Database theory: Using Word and Excel for basic Database functions (pdf)

Database Logic: create an address book, and link it to a list of gifts, in order to automate thank you letters.

After reading, proceed to the end of this document to see the Lab.

The Birthday loot:

Barney gave me a toaster, cost $10 from HEB
Betty gave me a DVR, cost $50 from Target, and
Wilma gave me a car, cost $50,000 from Central Texas BMW.

I need to write some thank you notes, but I would like to be able to re-task this information for later.

Option 1, great at thank you notes: I might enter the info to a word processing mail merge where info would be kept in a table, but it could be hard to pull just some info out.

Option 2, I could build a spreadsheet table to record the information, and spreadsheets are great at sorting; but terrible at making a 'thank you' note...

...sounds like a job for a database.
(After reading this document, you can refer to YouTube video https://youtu.be/2sbgzbuSaNY)

Some database terms:
Field (the equivalent of a column in a table), a group of related characters, such as 'last name' or 'zip code.'
Record (the equivalent of a row in a table), a group of related fields, usually describing an individual.
Example,

Example	Last Name	First Name	Middle Initial	Address
Me	Collins	Robert	C	555 Main, Anywhere, USA
My Dad	Collins	Robert	C	555 Main, Anywhere, USA

Note: there are two records with the same info (since middle name was not used)
Primary Key, a key field, this uniquely identifies ONE individual.

Example	Last Name	First Name	Middle Initial	Address	ID No. (key)
Me	Collins	Robert	C	555 Main, Anywhere, USA	555-12-1212
My Dad	Collins	Robert	C	555 Main, Anywhere, USA	555-12-1213

Often, a good candidate for key field is just to add a sequence number... such as record #1 or record #2

File, a series of related records is called a file, such as all the College 'Address' records table is a file.
Database, a series of related files is a database.

If you segregate your data, to protect who can assess certain parts, you now have a **relational database**...
In order to have a working relational database each database must share a field, often the key field.

So I need to build my data dictionary, where I describe my birthday loot.
Here are the fields I want to use so I can write a great thank you,
and since I am planning ahead, I also will grab some info so I can re-task the data later:
name, gift, room (where I'll keep the gift), ***adjective*** (to describe the gift), ***cost***, and ***store*** where purchased.

That way I could write the following letter

Planning a generic letter that a database could *personalize*

Dear [name], thanks for the [gift]. I will think about you every time I go into the [room].	After being run through a database, it will replace each field name with information from ONE record, letting me print three 'personalized' letters.	Dear Barney, thanks for the toaster. I will think about you every time I go into the kitchen. Dear Betty, thanks for the DVR. I will think about you every time I go into the Living room. Dear Wilma, thanks for the car. I will think about you every time I go into the garage.

Plus I could re-task this data to also generate the following letter:

Dear Target,
I recently got a toaster.
I believe it cost $10.
I'd like to get a refund.

Same info, different uses :)

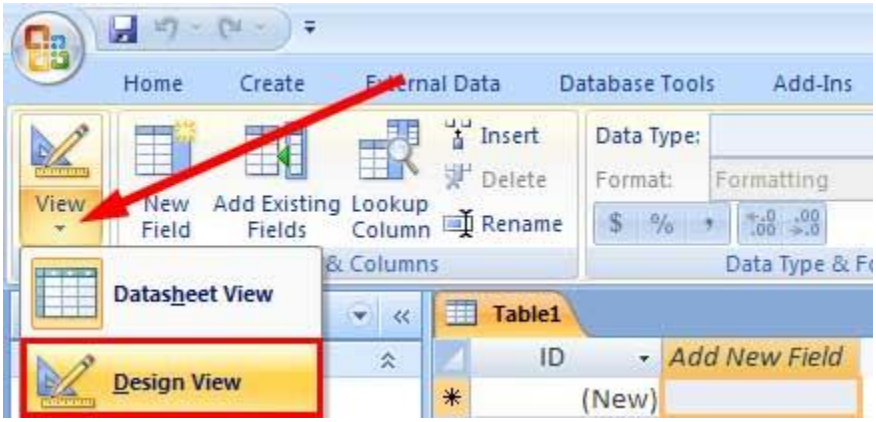
But, I sure don't want to return the BMW... so I only want to generate a return letter, if certain criteria are met, such as Gift ***Cost*** < $15.

More database terms:

Form, a method of adding records to a table (input)
Query, a method of selecting records (process), and
Report, a method of printing relevant parts of records (output)

Building the Database using Access (Some screenshots are from older versions, but still applicable)
YouTube video https://youtu.be/2sbgzbuSaNY

QUICK OVEVIEW After creating a blank database and saving , choose the triangle under View, and switch to Design View for your table.(Your screen may differ slightly; step by step follows.



Upon switching views, you will be prompted to save the table before proceeding.

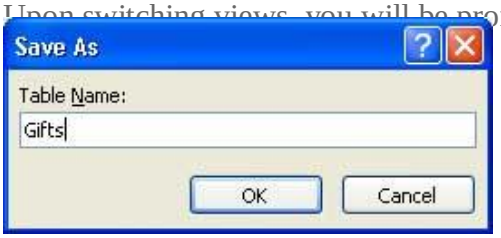
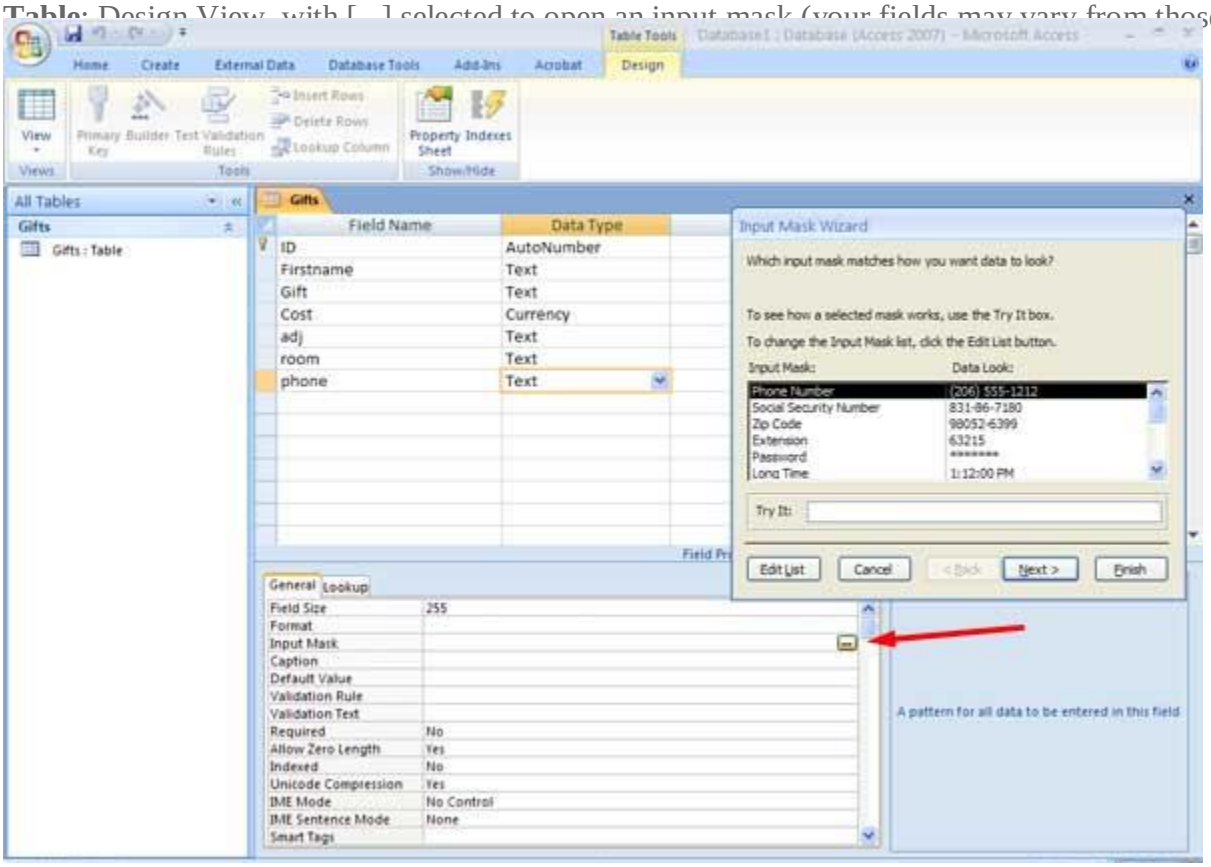
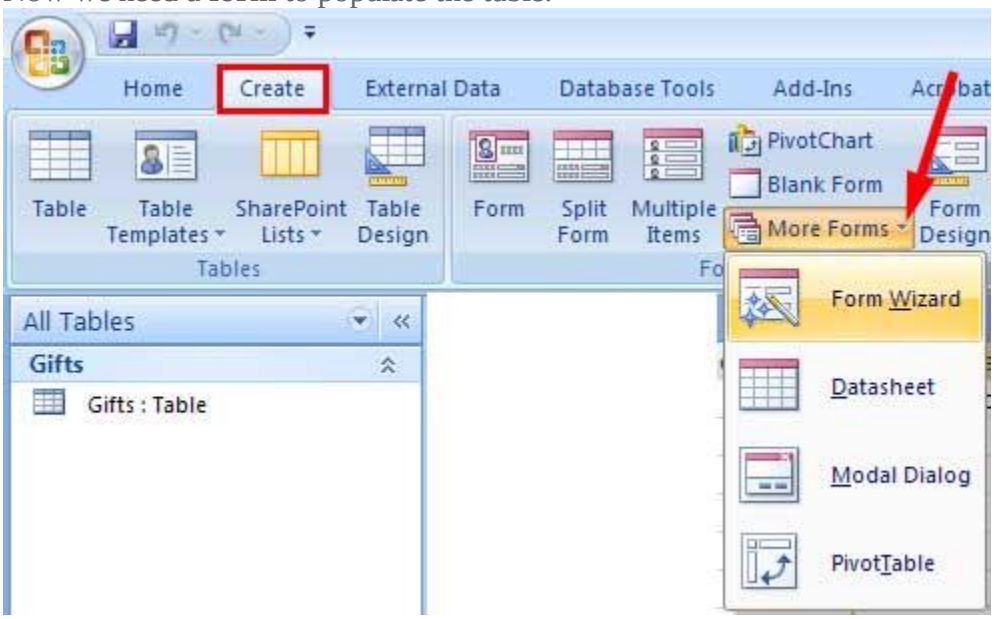


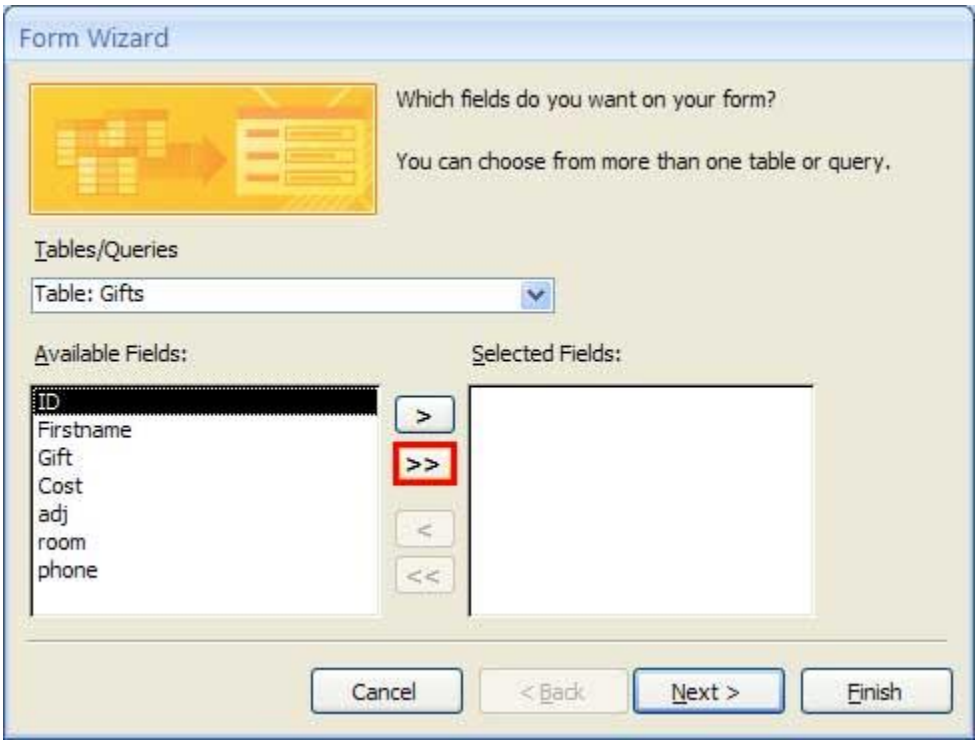
Table: Design View, with [...] selected to open an input mask (your fields may vary from those shown)



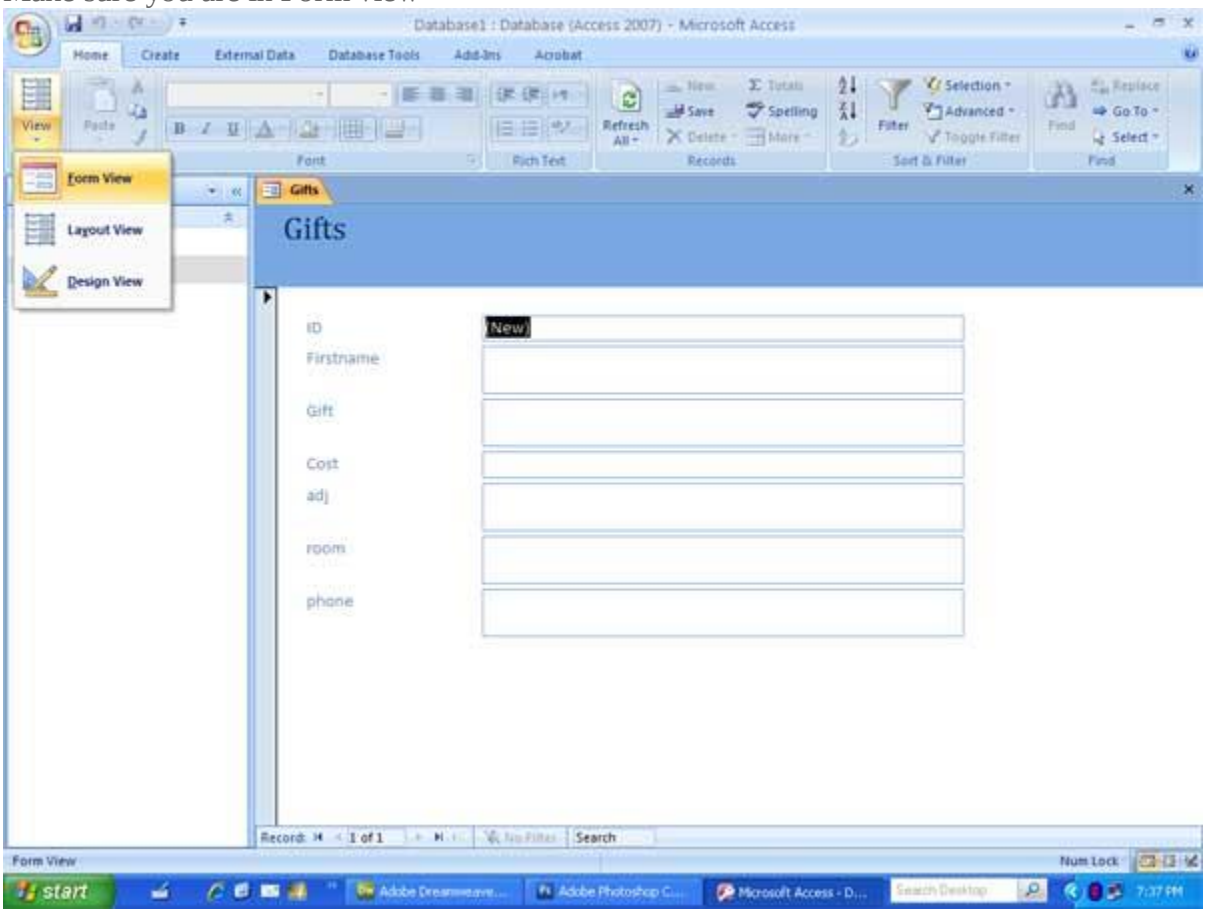
Close the table [x].

Now we need a **form** to populate the table.

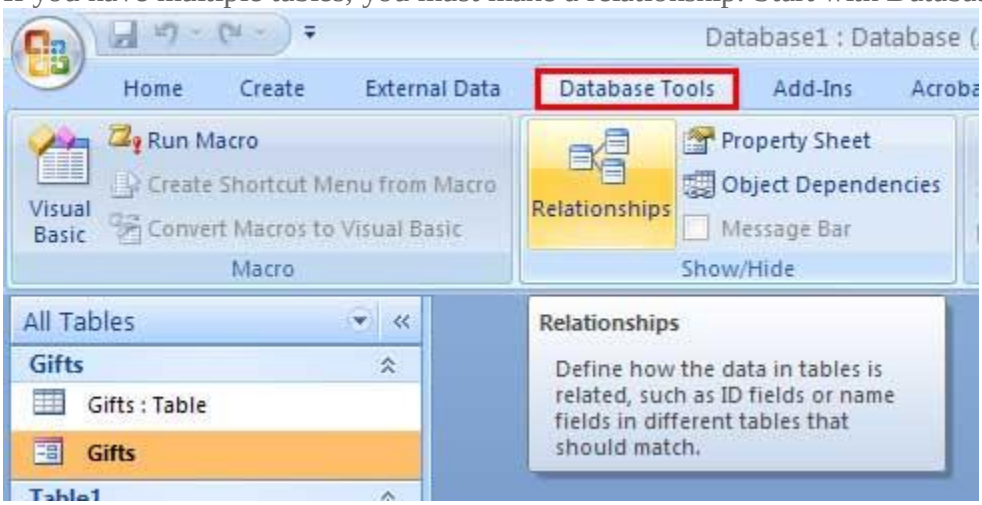




Make sure you are in Form view



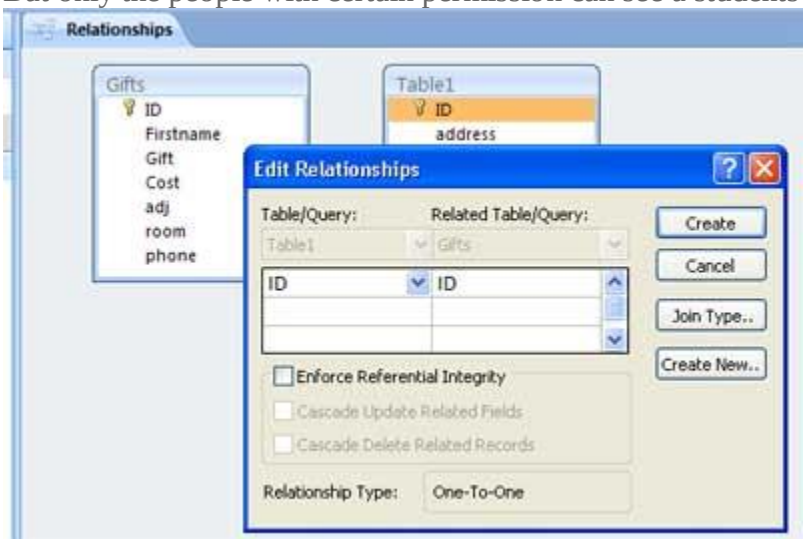
If you have multiple tables, you must make a relationship. Start with Database Tools, and choose Relationships



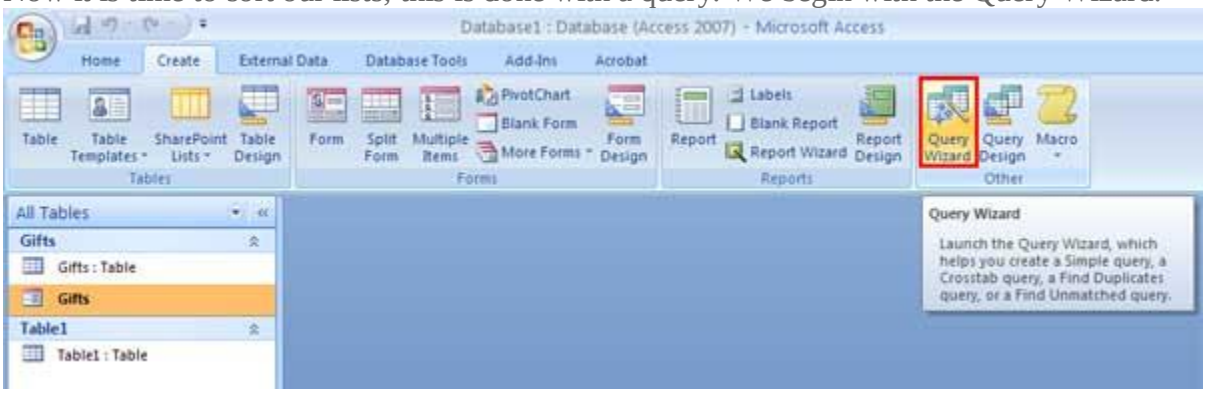
Add your tables, then drag the common field of one, and drop it on the other to create the relationship.

An example would be to have an address table, and a gift table.
If we have a Name field in each one, we can make the relationship, and segregate the date until it needs to be temporarily combined.
The school does this when it sends a bill or grades, the appropriate tables are temporarily joined, and the report is printed.

But only the people with certain permission can see a students grades, or address, or amount owed, etc.

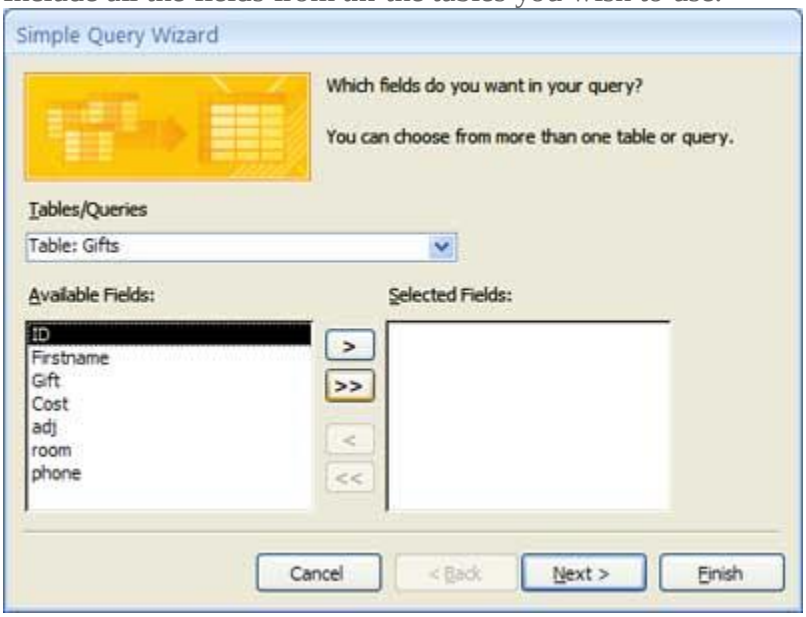


Now it is time to sort our lists, this is done with a query. We begin with the Query Wizard.



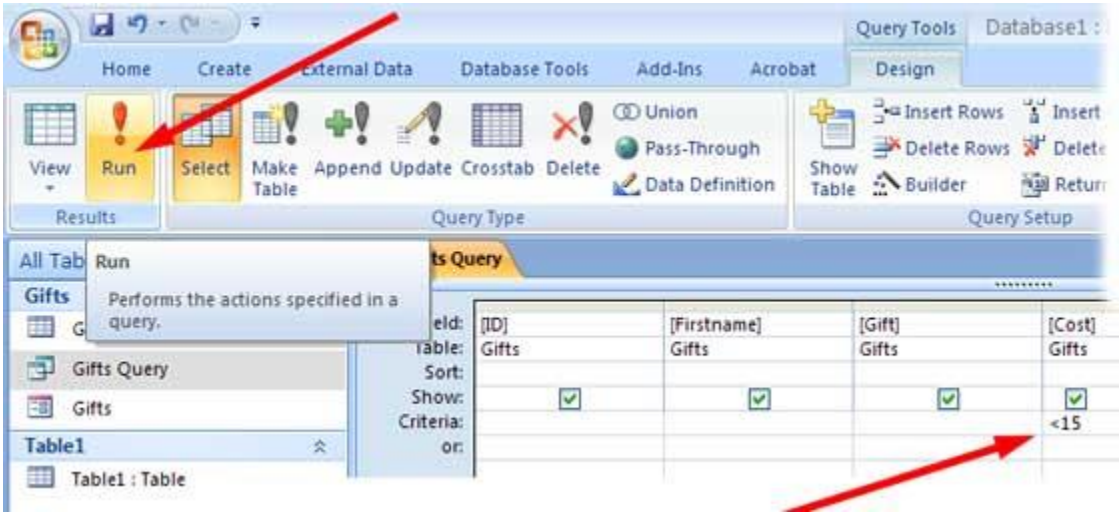
For now, we just need a simple query

Include all the fields from ***all*** the tables you wish to use.



Open the Query, and change to Design View

Now, we change the cost criteria to less than $15, and ! **Run** the query.



Once we have run and saved our queries, we can run **reports** on each query.

LAB 9

After reading over the Sample Access lab, I strongly suggest you watch YouTube video https://youtu.be/2sbgzbuSaNY to see how this is done.

Lab 9 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.

Lab 9 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 Lab 6 (see below... MUST be zipped) (5%)

Preview of quiz question:

Unique identifier	1. Primary Key, the key field
a collection of tables	2. Query
a way to output, or print	3. Form
Input, or view and maintain data	4. Report
Rows in a table	5. Record
easily view the fields and data types	6. Relational Database
Question to process	7. Datasheet View
data is presented in rows and columns	8. Design view

BONUS: 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
<20 will show only records that are less than $20
such as the criteria for State
="TX" will show only records where the State is TX
="N*" will show only records where the State name begins with N, such as NM

A parameter query might be for State
the criteria could be
[Which State are you looking for?]
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