

Now add your fields. They will be FirstName field, set as Key field, Short Text for data type. Then add

Gift, Adjective, Room, Cost, and Store fields (Cost is currency, not short text for data type; the others will be Short Text)

Design view. F6 = Switch panes. F1 = Help.

Field Name	Data Type
FirstName	Short Text
gift	Short Text
adj	Short Text
room	Short Text
cost	Currency
Store	Short Text

Close the Table.

Forms

Database administrators are paid a lot of money to design a database, thinking about a choices and how they relate to each other. Companies hire entry level people to add the records... and companies do not trust entry level people to mess with the table.

Just as you go to Amazon and fill out a form for them to capture you name, address, etc., you use a form in a database to add the row, or records, to the Table.

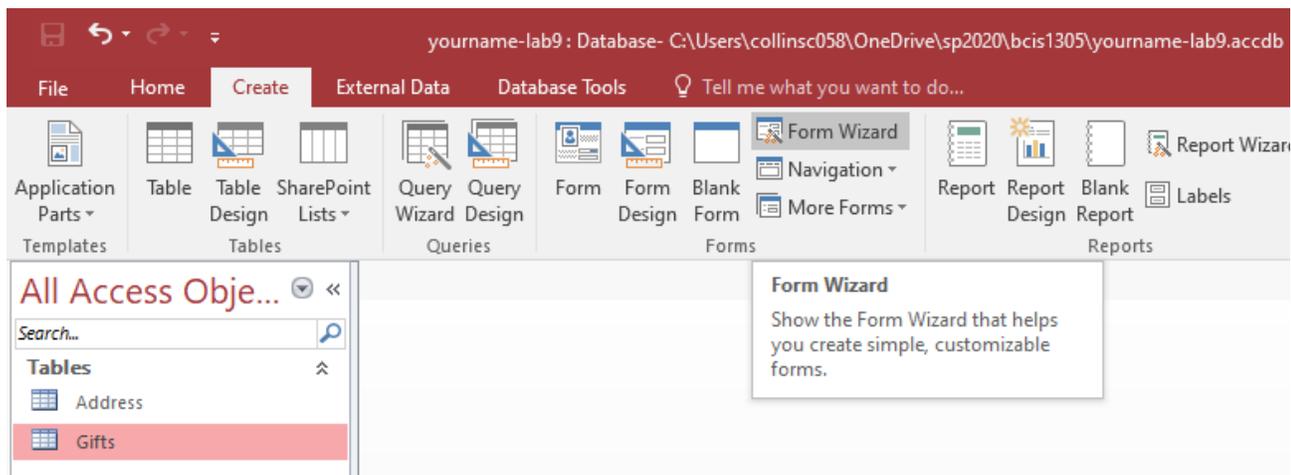
The form is another way to maintain data integrity, to keep garbage out, as you can't change the table structure from a form, just add to the table.

We'll create an input form with the Form Wizard called Address-form. And then we'll create an input form with the Form Wizard called Gifts-form.

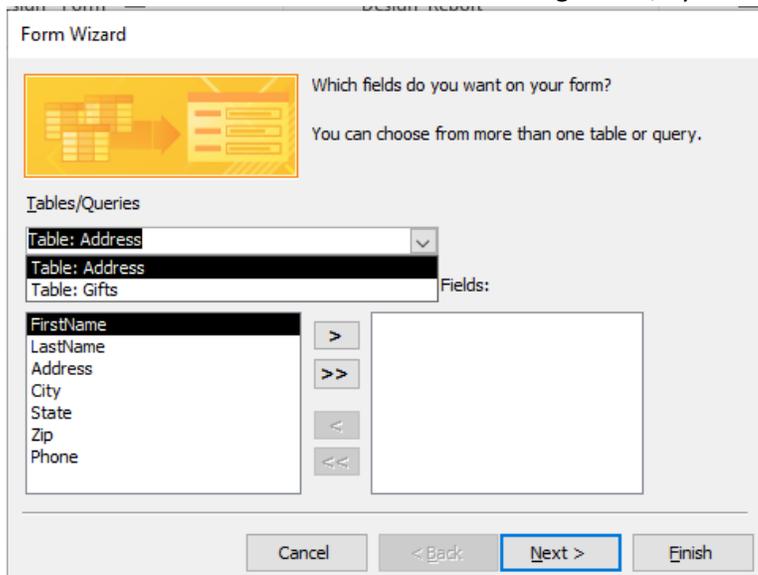
Later, we'll add about 5 friends to the Address table, using the form, and then add one gift per addressee that you added in the Gifts table.

Note: You must use the EXACT same First Names in the Gifts table that you used in Address. We'll vary the cost range from below \$10 to above \$10, to way above \$10.

Start by going to the Create tab, and select Form Wizard



You will need to select the Address table to begin with, by selecting the [V] arrow on the selector below Tables/Queries.



Then click the [>>] button to load all the fields in the Address table into the form,

Form Wizard

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

Tables/Queries
Table: Address

Available Fields:

Selected Fields:

FirstName
LastName
Address
City
State
Zip
Phone

Cancel < Back Next > Finish

then select Next. For this first database, we will accept the option presented, Columnar,

Form Wizard

What layout would you like for your form?

Columnar
Tabular
Datasheet
Justified

Cancel < Back Next > Finish

then Next,

Form Wizard

What title do you want for your form?
Address-form

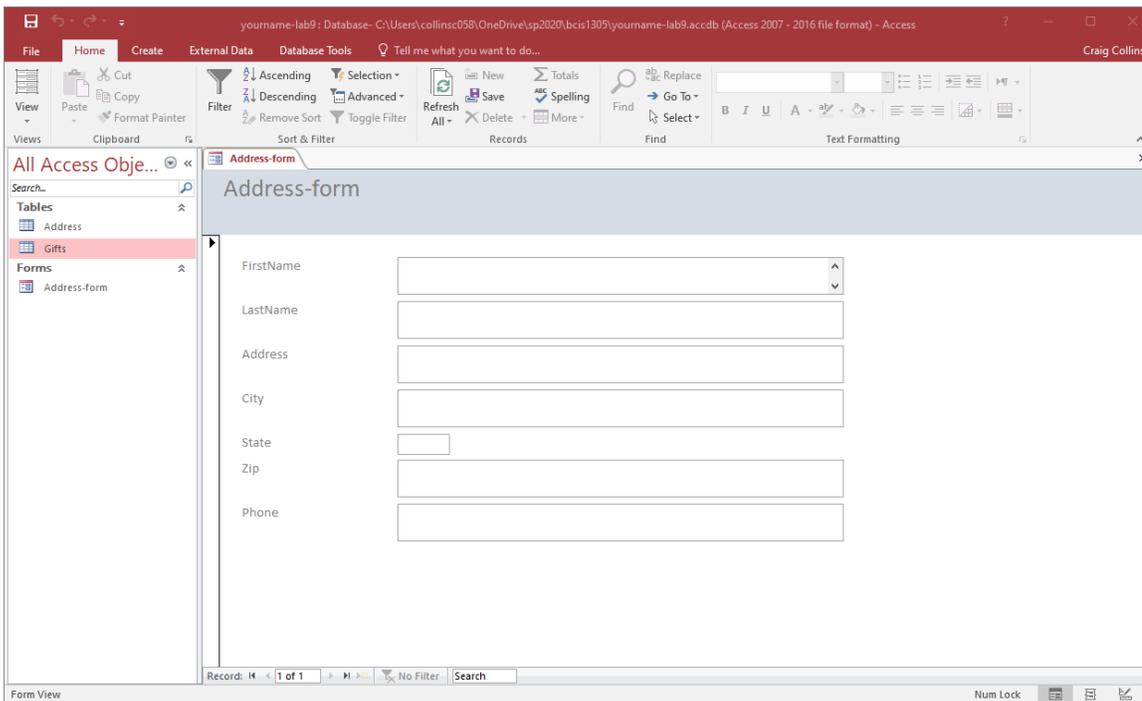
That's all the information the wizard needs to create your form.
Do you want to open the form or modify the form's design?

Open the form to view or enter information.
Modify the form's design.

Cancel < Back Next > Finish

and change the name to Address-form, and Finish.

This will open the form, to let you add records indirectly to the table.



Type a name into FirstName, then press Tab.

Add a last name to LastName, then press tab.

Add a street address to Address, then press tab.

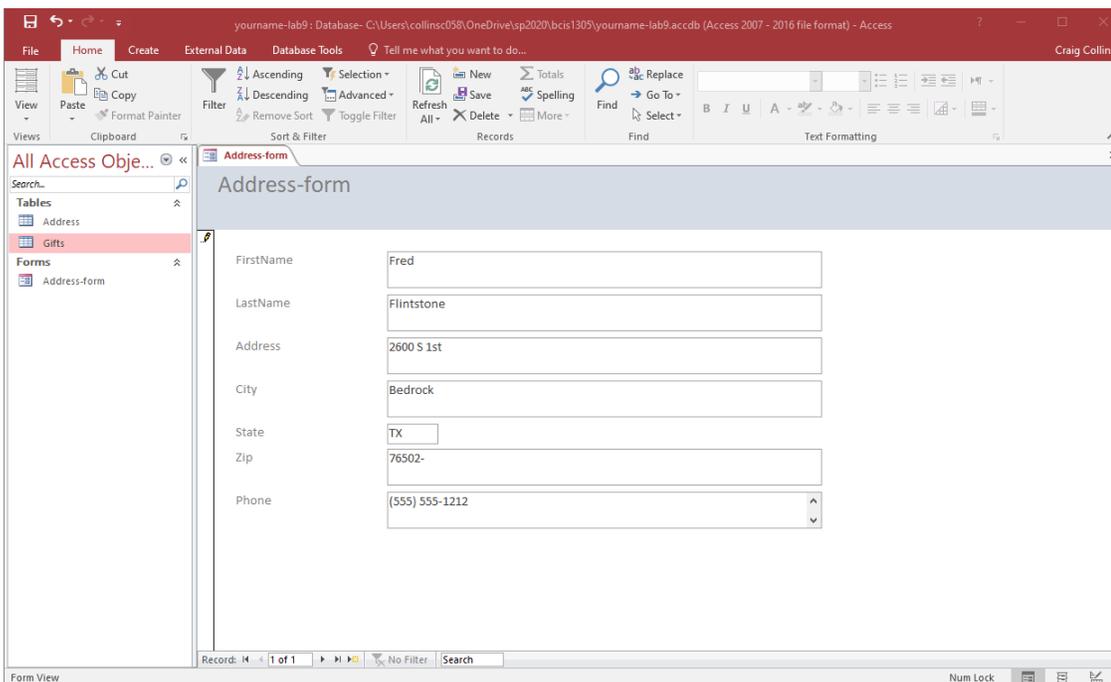
Add a City name to City, then press tab.

Now, try to spell Texas in the State field. It won't let you go beyond 2 characters. This is validation in action.

Change State to TX and press tab.

Try typing abc in the Zip field. It won't let you type letters, it is expecting numbers. You can type 5 or 9 numbers... not less, not more. Again validation.

Type in a phone number for Phone.

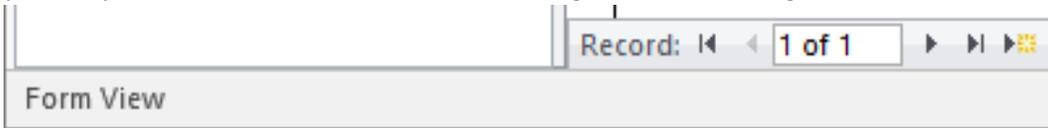


If you were finished, you could close the form, but we need to add more records, 3 for now at least, if not more.

To add a new record, you can press Tab from the last field on the form

or

you can press the > on the bottom line, to the right of "1 of 1," to go to the next record,

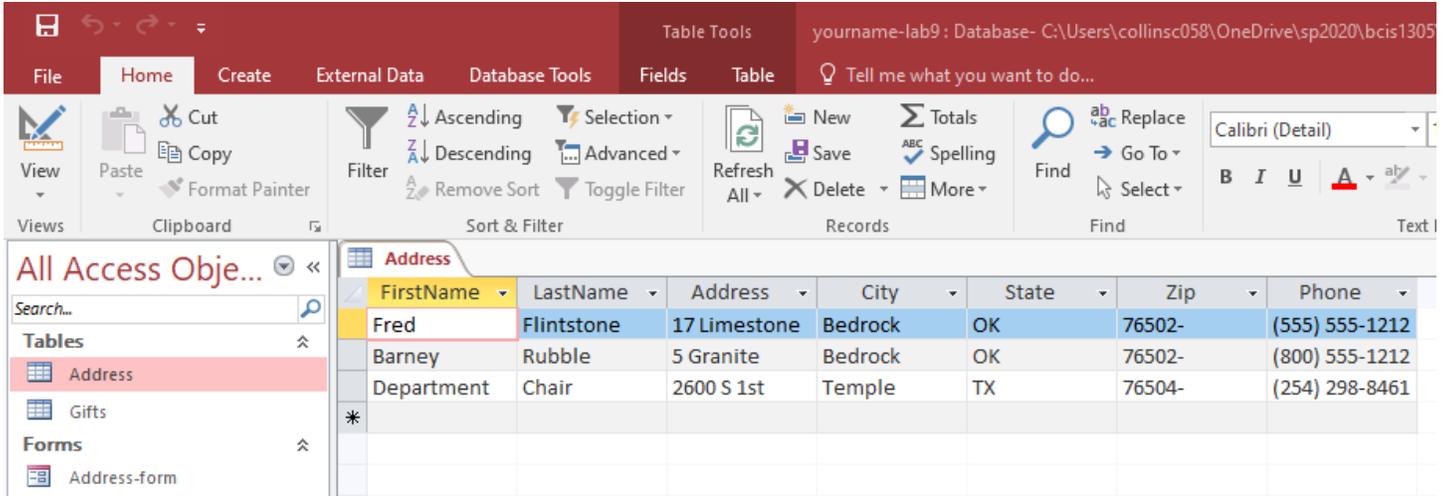


or

you can press the |> on the bottom line to the right of 1 of 1 to go to the end which would create a new record, or

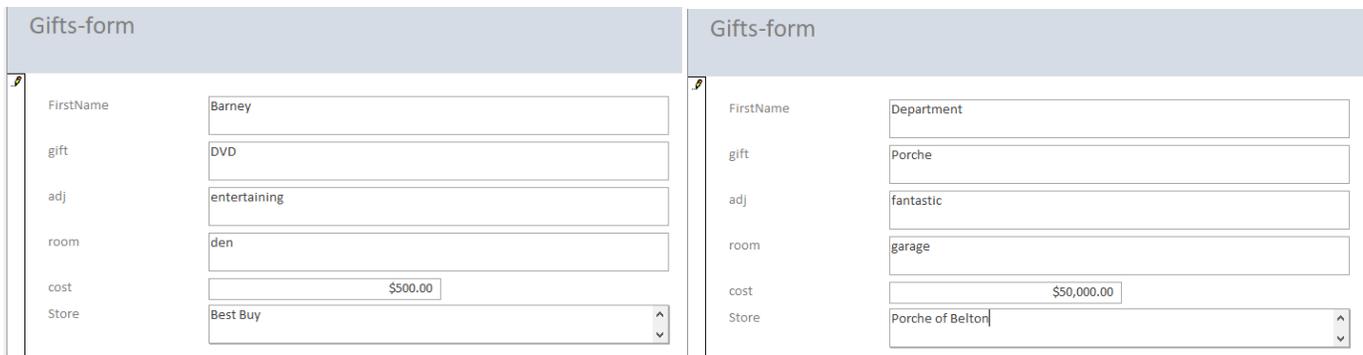
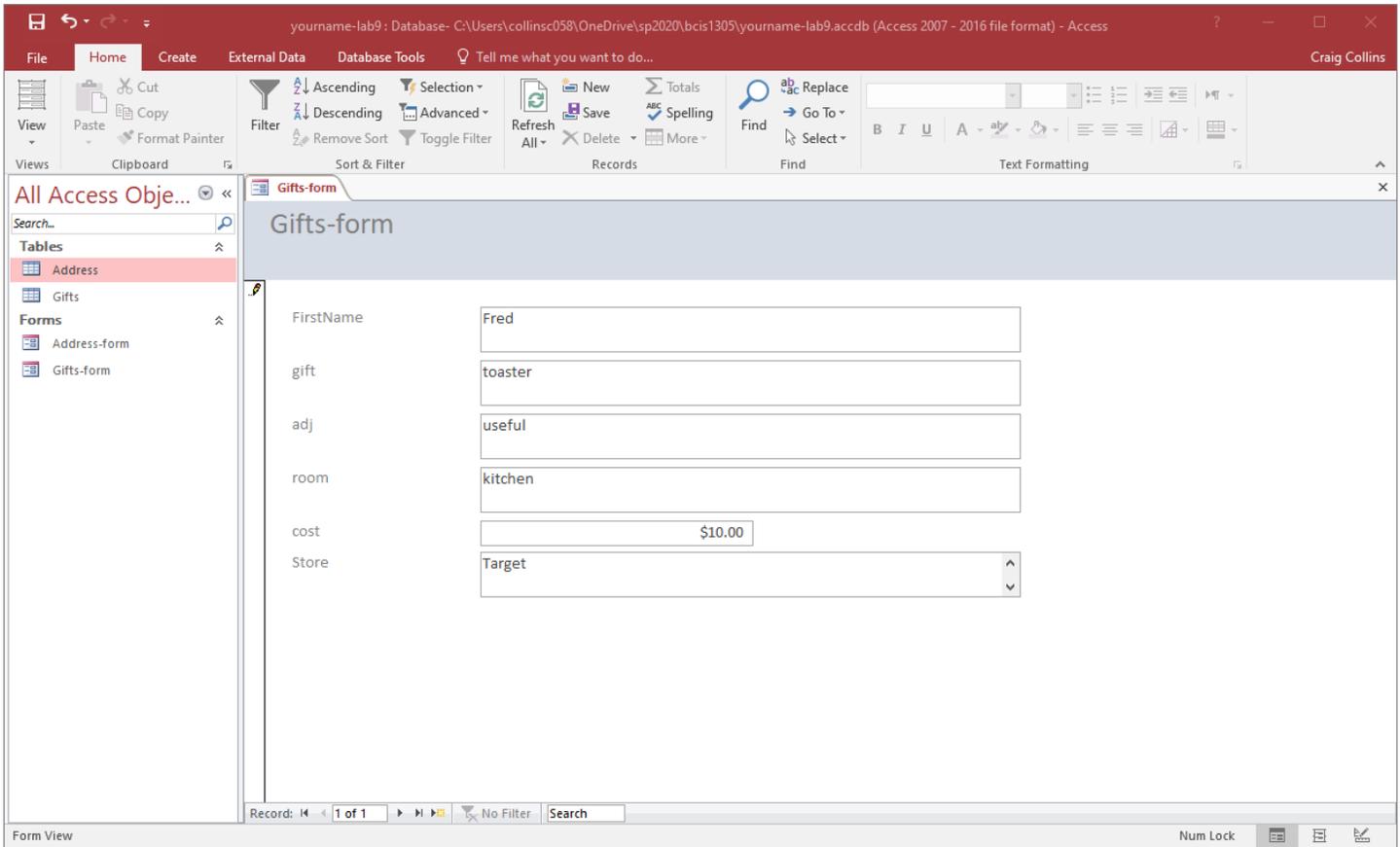
you can press the >* on the bottom line to the right of 1 of 1 to create a new record.

To verify the table was updated, close the form and then double click the Address table.



Close the table.

Repeat the steps to populate the Gifts table, by creating a Gifts-form, as above.
 You MUST use exactly the same name for FirstName for each record that you used earlier.

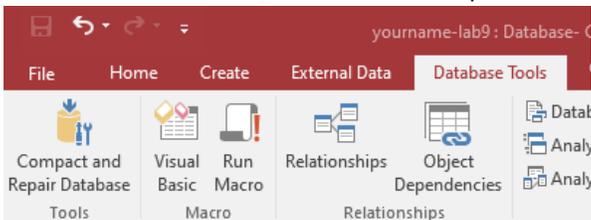


Make sure you add enough records... but for every person in Address, they also need to be in Gifts, for this simple exercise.

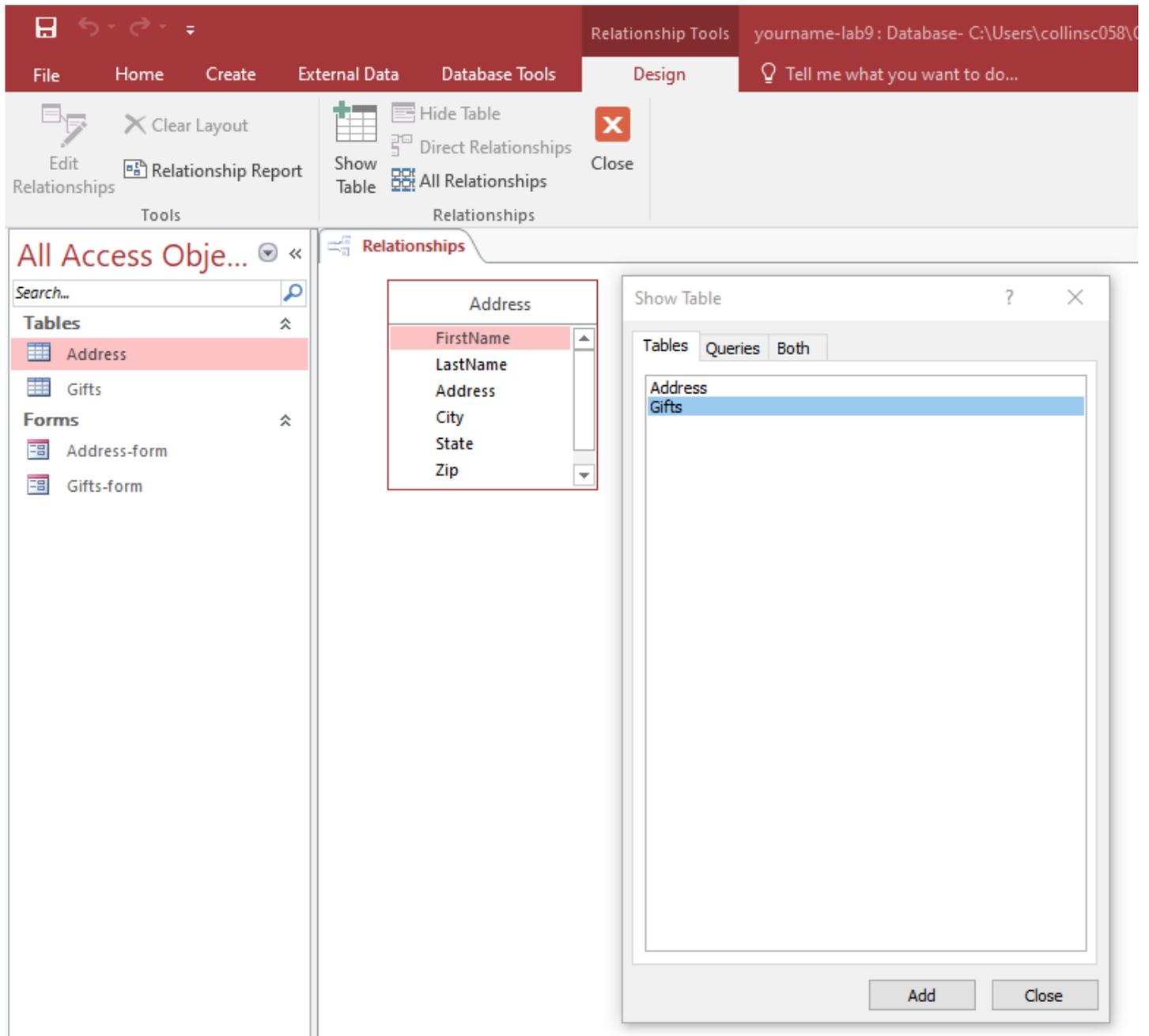
Relationship

A relationship joins different tables together, to let you view information in more than one table at the same time. Since FirstName is in both tables, we have a common field that we can use.

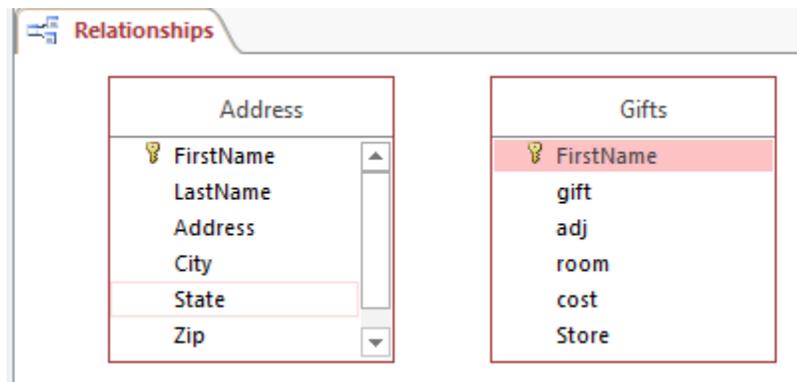
Select Database Tools, then Relationships



Select Address from the tables tab, and choose Add;

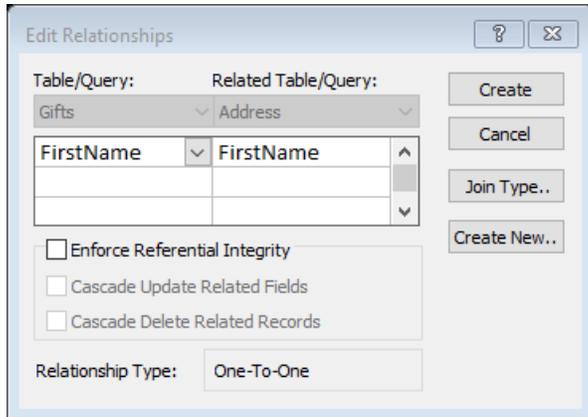


then select Gifts from the tables tab, and choose Add. Close the Show Table dialog box.



This is the fun part. Click on one of the FirstName fields and drag it on top of the other FirstName field.

You'll see the following, we'll accept this limited relationship for this simple model... meaning for each person in the Address table, there is the same person in the Gifts table... a one-to-one relationship.



Click create. A line joins the two fields. Try to drag one box around... they stay related. Close the Relationship window, and Save the changes.

There are three types of table relationships in Access.

A one-to-many relationship

Let's use an order tracking database that includes a Customers table and an Orders table as an example. A customer can place any number of orders. It follows that for any customer represented in the Customers table, there might be many orders represented in the Orders table. The relationship between the Customers table and the Orders table is a one-to-many relationship.

To represent a one-to-many relationship in your database design, take the primary key on the "one" side of the relationship and add it as an additional field or fields to the table on the "many" side of the relationship. In this case, for example, you add a new field — the ID field from the Customers table — to the Orders table and name it Customer ID. Access can then use the Customer ID number in the Orders table to locate the correct customer for each order.

A many-to-many relationship

Now let's look at the relationship between a Products table and an Orders table. A single order can include more than one product. On the other hand, a single product can appear on many orders. Therefore, for each record in the Orders table, there can be many records in the Products table. In addition, for each record in the Products table, there can be many records in the Orders table. This relationship is called a many-to-many relationship. Note that to detect existing many-to-many relationships between your tables, it is important that you consider both sides of the relationship.

To represent a many-to-many relationship, you must create a third table, often called a junction table, that breaks down the many-to-many relationship into two one-to-many relationships. You insert the primary key from each of the two tables into the third table. As a result, the third table records each occurrence, or instance, of the relationship. For example, the Orders table and the Products table have a many-to-many relationship that is defined by creating two one-to-many relationships to the Order Details table. One order can have many products, and each product can appear on many orders.

A one-to-one relationship

In a one-to-one relationship, each record in the first table can have only one matching record in the second table, and each record in the second table can have only one matching record in the first table. This relationship is not common because, most often, the information related in this way is stored in the same table. You might use a one-to-one relationship to divide a table with many fields, to isolate part of a table for security reasons, or to store information that applies only to a subset of the main table. When you do identify such a relationship, both tables must share a common field.

From <https://support.office.com/en-us/article/guide-to-table-relationships-30446197-4fbe-457b-b992-2f6fb812b58f>

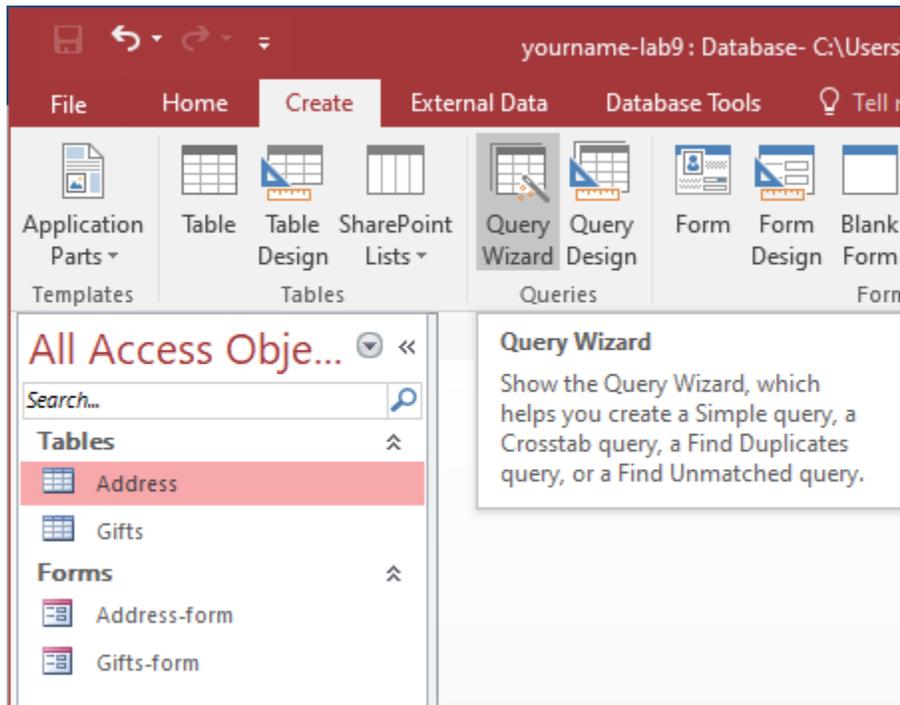
Queries

A query is a way to ask the database a question, based on <, >, =, and a few more options to discuss later. Rather like a conditional format in Excel.

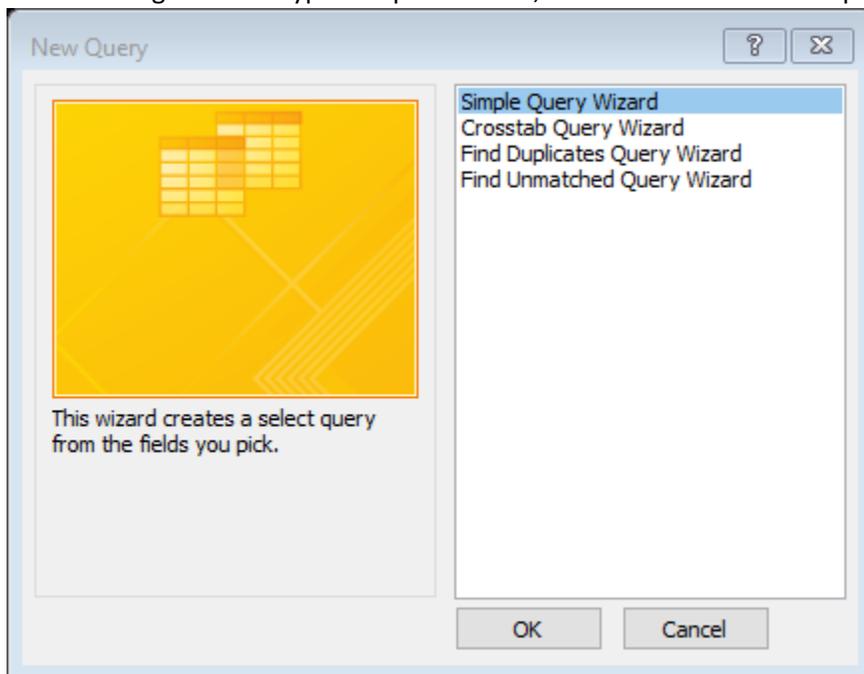
Our first query will be to see everything in both tables at once, so we will only use = to begin with.

We'll need to bring each field in to the query, but since FirstName is in both tables, we'll only select it once.

Start on the Create tab, and choose Query Wizard

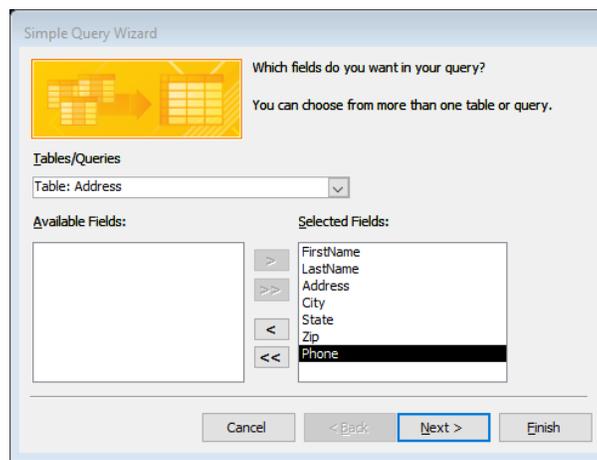
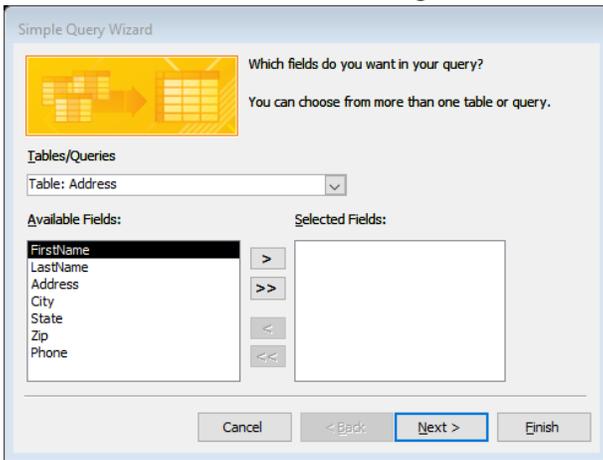


We'll investigate other types of queries later, but we'll start with a Simple Query

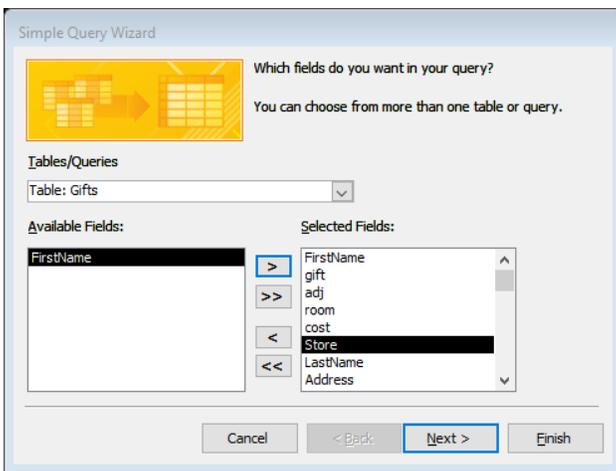


Click OK

Select the Table: Address, and bring over all the fields with >>

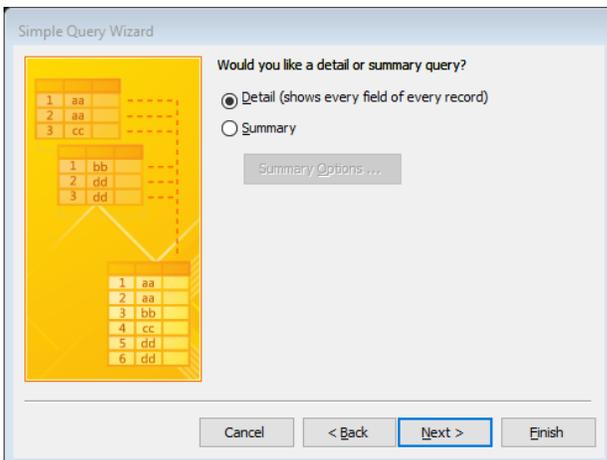


Now we need to get all but FirstName from the Table: Gifts, just select the first field below FirstName, and click > until all the fields but FirstName are moved over...



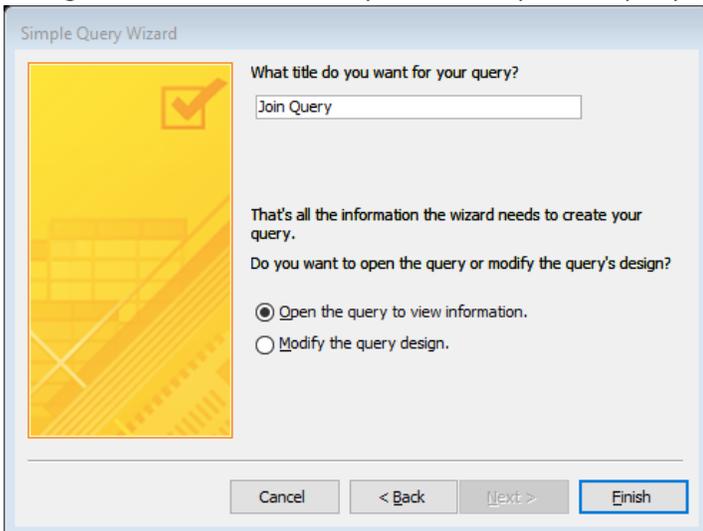
Click Next.

We want the details...



click Next.

Change the name to Join Query, and let it open the query... click Finish



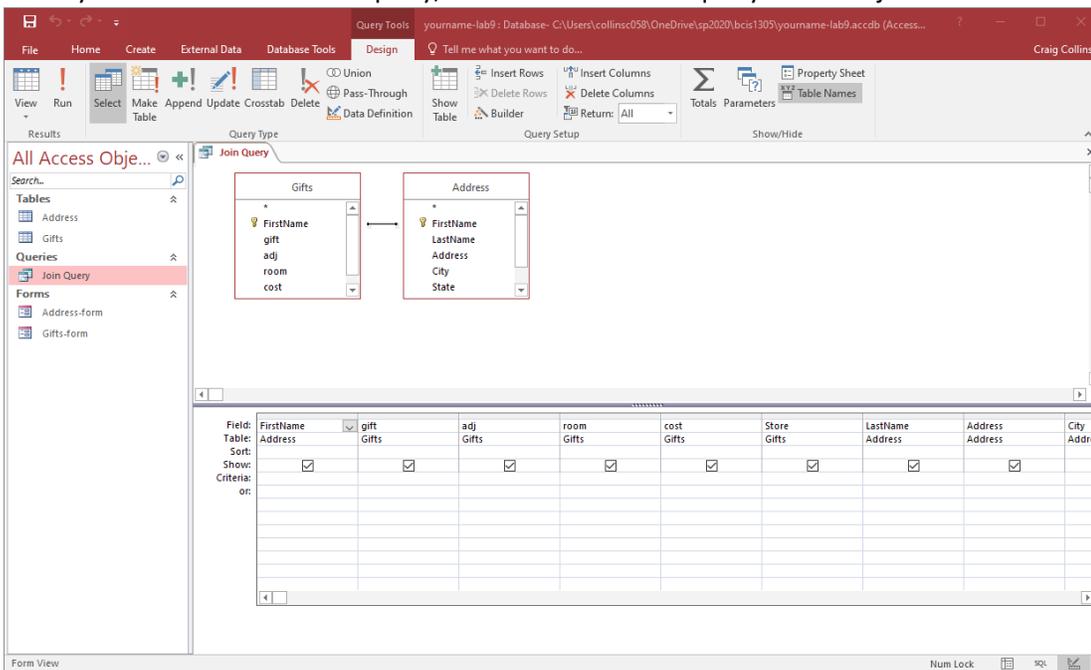
You should see the query open, and all the items you have added should be there.

Modify a query

Now let's make a new query, based on the join query, that will only show us gifts more that \$10, that is criteria for the cost field will be >10

Make sure you are on the home tab, not Create. Change to Design view.

Note: you could also close the query, and double click the query in the Objects tab.

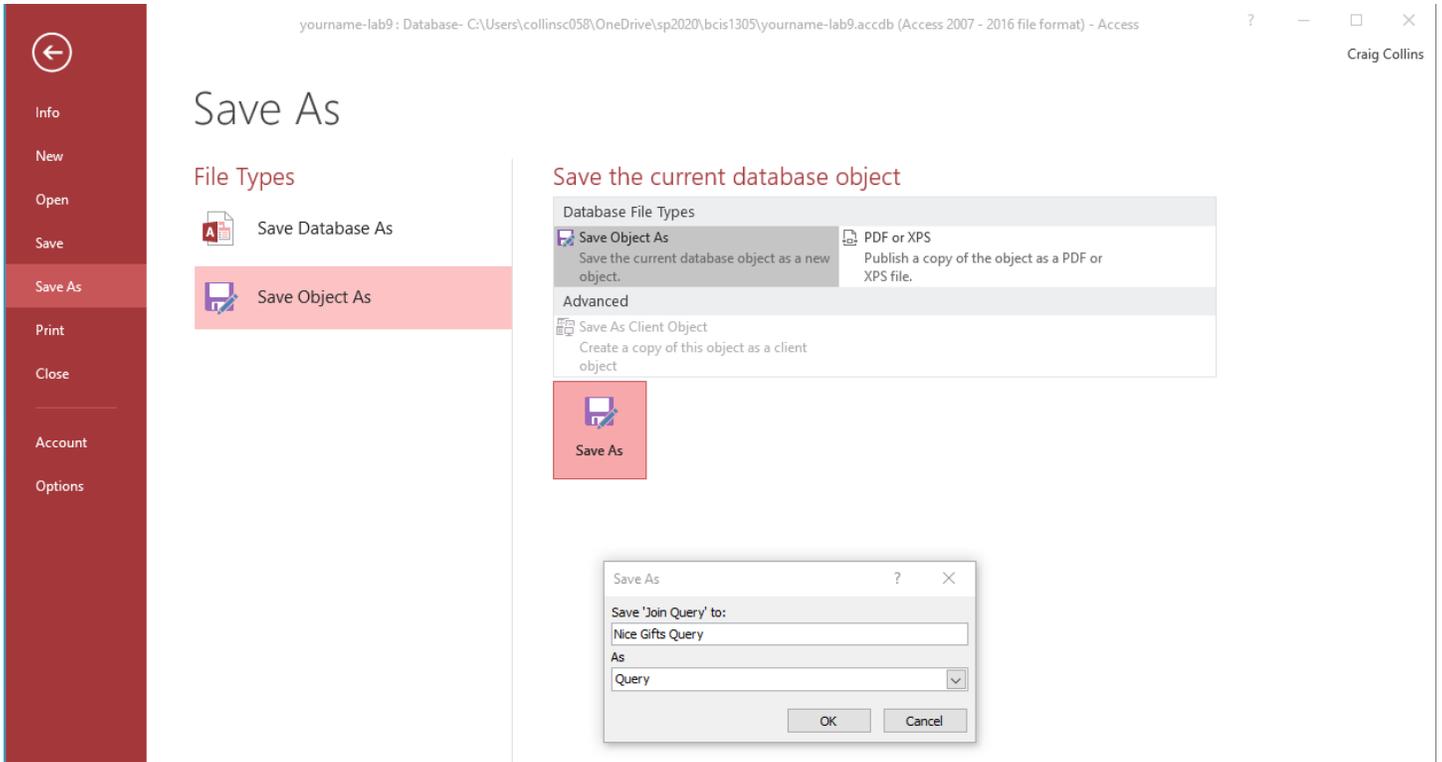


Locate the gifts field, and in the Criteria row, type >10.

Field:	FirstName	gift
Table:	Address	Gifts
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		>"10"
or:		

Then click the big ! Run button. You will only see the items that are >10.

To save this new query select File>Save As but click on Save Object As the click the Save As icon.
Name this query Nice Gifts query.



Close all the Access windows, but leave Access running.

Reports

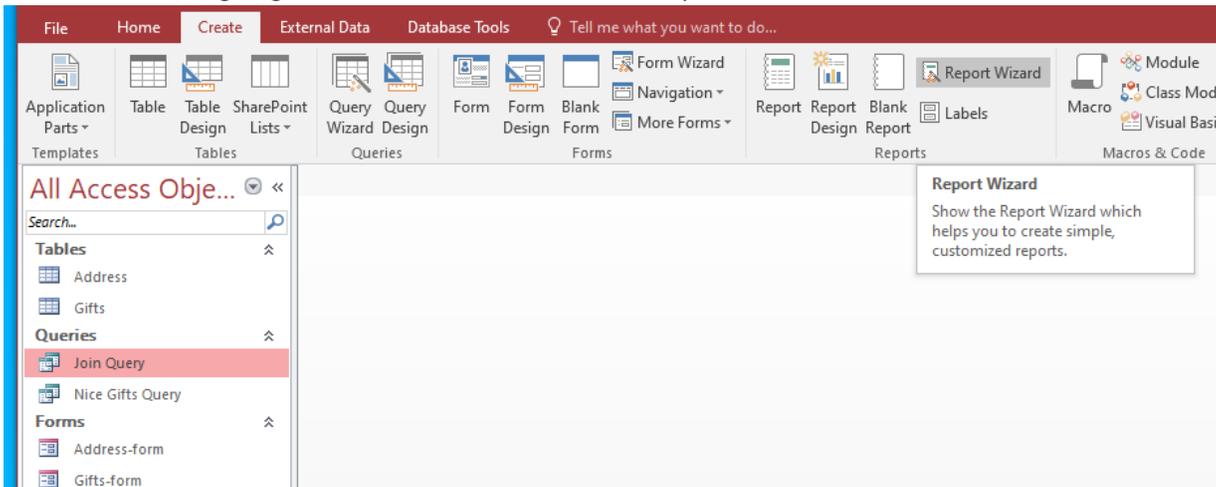
Access has very weak reporting tools; many people actually by other products, such as Crystal Reports to build better reports, but we need to complete the input-process-output cycle.

Recall input is done with a form

Process is done with a query

Output is done with a report.

Of course, we are going to the Create tab, and use the Report Wizard



Make sure the selection is set to Query: Join Query. Select all the fields with >>

We will not do any grouping or sorting, and well lay it out in the default Tabular.

Name the report Join Report, and Finish. Not a great form, but enough for this database.

A bit more detail, and a preview of additional Database topics covered in other classes

Logical operators < + > like, such as the criteria for Cost >10 will show only records that are greater than \$10
Cost<20 will show only records that are less than \$20

Criteria for State ="TX" will show only records where the State is TX

State="N*" will show only records where the State name begins with N, such as NM

A parameter query might be for State the criteria is done by typing

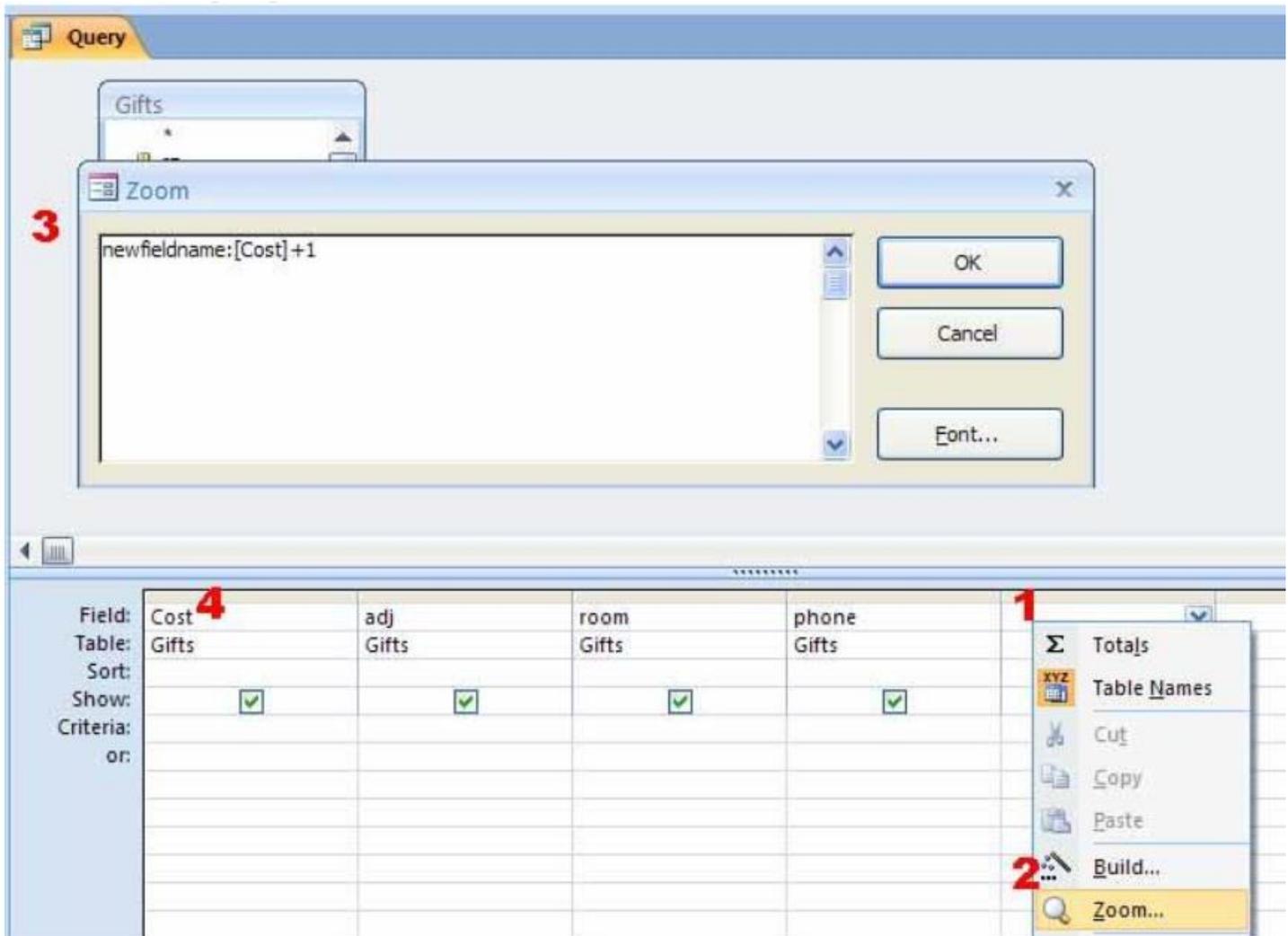
[Which State are you looking for?]

in the State field criteria box...

and you would be prompted to provide a State name when you run the query

A Calculated Field is a field that can be added to a query to compute data in other fields.

- Step 1 Place your insertion point in the first open column,
- Step 2 right click, and choose Zoom.
- Step 3 Type in what you want to call this new field, followed by a colon (:), and then the math to be performed Item 4:
If using a field for one of the values, place that field name in brackets, spelled exactly as the field name Example: to take the value in that record for Cost, and adding 1 to it would be **NewFieldName:[Cost]+1**



Assignment

Option 1 Thoroughly describe setting up the above database, include creating tables, using the data dictionary, setting the primary key, adding records with a form, creating the relationship, creating the first query, and modifying the query.

Option 2 Create & submit an Address Book/Gift register, call it yourname-lab9.accdb, and save it in a folder called lab 9

Be sure to include the following:

- ☑ A table called Address created in Design mode (5%)
- ☑ First_Name field, set as Key field (5%) Short Text for data type
- ☑ Last_Name (5%) Short Text for data type
- ☑ Address (5%) Short Text for data type
- ☑ City (5%) Short Text for data type
- ☑ State, two character max (5%) Short Text for data type
- ☑ Zip, text field with mask (5%) Short Text for data type
- ☑ Phone Number, text field with mask (5%) Short Text for data type
- ☑ Create an input form called Address-form w/Wizard (5%)
- ☑ Add at least 5 records, no duplicate names (5%)

- ☑ Create second table called Gifts (5%)
- ☑ First_Name field, set as Key field (5%) Short Text for data type
- ☑ include Gift, Adjective, Room, Cost, and Store fields (5%) (**Cost is currency**, not text; the rest will be Short Text for data type)
- ☑ Create a input form w/ Wizard called Gifts-form **based on the Gifts table** and add at least one gift per addressee that you added in the other table. **You must use the EXACT same First Names used in Address** Vary the cost range from below \$10 to above \$10, to way above \$10 (5%)
- ☑ Relate your two tables on First Name (5%)
- ☑ Create a query based on both tables, one of each field, named join-query
- ☑ Modify join-query to merge all records where gift was more or less than \$10 ie > 10 (5%)
- ☑ Save Query as Nice-Gifts-query or Cheap-Gifts-query(5%) (File\Save As\Save Object as... and save query with the new name
- ☑ Create a report based on the join-query called join-report (5%).
- ☑ Create a report based on the Nice_Gifts-query called Nice_Gifts-report (5%)
- ☑ Appropriately submit (MUST be zipped) (5%)

Preview of quiz question:

Unique identifier
a collection of tables
a way to output, or print
Input, or view and maintain data
Rows in a table
easily view the fields and data types
Question to process
data is presented in rows and columns

1. Primary Key, the key field
2. Query
3. Form
4. Report
5. Record
6. Relational Database
7. Datasheet View
8. Design view