

Business Analytics (SCM 651) Fall 2017, Introduction to Access

We focus on how to use Access together with Excel, and how to prepare data for analysis using Access. We will cover:

1. Importing Excel data into Access.
2. Working with a table in access:
 - Datasheet and design views
 - Use of datasheet view to check if missing values exist
 - Use of a make-table query to compute new variables
 - Use of a make-table query to compute summary statistics
3. Relationships between tables:
 - Different join types
 - Use of join to combine data from two tables
 - Create and export new tables
4. Combining Excel and Access:
 - Select random sample from a data file
 - Splitting a data file into estimation (training) and validation samples
5. Data Cleaning:
 - Identifying and removing duplicates
 - Identifying and removing missing data
 - Identifying mismatched cases

We will first use the following Excel data files posted in the Blackboard folder on Access. Please right click on the file names and save on desk top.

Oj for Access.xls (this file has three worksheets: ojmovement, ojstore, and oj product codes)

- The ojmovement worksheet has data on weekly movements of three brands of orange juice (HH, TROPICANA, MINUTEMAID) at different Dominicks stores.
 - The ojstore worksheet has demographic data for these stores.
 - The oj product codes worksheet has the upc and detailed brand information
- We will use the ojmovement and ojstore worksheets.

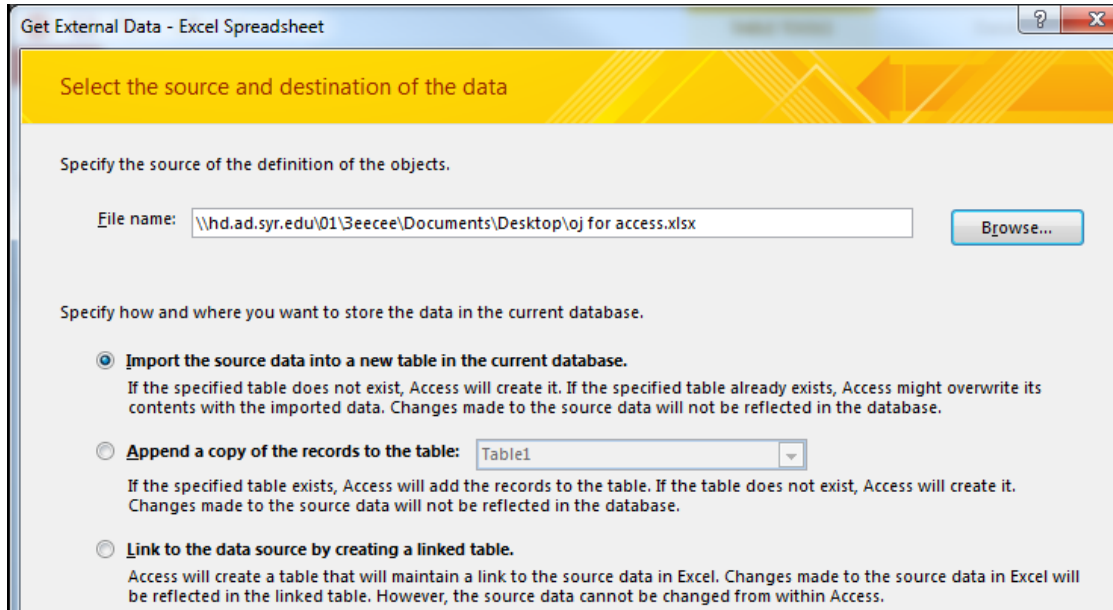
1. Importing Excel Data into Access

Open Microsoft Access by clicking start → all programs → Microsoft Office → Access

Click blank database → create

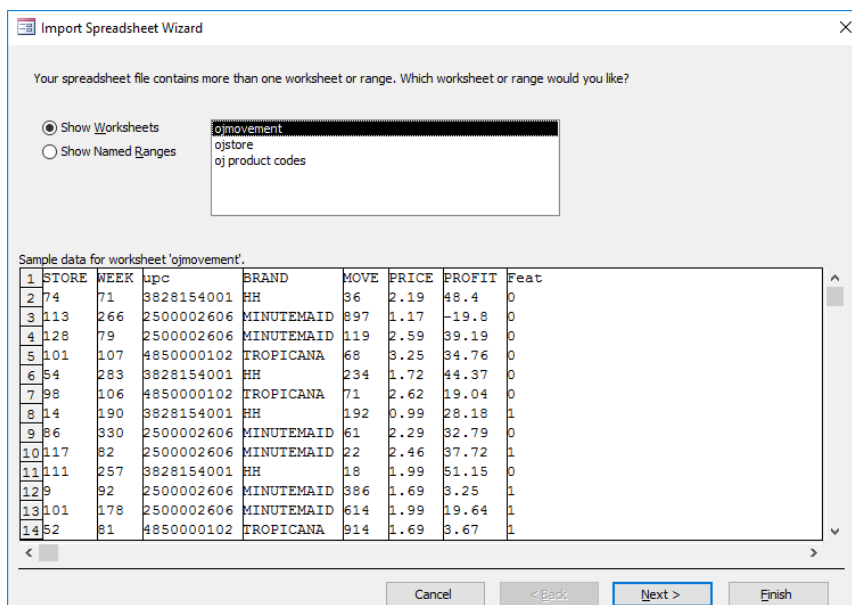
Click external data → Excel (in import & link)

Use browse button to find “oj for access.xls” and click OK



Select the first worksheet you want to import (say, ojmovement) and click Next.

Access will try to detect if the first row has labels in it. Click Next to confirm.



Import Spreadsheet Wizard

Microsoft Access can use your column headings as field names for your table. Does the first row specified contain column headings?

☒ First Row Contains Column Headings

	STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat
1	74	71	3828154001	HH	36	2.19	48.4	0
2	113	266	2500002606	MINUTEMAID	897	1.17	-19.8	0
3	128	79	2500002606	MINUTEMAID	119	2.59	39.19	0
4	101	107	4850000102	TROPICANA	68	3.25	34.76	0
5	54	283	3828154001	HH	234	1.72	44.37	0
6	98	106	4850000102	TROPICANA	71	2.62	19.04	0
7	14	190	3828154001	HH	192	0.99	28.18	1
8	86	330	2500002606	MINUTEMAID	61	2.29	32.79	0
9	117	82	2500002606	MINUTEMAID	22	2.46	37.72	1
10	111	257	3828154001	HH	18	1.99	51.15	0
11	9	92	2500002606	MINUTEMAID	386	1.69	3.25	1
12	101	178	2500002606	MINUTEMAID	614	1.99	19.64	1
13	52	81	4850000102	TROPICANA	914	1.69	3.67	1
14	123	245	2500002606	MINUTEMAID	26	2.45	38.36	1

Cancel < Back Next > Finish

Access next tries to determine the nature of data in the fields (number, text, etc.). Access is good at doing this. However, you can change the data type if you wish to. Click Next to confirm.

Import Spreadsheet Wizard

You can specify information about each of the fields you are importing. Select fields in the area below. You can then modify field information in the 'Field Options' area.

Field Options

Field Name: Data Type: Indexed: ☐ Do not import field (Skip)

	STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat
1	74	71	3828154001	HH	36	2.19	48.4	0
2	113	266	2500002606	MINUTEMAID	897	1.17	-19.8	0
3	128	79	2500002606	MINUTEMAID	119	2.59	39.19	0
4	101	107	4850000102	TROPICANA	68	3.25	34.76	0
5	54	283	3828154001	HH	234	1.72	44.37	0
6	98	106	4850000102	TROPICANA	71	2.62	19.04	0
7	14	190	3828154001	HH	192	0.99	28.18	1
8	86	330	2500002606	MINUTEMAID	61	2.29	32.79	0
9	117	82	2500002606	MINUTEMAID	22	2.46	37.72	1
10	111	257	3828154001	HH	18	1.99	51.15	0
11	9	92	2500002606	MINUTEMAID	386	1.69	3.25	1
12	101	178	2500002606	MINUTEMAID	614	1.99	19.64	1
13	52	81	4850000102	TROPICANA	914	1.69	3.67	1
14	123	245	2500002606	MINUTEMAID	26	2.45	38.36	1

Cancel < Back Next > Finish

- Next, Access asks you to select a primary key or allow Access to add a primary key.
- A **primary key** is a field that identifies each row of data uniquely.
- At this point click on **no primary key**, and click Next.

Import Spreadsheet Wizard

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

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

ID	STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat
1	74	71	3828154001	HH	36	2.19	48.4	0
2	113	266	2500002606	MINUTEMAID	897	1.17	-19.8	0
3	128	79	2500002606	MINUTEMAID	119	2.59	39.19	0
4	101	107	4850000102	TROPICANA	68	3.25	34.76	0
5	54	283	3828154001	HH	234	1.72	44.37	0
6	98	106	4850000102	TROPICANA	71	2.62	19.04	0
7	14	190	3828154001	HH	192	0.99	28.18	1
8	86	330	2500002606	MINUTEMAID	61	2.29	32.79	0
9	117	82	2500002606	MINUTEMAID	22	2.46	37.72	1
10	111	257	3828154001	HH	18	1.99	51.15	0
11	9	92	2500002606	MINUTEMAID	386	1.69	3.25	1
12	101	178	2500002606	MINUTEMAID	614	1.99	19.64	1
13	52	81	4850000102	TROPICANA	914	1.69	3.67	1
14	123	245	2500002606	MINUTEMAID	26	2.45	38.36	1

Cancel < Back Next > Finish

Finally, Access allows you to name the table. The default name is the name of the worksheet in Excel (here, it is ojmovement). Click Finish and click Close in the next screen.

Import Spreadsheet Wizard

That's all the information the wizard needs to import your data.

Import to Table:

☐ I would like a wizard to analyze my table after importing the data.

Cancel < Back Next > Finish

Similarly, import the worksheet **ojstore** and select **STORE** as the **primary key**. Access shows the two tables imported and a table called Table 1 it automatically creates under Tables on the left pane of the window under "All Access Objects." This pane is called the **navigation pane**. Later, when we create **queries**, they will also be listed in the navigation pane.

2(a) Working with one table: Datasheet Views of Table

Doubleclick on the ojmovement table in the left pane (navigation pane). The worksheet opens on the right pane of the window. This is called the **Datasheet view**, which is similar to the worksheet in Excel with additional capabilities for sorting and filtering.

STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat
74	71	3828154001	HH	36	2.19	48.4	0
113	266	2500002606	MINUTEMAID	897	1.17	-19.8	0
128	79	2500002606	MINUTEMAID	119	2.59	39.19	0
101	107	4850000102	TROPICANA	68	3.25	34.76	0
54	283	3828154001	HH	234	1.72	44.37	0

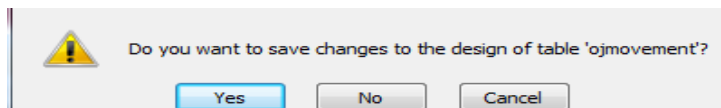
Click on the down arrow next to the column header Store. This allows you to sort or filter the data.

ID	STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat	Click to Add
1					36	2.19	48.4	0	
2				UTEMAID	897	1.17	-19.8	0	
3				UTEMAID	119	2.59	39.19	0	
4				PICANA	68	3.25	34.76	0	
5					234	1.72	44.37	0	
6				PICANA	71	2.62	19.04	0	
7					192	0.99	28.18	1	
8				UTEMAID	61	2.29	32.79	0	
9				UTEMAID	22	2.46	37.72	1	
10					18	1.99	51.15	0	
11				UTEMAID	386	1.69	3.25	1	
12				UTEMAID	614	1.99	19.64	1	
13				PICANA	914	1.69	3.67	1	
14				UTEMAID	26	2.45	38.36	1	
15					50	1.29	23.25	1	

Uncheck select all, select Blanks, and click OK. You get the pane below, which means there is **no** missing data (blanks) for Store.

ID	STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

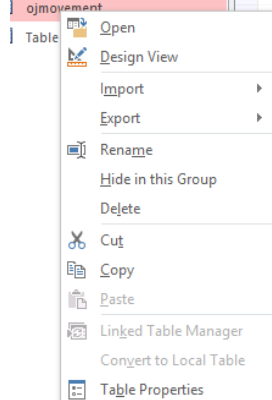
Close the datasheet by clicking on the X at the upper right corner of the right pane and click **No** when Access asks if you wish to save changes to the design of ojmovement.



2(b) Design View of a Table

The design view of a table shows the structure of the data. You can get the design view in three ways:

1. In the navigation pane, right click on the name of the table and select design view by clicking.

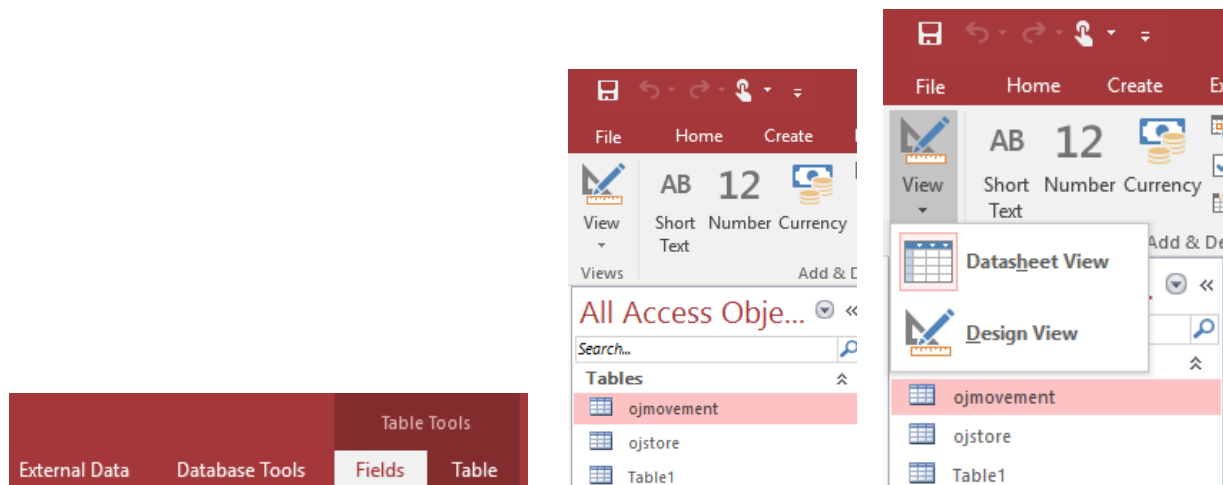


2. Open the table in data sheet view. In the bottom of the right pane, there are two icons



The left icon is for data sheet view, and the right icon is for design view. You can switch views by clicking.

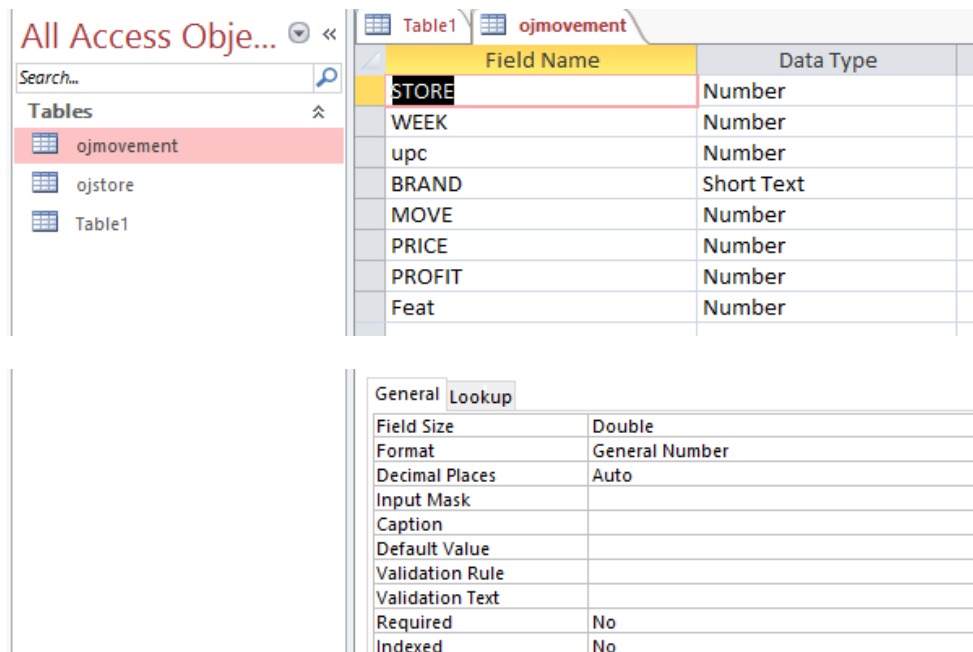
3. Open the table in data sheet view. In the top ribbon, click fields under Table Tools. At the left of the top ribbon, click View ➔ Design View



Switching between Views: In the design view and the data sheet view, at the bottom right of the right pane, you will find the icons for datasheet view and design view. You can switch between the two views by clicking on the icons. (Later on, when we run queries, a third icon called SQL view will appear between the datasheet and design view icons.)



If you use any one of the three methods listed above, you get the design view of the table in the right pane. The design view of ojmovement is shown below.



Field Name	Data Type
STORE	Number
WEEK	Number
upc	Number
BRAND	Short Text
MOVE	Number
PRICE	Number
PROFIT	Number
Feat	Number

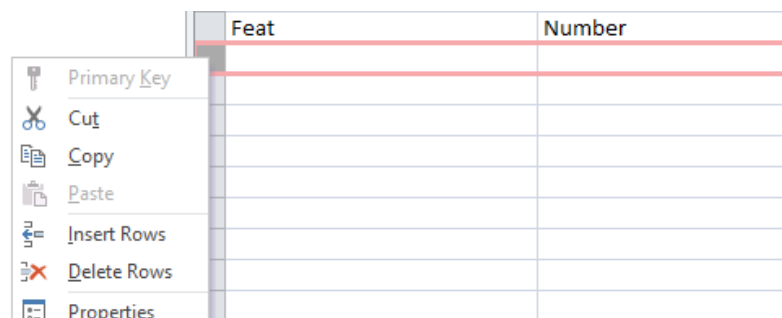
General	
Field Size	Double
Format	General Number
Decimal Places	Auto
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Indexed	No

In the design view, you can:

- Delete a field by right clicking at the left of the field name and clicking “delete rows”
- Change data type of a field
- Create a variable by clicking “Insert rows”

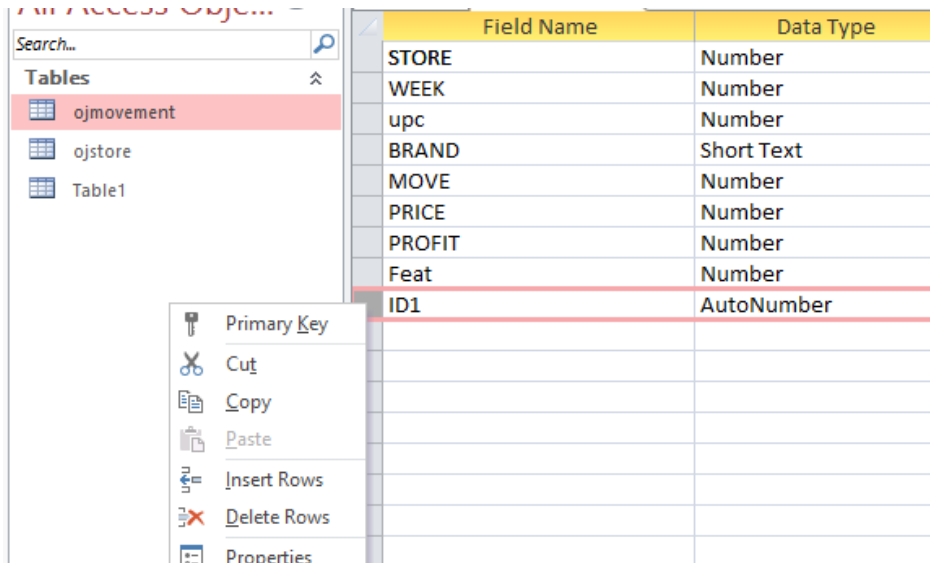
Example:

- Open the table ojmovement in design view.
- Right click at the left of the row below Feat and click Insert Rows.

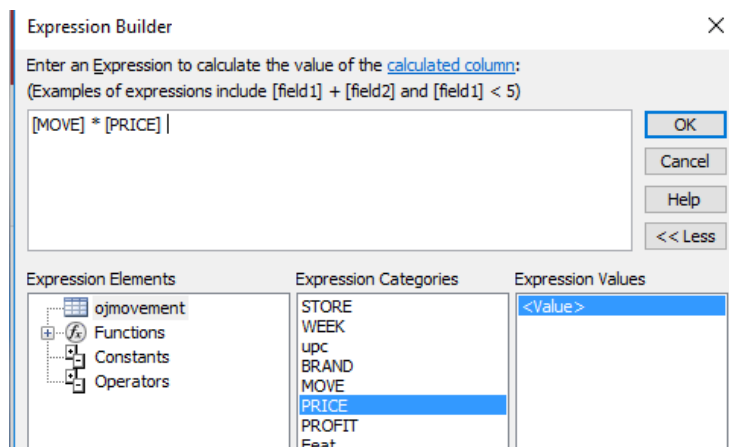


Feat	Number

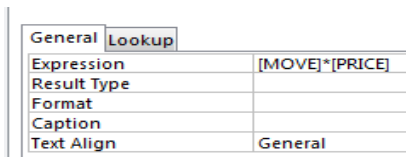
- Type ID1 in “Field Name,” click under “Data Type” and click “AutoNumber.” If you click on the datasheet view icon at the bottom, you can see that ID1 is a column of consecutive integers starting with 1.
- Return to design view and right click at the left of ID1. You can select ID1 as the primary key by clicking on the icon for primary key.



- Right click at the left of the next row, type Revenue in “Field Name”, click under “Data Type” and click “Calculated.” In the Expression Builder box, type [MOVE]*[PRICE], then click OK



In the right column of the lower box, you will see the expression [MOVE]*[PRICE]. You can edit the expression.



Click on X at the top right corner of the right pane and save the changes to the table.

Click on ojmovement and open the table in datasheet view. There are two new fields: ID1 and REVENUE

Table1		ojmovement								
STORE	WEEK	upc	BRAND	MOVE	PRICE	PROFIT	Feat	ID1	REVENUE	
74	71	3828154001	HH	36	2.19	48.4	0	1	78.84	
113	266	2500002606	MINUTEMAID	897	1.17	-19.8	0	2	1049.49	
128	79	2500002606	MINUTEMAID	119	2.59	39.19	0	3	308.21	
101	107	4850000102	TROPICANA	68	3.25	34.76	0	4	221	
54	283	3828154001	HH	234	1.72	44.37	0	5	402.48	

Note on creating new variables calculated from existing variables:

- You can create a new variable from a calculation involving existing variables in two ways: in design view as we did above, or using a Make-table Query (discussed later).
- If you create a new variable using a calculated field, you **cannot** select the variable in a query.
- If you create a new variable using a Make-table Query, you **can** use the variable in a query.

2(c) Working with one table: Use Make Table Query to compute summary statistics

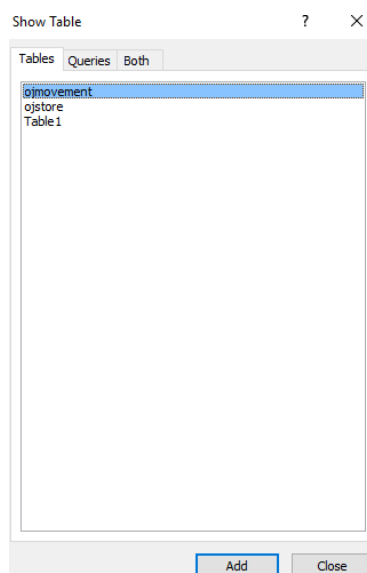
Query: A query is a more powerful version of a filter in Excel. Using queries, we can:

- Compute summary statistics from a table for the whole table or a subset of observations.
- Create new variables. For example, we can compute new variables from variables in a table. Unlike a variable calculated in **Design View**, a variable created in a query (called **Make Table** query) can be used in a query.
- Append tables.
- Combine information from two or more tables.

We first show how to obtain summary statistics from one table using a query.

Click CREATE ➔ Query Design

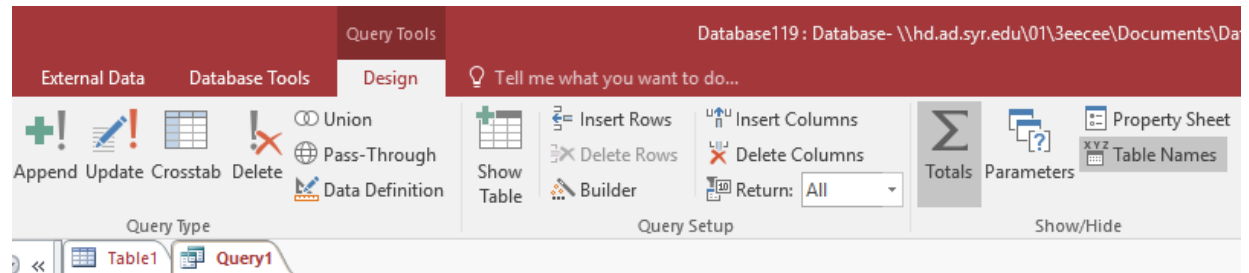
In the **Show Table** pane, click ojmovement ➔ Add ➔ Close



Select STORE, BRAND, Feat, MOVE and PRICE by double clicking

Field:	STORE	BRAND	Feat	MOVE	PRICE
Table:	ojmovement	ojmovement	ojmovement	ojmovement	ojmovement
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

Click on the Σ sign in the top ribbon. “Group By” will appear in the “TOTAL” row in each row in the table at the bottom of the window.



Field:	STORE	BRAND	Feat	MOVE	PRICE
Table:	ojmovement	ojmovement	ojmovement	ojmovement	ojmovement
Total:	Group By	Group By	Group By	Group By	Group By
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

Click next to GROUP BY and select COUNT for STORE, and AVG for PRICE and MOVE.

Field:	STORE	BRAND	Feat	MOVE	PRICE
Table:	ojmovement	ojmovement	ojmovement	ojmovement	ojmovement
Total:	Count	Group By	Group By	Avg	Avg
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

Click **Make Table** in the top ribbon, type in a name for the table (for example, AVG MOVE PRICE), click OK and click the **!** (run) icon in the top ribbon. In the navigation pane, find “AVG MOVE PRICE.”

Open AVG MOVE PRICE in datasheet view. This gives the averages of MOVE and PRICE for each combination of BRAND and Feat, and also the number of cases where a store offered that combination of BRAND and Feat. **Note that you did not select any statistic for BRAND and Feat.**

CountOfSTO	BRAND	Feat	AvgOfMOVE	AvgOfPRICE
5736	HH		0 214.838034441805	1.83714222090259
4578	HH		1 406.300349497597	1.48161642638704
7521	MINUTEMAID		0 79.7666533705624	2.31286796968491
3639	MINUTEMAID		1 219.780983786755	1.95971695520742
7339	TROPICANA		0 96.4027796702548	2.85814552391331
3830	TROPICANA		1 234.209399477807	2.39991122715399
*				

Close the data sheet by click on X at the top right corner of the pane. Close the query by clicking on the X at the top right corner of the pane and save changes to the query. You can name the query. I called it “make AVG MOVE PRICE.” Default names given by Access are query 1, query 2, etc.

2(d) Working with one table: Computing New Variables with Make-Table Query

We now use a make-table query to create a new table that includes the original variables plus new, computed variables. In this new table, we **cannot** include variables calculated using the Design View of the data sheet.

Click CREATE → Query Design

In Show Table, click ojmovement → Add → Close

- Double click to select STORE, WEEK
- In the fields next to WEEK, type
STOREWEEK: $1000 * [\text{STORE}] + [\text{WEEK}]$
REM: $[\text{WEEK}] \bmod 52$
- Double click to select upc, BRAND, MOVE
- In the field next to MOVE, type logmove: $\log([\text{MOVE}])$
- Double click to select PRICE
- In the field next to PRICE, type logprice: $\log([\text{PRICE}])$
- Double click to select PROFIT and Feat
- Next, type ONE: 1

At the bottom of the right pane, there are three icons:



From left to right, these are: data sheet view, SQL view, and design view

Click on data sheet view to preview the table before saving it.

Click Make table, select a name for the table (I selected ojmovement new), click OK, click !

Click Yes

You now have a new table **ojmovement new** that includes the fields you selected and defined.

Close the query by clicking on X at the top right corner of the pane and save changes. (When saving, you can give a name to a query. Later, you can right click on the query and open it in “Design View” and use it or make changes.)

Show Table ? X

Tables Queries Both

ojmovement
ojstore
Table1

Add Close

Field:	STORE	Week	STOREWEEK: 1000*[ST	REM: [WEEK] Mod 52	upc	BRAND	MOVE	logmove: Log([MOV
Table:	ojmovement	ojmovement			ojmovement	ojmovement	ojmovement	
Sort:								
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:								

STOREWEEK: 1000*[STORE]+[WEEK]	REM: [WEEK] Mod 52	upc	BRAND	MOVE	logmove: Log([MOVE])
		ojmovement	ojmovement	ojmovement	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

File Home Create External Data Database Tools Query Tools Design

View Run Select Make Table Append Update Crosstab Delete Union Pass-Through Data Definition

Make Table ? X

Make New Table

Table Name:

☒ Current Database

☐ Another Database:

File Name:

Browse... OK Cancel

The new variables created are:

STOREWEEK = 1000*store + week (For a given brand or upc in this table, this number is unique.)

REM: This is the remainder if you divide week by 52. In the Dominicks data set, the first week starts on September 14, 1989. So, we can define seasons as follows:

Fall: Rem is 0-10, 50, 51 (three month period starting with the last week of August)

Winter: Rem is 11-23 (three month period following Fall), Spring: Rem is 24-36, Summer: Rem is 37-49

ONE: 1 for every row

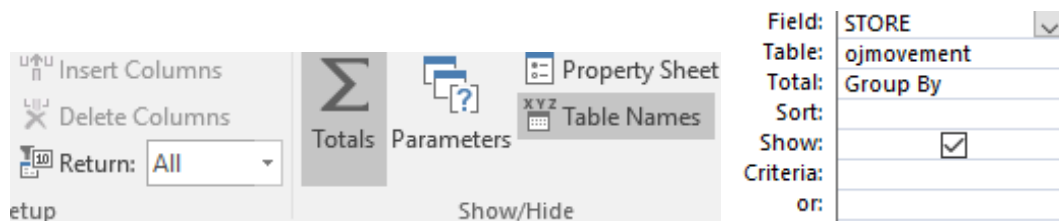
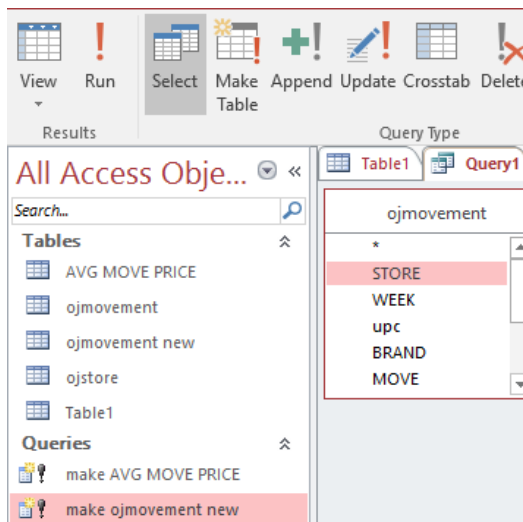
Note: Access is **not** case sensitive when you write names of variables or search for a character string.

2(e) Working with one table: Use Make Table Query to Create table with a subset of variables

We now create a table that includes the field [STORE] only from the table ojmovement.

- Click CREATE → Query Design
- In Show Table, click ojmovement → Add → Close
- Double click to select STORE
- Click Make table, enter name STORE ONLY, click ok
- If you run the query now using !, there will be many duplications as the same store appears for many combinations of BRAND and WEEK.
- To remove duplicates, click on the Σ sign in the top ribbon. In the data sheet view, you will now only see 82 unique cases of STORE.
- Click !

You will get a table called STORE ONLY with 82 cases from the original table of 33,643 cases.



2(f) Working with one table: Applying Filters with Make-Table Query

We now create a subset of the data in the table **ojmovement new** using the following filters:

- Brand name is HH
- Season is winter (remainder is 11:23)
- Price is between 1.25 and 1.5

Click CREATE → Query Design → ojmovement new → add → close

Select all variables by double clicking one at a time

In the row called “criteria” type:

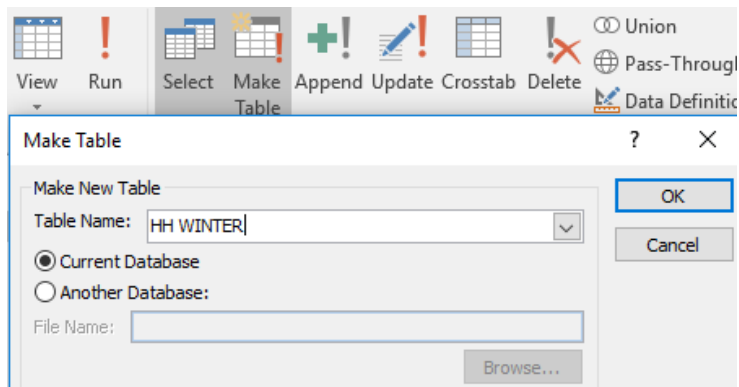
>=11 and <= 23 under REM

= “HH” under BRAND

>=1.25 and <=1.5 under PRICE

Click Make Table, enter name of table (I used HH WINTER), click !

The table HH WINTER contains 327 observations for HH during Winter when price is between 1.25 and 1.5.



STOREWEEK	REM	upc	BRAND	MOVE	logmove	PRICE	logprice
ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	>=11 And <=23		"HH"			>=1.25 And <=1.5	

The following can also be used in the Criteria field:

- HH **instead of** "HH."
- Between 11 and 23 **instead of** >=11 and <=23.
- Between 1.25 and 1.5 **instead of** >=1.25 and <=1.5.

2(g) Note on Filters

Filtering with Multiple Conditions

- We can have a query where multiple conditions must be satisfied simultaneously, that is, an **AND** condition is satisfied. We used that in the previous example where the three conditions Season is Winter, BRAND = HH, and Price is between 1.25 and 1.5 were satisfied simultaneously.
- We can also have an OR condition within one field. For example, we may want to apply the filter: price is between 1.25 and 1.5, or price is between 3.0 and 3.5.

We can enter these two criteria (between 1.25 and 1.5, and between 3 and 3.5) in multiple lines.

STOREWEEK	REM	BRAND	MOVE	logmove	PRICE
ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
					>=1.25 And <=1.5
					>=3 And <=3.5

Suppose we want to **also satisfy** the criterion that the season is winter. We need to enter this condition in both lines (corresponding to price between 1.25 and 1.5, and price between 3.0 and 3.5).

STOREWEEK	REM	BRAND	MOVE	logmove	PRICE
ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
	>=11 And <=23				>=1.25 And <=1.5
	>=11 And <=23				>=3 And <=3.5

Some Filtering Criteria

Like “*XX*” in a field selects all cases that include the character string XX. This criterion can be used in the Criteria field. It can also be used to create a new variable using a Make Table query.

Example:

- The criterion LIKE “*TROP*” in the field BRAND will select all cases for TROPICANA.
- The criterion LIKE “*T*” in the field BRAND will select both TROPICANA and MINUTE MAID

Is Null in a field selects all cases where the value is missing (null)

Is Not Null in a field selects all cases where the value is not missing

Not Is Null can also be used instead of **Is Not Null**.

BETWEEN 11 and 23 in a field will choose all cases where the variable is ≥ 11 and ≤ 23 .

Creating New Variables with Switch Function

The Switch function can be used to create new variables in two ways:

1. Switch(condition 1, value 1, condition 2, value 2, ..., condition n, value n)

To use this form, you provide a complete list of conditions and the value to be assigned to each condition. If condition 1 is satisfied, the variable is assigned value 1, and so on.

Example: Switch([REM] between 11 and 23, "Winter", [REM] <11 or [REM] > 23, "Other")

2. Switch(condition 1, value 1, condition 2, value 2, ..., condition n, value n, true, "Other")

In this case, if none of the n conditions is satisfied, the variable will be assigned the value "Other."

Example: Switch([REM] between 11 and 23, "Winter", [REM] between 24 and 36, "Spring", [REM] between 37 and 49, "Summer", true, "Fall")

Note on missing data: In the ojmovement data set, there are no missing data. In general, you need to allow for missing data when you create a new variable. In Access, "" (that is, two double quotation marks with nothing in between) means a blank cell. So, if you want the SEASON to be blank if [REM] is missing, use:

Switch([REM] Is Null, "", [REM] between 11 and 23, "Winter", [REM] between 24 and 36, "Spring", [REM] between 37 and 49, "Summer", true, "Fall")

Creating New Variables using IIF Function

The IIF function can also be used to create new variables. The function IIF function is similar to IF in Excel but more versatile. The form of the function is

IIF(condition to be satisfied, value if true, value if false)

Example:

IIF([BRAND] = "HH", 1, 0) will return 1 if BRAND is HH, and 0 if BRAND is not HH.

The value if false can again be an IIF function.

Example 1: IIF([BRAND]="HH", 1, IIF([BRAND] LIKE "*TROP*", 2, 3)) returns 1 if HH, 2 if Tropicana, 3 if Minute Maid.

Example 2: We can create a new variable SEASON (Winter if REM is between 11 and 23, Spring if REM is between 24 and 36, Summer if REM is between 37 and 49, otherwise FALL) as follows:

SEASON: IIF([REM] > 10 AND [REM] < 24, "WINTER", IIF([REM] > 23 AND [REM] < 37, "SPRING", IIF([REM] > 36 AND [REM] < 50, "SUMMER", "FALL")))

The following will also create SEASON:

SEASON: IIF([REM] between 11 and 23,"WINTER",IIF([REM] between 24 and 36,"SPRING",IIF([REM] between 37 and 49,"SUMMER","FALL")))

Wildcard Characters used with Like

Character	Description	Examples
*	Matches any number of characters which can be numbers or alphabetical letters.	<ul style="list-style-type: none"> Like <code>"*App*"</code> finds any string that includes App Like <code>"*Ap"</code> finds any string that ends with Ap Like <code>"Ap*"</code> finds any string that begins with Ap
?	Matches a single character at a given location.	Like <code>"b?ll"</code> finds bull, bill, bell and ball
[]	Matches characters within the brackets	Like <code>"b[ue]ll"</code> finds bull and bell, but not ball and bill.
!	Excludes characters within the brackets.	Like <code>"b[!ue]ll"</code> finds ball and bill, but not bull and bell
-	Matches a range of characters, presented in ascending order (A to Z, not Z to A)	Like <code>"a[b-d]e"</code> finds abe, ace and ade
#	Matches any single numeric character	<code>2#4</code> finds 204, 214 and 224

Example of use of Like:

Suppose we collected data in Qualtrics with the question:

Which brands of laptop have you ever owned? (Check all that apply)

☐ None ☐ Apple/Mac ☐ Dell ☐ HP ☐ Other (specify)

In the dataset generated by Qualtrics, the items selected will be listed in a single column, and the cell in the column will be blank if there is no answer. For example, your dataset may look like:

Laptop
Apple
Dell, HP
Dell
Apple, HP
None

From this data table, you can create a dummy variable Apple which is 1 if Apple is selected and 0 if not as follows:

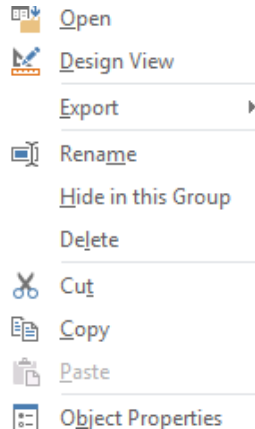
Apple: `switch([laptop] like "*App*",1,[laptop] is Null, "",true,0)`

The cell will be blank if the variable is missing in the data set (blank cell)

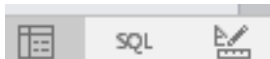
2(h) Note on Queries

We have so far used the Make Table Query. Later we will also use the Query Wizard to find duplicates in a table and unmatched cases between two tables. When you use the Query Wizard, the query is saved automatically. If you use the Make Table query, you have the choice to save the query or not.

- The saved queries are listed in the navigation pane (left pane).
- If you right click on the name of a query in the list, you get a menu that allows you, for example, to rename the query.

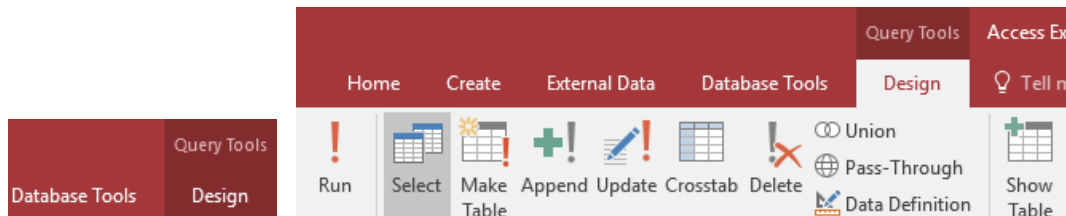


- If you right click Design View in the menu above, you get the design view that allows you to make changes, make new tables, etc. In the design view or the datasheet view, the bottom right of the pane gives you the following:



From left to right, the three icons mean Datasheet View, SQL View and the Design View of the query.

- When we run any of these queries, we make a table.
- Sometimes, in design view, you do not see the icons for make table and run (!). Then, in top ribbon, click Design under Query Tools to get the menu.



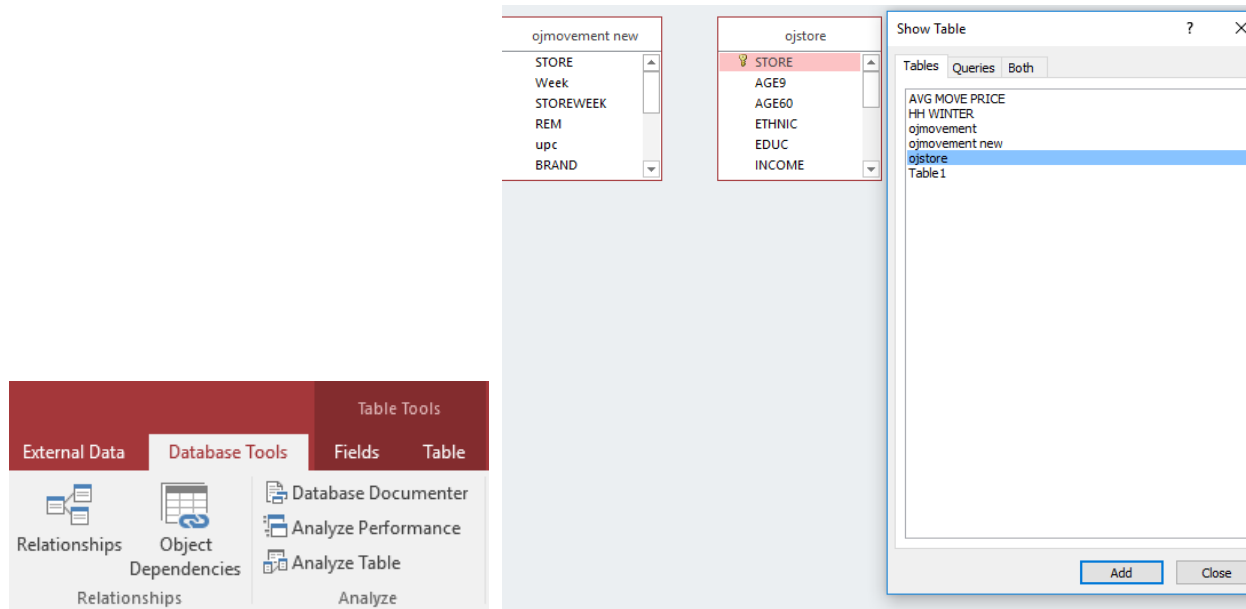
- By clicking on the Datasheet View icon, you can preview the table before making and saving it.
- **Removing duplicates:** Suppose you have used Create → Query Design and selected one or more variables from a table. If you click \sum then you create a table of unique combinations of the variables selected. You can click the data sheet view to see the unique combinations without duplicates. You can use Make Table → Name Table → OK → ! to save the table of unique values with duplicates removed.

3. Relationship between Tables

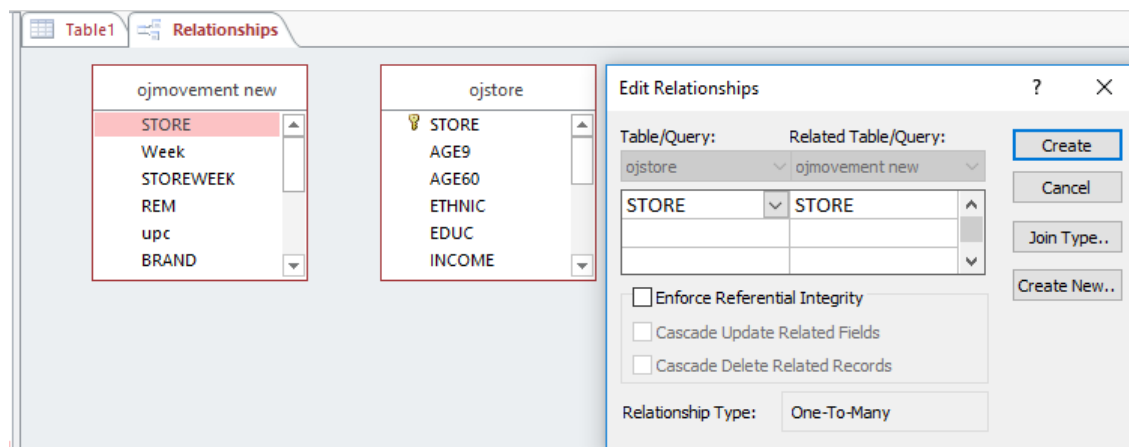
If two tables have a field in common, we can join the tables. The field may have different names in the two tables and can be character variables as well as numbers. We will first join the tables ojmovement new and oystore using the field STORE.

Click Database Tools → Relationships

In Show Table, click ojmovement new → Add → oystore → Add → Close

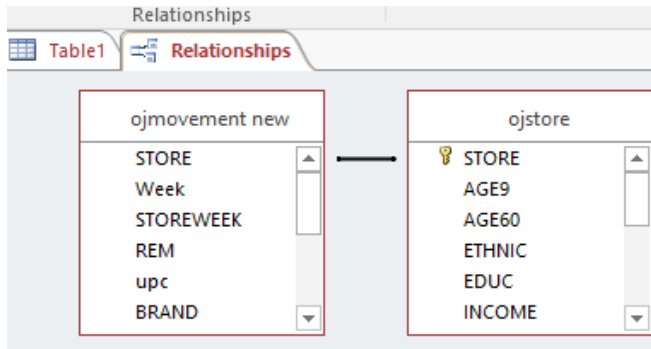


Click on STORE in oystore and drag to STORE in ojmovement new.

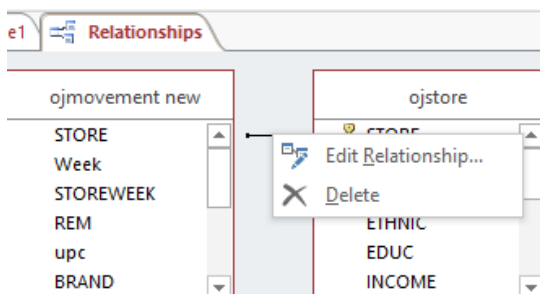


Default join type: The default join type will keep only cases where the value of STORE is same in both tables.

Click **Create**. This establishes the relationship between the two tables.

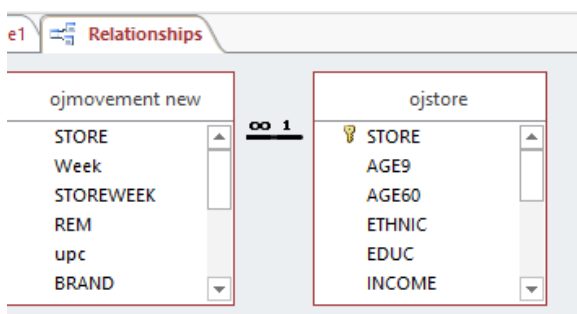


Right click on the line linking the tables. This allows you to edit or delete the relationship.



Referential Integrity: If each value of STORE in one table has at least one case with the same value of STORE in the other table, we have **Referential Integrity**. To check if that is true here, click on “Enforce Referential Integrity,” then click **Create**.

Note: We get a 1 to infinity relationship as for every store in ojstore, there are many cases in ojmovement new.



Close the right pane and save the relationship.

4. Query with Joined Tables

We will now join the tables ojmovement new and ojstore to create the following tables:

- Table with STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat, Age9, Age60, INCOME, EDUC (we will call it OJ MOVEMENT WITH STORE)
- Table with STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat, Age9, Age60, INCOME, EDUC for HH only (HH WITH STORE)
- Table with STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat, Age9, Age60, INCOME, EDUC for Tropicana only (TROPICANA WITH STORE)
- Table with STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat, Age9, Age60, INCOME, EDUC for Minute Maid only (MINUTEMAID WITH STORE)

Click CREATE → Query Design

In Show Table, click ojmovement new → Add → ojstore → Add → Close

- Note that the relationship is shown in the pane.
- You can edit or delete the relationship by taking the cursor to touch the joining line and right clicking the mouse.
- Any change in relationship you make here will apply to the query only.

By double clicking select the following:

- From ojmovement new: STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat
- From ojstore: AGE9, AGE60, EDUC, INCOME

Click Make Table → Type OJ MOVEMENT WITH STORE in the name box → Click OK → !

A table called OJ MOVEMENT WITH STORE is created.

Now type HH in the criteria under BRAND, click Make Table, change name of table to HH WITH STORE, click OK → !

The screenshot shows the 'Make Table' dialog box with the following details:

- Table Name:** HH WITH STORE
- Current Database:** Selected
- File Name:** (Empty)

Below the dialog box is a query design grid with the following data:

Field:	STORE	Week	STOREWEEK	upc	BRAND	MOVE
Table:	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new	ojmovement new
Sort:						
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					"HH"	

- This creates the table HH WITH STORE.
- Similarly create TROPICANA WITH STORE and MINUTEMAID WITH STORE.
- Note that in each of these three brand-specific tables, STOREWEEK uniquely identifies each row.

Close the query.

Application of Joining Tables: Creating Table of Same Store Sales

We will now create a table that contains same store sales for HH and TROPICANA.

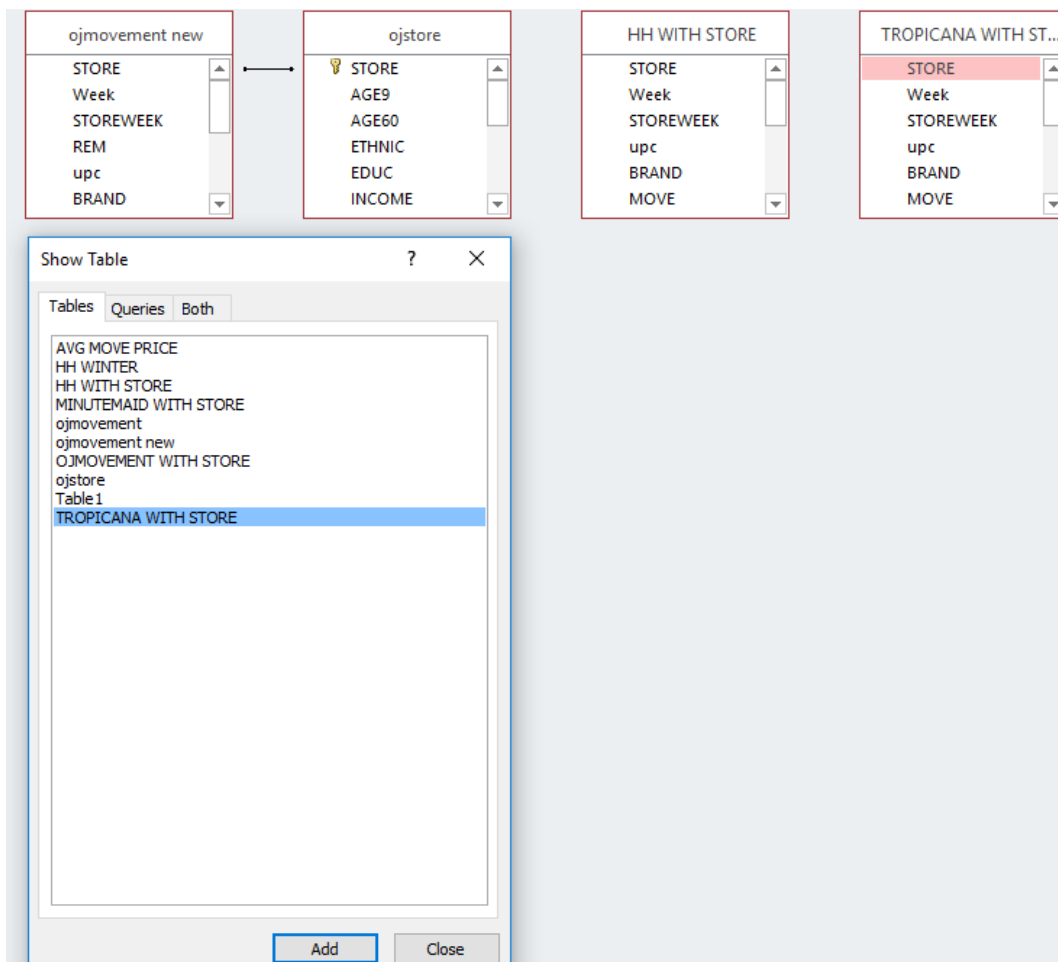
Step 1: Establish relationship between HH WITH STORE and TROPICANA WITH STORE

We use STOREWEEK, which is unique to each observation in either table, to join the tables.

Click Database Tools ➔ Relationships

The tables ojmovement new and ojstore with their relationship will appear.

Click Show Table. Click HH WITH STORE ➔ Add ➔ TROPICANA WITH STORE ➔ Add ➔ Close



Drag STOREWEEK from HH WITH STORE to STOREWEEK in TROPICANA with STORE. Click Create.

Close the pane and save relationships.

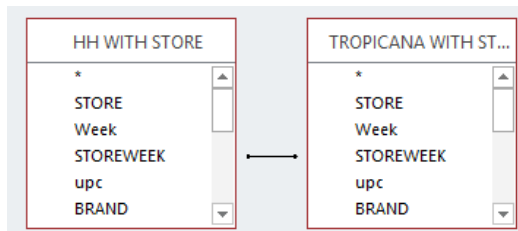
Step 2. Creating Table of Same Store Sales

We will now create a table called **HH TROPICANA SAME STORE** that includes:

STORE, WEEK, BRAND (HH), MOVE (HH), logmove (HH), PRICE (HH), logprice (HH),Feat (HH), BRAND (TROPICANA), logmove (TROPICANA), PRICE (TROPICANA), logrpice (TROPICANA), Feat (TROPICANA), AGE9, AGE60, INCOME, EDUC

Click CREATE → Query Design → HH WITH STORE → Add → TROPICANA WITH STORE → Add → Close

The design view shows the link between the two tables you created in Relationships. You can right click on the link and edit it here. The change will only be saved for the query and will not affect the relationships defined before.



Double click to select:

From HH WITH STORE: STORE, WEEK, BRAND, MOVE, logmove, PRICE, Feat

From TROPICANA WITH STORE: BRAND, MOVE, logmove, PRICE, logprice, Feat, AGE9, AGE60, EDUC, INCOME

Click Make Table, type in name HH TROPICANA SAME STORE, click OK → !

Note that the number of cases in this table is only 4022.

Close the query. Right click on the name HH TROPICANA SAME STORE in the left pane. Export to Excel.

You can use this data set to estimate the cross price elasticity of demand of either brand with respect to the price of the other brand.

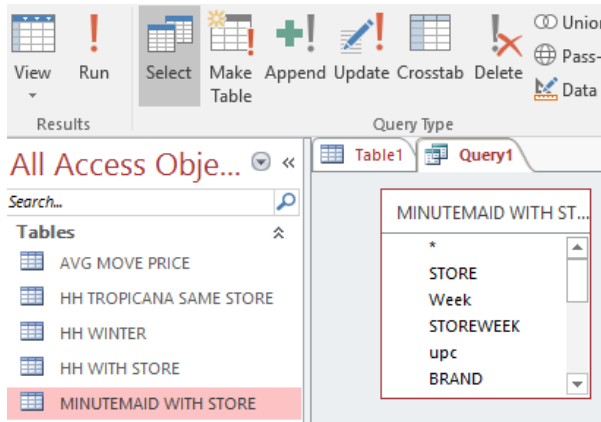
If you run a regression with logmove of HH as the dependent variables, and the logprices of HH and Tropicana as the independent variables, the coefficient of logprice of HH is the own price elasticity of demand, and the coefficient of logprice of Tropicana is the cross price elasticity of demand.

5. Appending Tables

If two tables have the same fields, we can append them to make a single table.

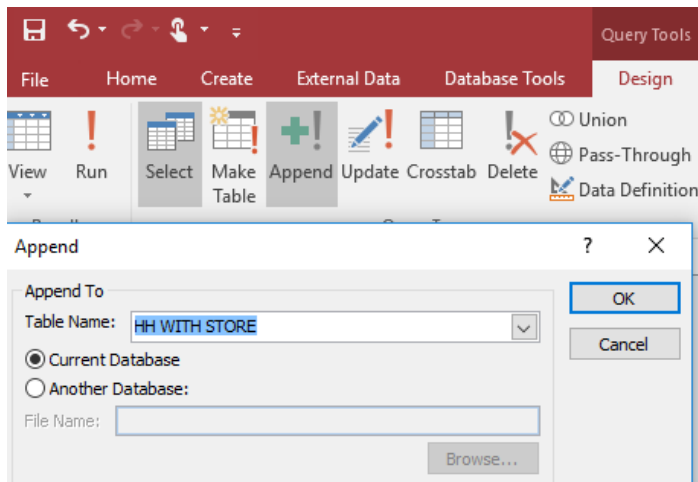
Example: We will append the table **MINUTEMAID WITH STORE** to the table **HH WITH STORE**.

Click **CREATE** → **Query Design** → **MINUTEMAID WITH STORE** → **Add** → **Close**



In the box **MINUTEMAID WITH STORE**, double click on ***** at the top of the list of variables.

In the top ribbon, click **+**!



Using the drop down arrow, select **HH WITH STORE**

click **OK** → !

The table **HH WITH STORE** now contains data for both **HH** and **Minute Maid**.

Close and save the query.

Right click on **HH WITH STORE** and rename to **HH MINUTEMAID WITH STORE**.

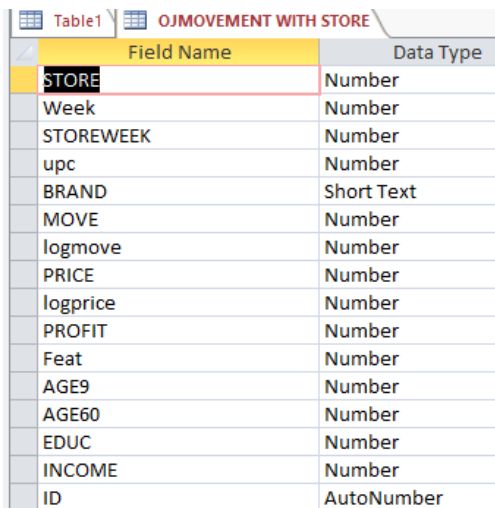
6. Selecting a Random Sample

We will combine Excel and Access to select a random sample from the table **OJ MOVEMENT WITH STORE**. This datasheet does not have any missing values (blank cells). If your data set has missing values, you can remove missing values by using a **make-table query** and using **Is Not Null** as criteria.

In this example, we will split the table into two tables: a random sample of 1000 cases, and the remaining cases.

Step 1. Adding an ID to each case of OJ MOVEMENT WITH STORE

- Open OJMOVEMENT WITH STORE in design view. (For example, right click on the name of the table in the navigation pane and click Design View)
- Right click at the left of the row below INCOME, click insert rows, type in ID and select AutoNumber.



Field Name	Data Type
STORE	Number
Week	Number
STOREWEEK	Number
upc	Number
BRAND	Short Text
MOVE	Number
logmove	Number
PRICE	Number
logprice	Number
PROFIT	Number
Feat	Number
AGE9	Number
AGE60	Number
EDUC	Number
INCOME	Number
ID	AutoNumber

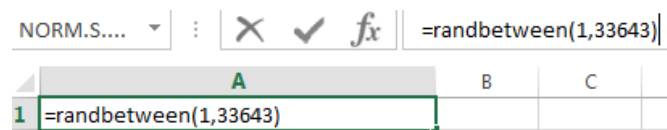
- Save the change.

The ID runs consecutively from 1 to 33,643 and uniquely identifies each row of the data. So, ID is a primary key for the table.

Step 2. Generate a random sample using Excel.

Open Excel.

In cell A1, type = **randbetween(1,33643)** and hit enter



Mark cell A1, click control C, and paste in cells A2:A1500. (Note that A1 has changed)

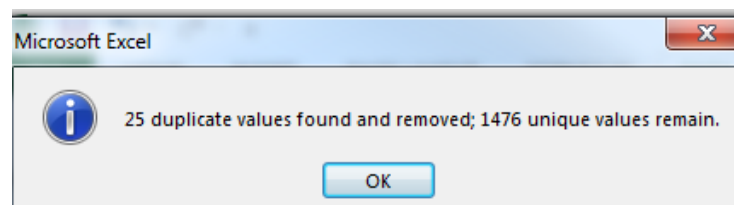
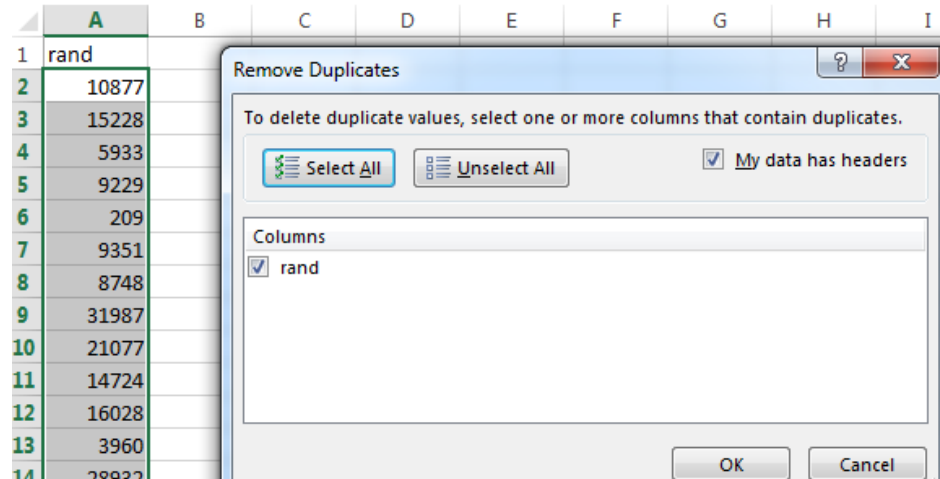
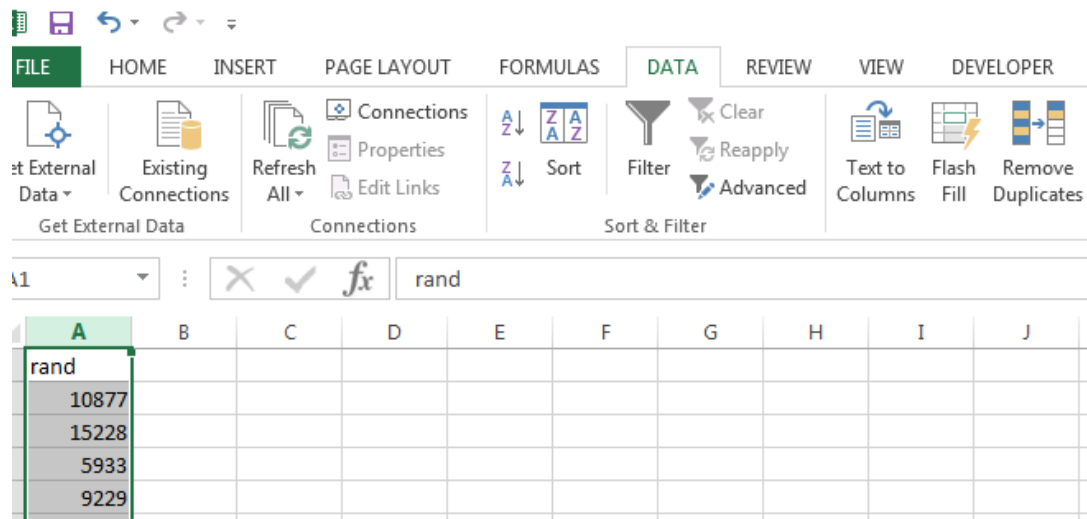
You have now created 1500 random numbers from {1,...,33643}. Some of these are duplicates.

Copy A1:A1500 and paste in B2:B1501 using **paste values**.

Type rand in cell B1. Delete column A. Column A now has rand as the header, and random numbers in cells A2:A1502

Removing duplicates: Select column A by clicking A (at the top of the column).

Click Data → Remove Duplicates → OK



In the dialog box, click OK.

Delete rows below A1001. You now have 1000 unique random numbers between 1 and 33,643.

In cell B1, type in header flag.

Type 1 in cell B2, and copy and paste in cells B3:B1001. You can do it as follows:

- Mark B2, hold control and type C
- Take cursor to the right bottom corner of B2 until a + sign appears
- Double click on the + sign. This populates the column with 1 up to B1001.

	A	B	C	D
1	rand	flag		
2	7797	1		
3	17214			

You now have two columns: 1000 random numbers in cells A2:A1001 (header rand), and 1000 entries of 1 in cells B2:B1001 (header flag).

Rename sheet 1 to an appropriate name (I used **rand**) and save the Excel file (I used **rand**).

Step 3. Combining to select random sample

- Import the Excel file rand into Access. Choose rand as the primary key.
- You can now divide OJ MOVEMENT WITH STORE into an estimation sample and a validation sample in two ways.

Method 1 (using SQL):

- Click Create ➔ Query Design ➔ OJ MOVEMENT WITH STORE ➔ Add ➔ Close
- Double click to select all the variables you want to include in the table. Make sure ID is one of the variables.
- In the criteria box under ID, type In (select rand from rand)

Field:	EDUC	INCOME	ID
Table:	OJMOVEMENT WITH S	OJMOVEMENT WITH S	OJMOVEMENT WITH STORE
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			In (select rand from rand)

- The data sheet view shows the random sample drawn from OJMOVEMENT WITH STORE. Click Make Table and save the table. This is the randomly drawn estimation sample of 1000 cases.
- Change criteria under ID to Not In (select rand from rand)

Field:	EDUC	INCOME	ID
Table:	OJMOVEMENT WITH S	OJMOVEMENT WITH S	OJMOVEMENT WITH STORE
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Not In (select rand from rand)

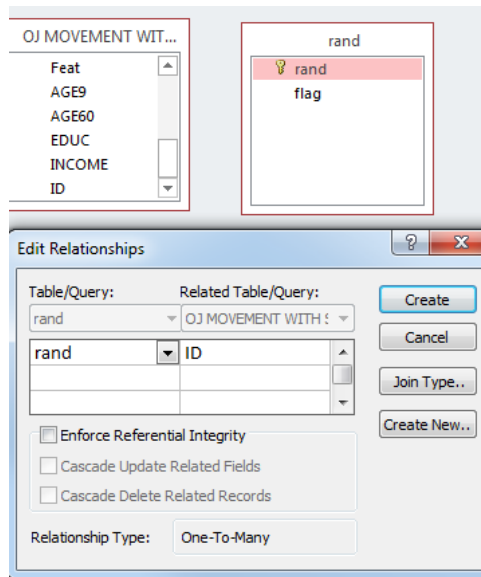
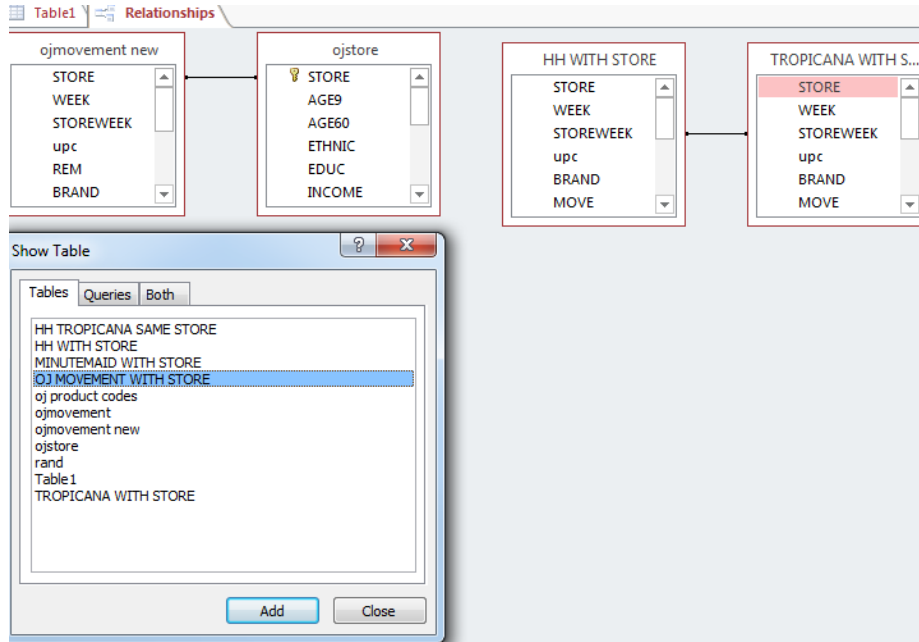
This gives a table of the other 32643 cases not in the random sample. Save it as validation sample.

Caution: It takes a long time to draw the validation sample this way. Method 2 is much quicker.

Method 2 (not using SQL):

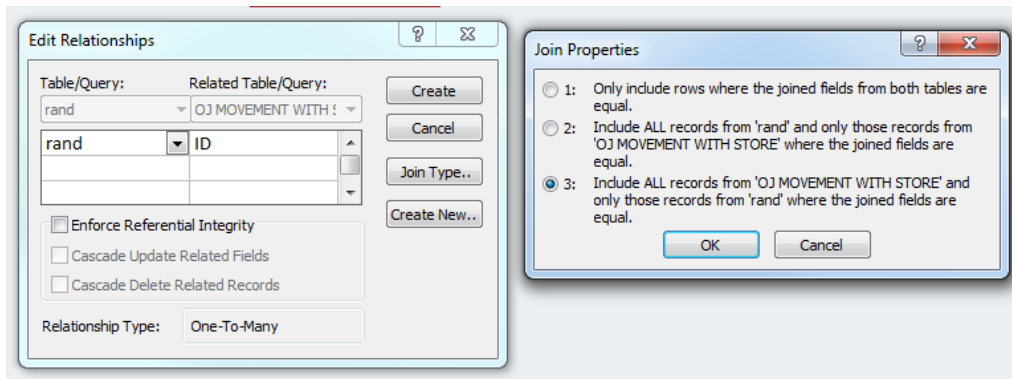
Click Database Tools ➔ Relationships ➔ Show Table

In Show Table, click OJ MOVEMENT WITH STORE ➔ Add ➔ rand ➔ Add ➔ Close



Drag rand in Table rand to ID in Table OJ MOVEMENT WITH STORE

Click Join Type ➔ include ALL records from 'OJ MOVEMENT WITH STORE' and only those records from 'rand' where the joined fields are equal ➔ OK ➔ Create



Save the changes in relationships.

Click CREATE → Query Design → OJ Movement with Store → Add → rand → Add → close

Double click to select

From OJ MOVEMENT WITH STORE: STORE, WEEK, STOREWEEK, upc, BRAND, MOVE, logmove, PRICE, logprice, PROFIT, Feat, AGE9, AGE60, EDUC, INCOME, ID

From rand: Flag

Click make table and save table as OJ MOVEMENT FLAG

Creating Table for random sample:

Under Flag, in the Criteria row, type **Is Not Null**

Field:	INCOME	ID	flag
Table:	OJ MOVEMENT WITH	OJ MOVEMENT WITH	rand
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Is Not Null
or:			

Click Make Table, enter name **random sample**, click **ok** → !

You now have a table called **random sample** that has 1000 cases randomly selected from OJ MOVEMENT WITH STORE.

Creating Remaining Table: Change the entry for criteria to **Is Null**.

Click **make-table**, enter name **remaining table**, and click !

You now have two new tables: random sample (1000 cases) and remaining table (32,643 cases). You can delete the field Flag from both tables.

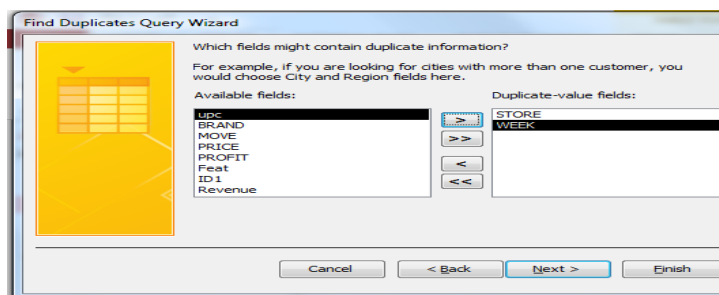
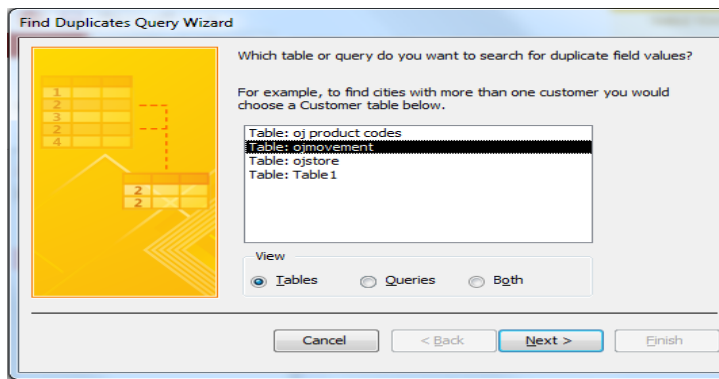
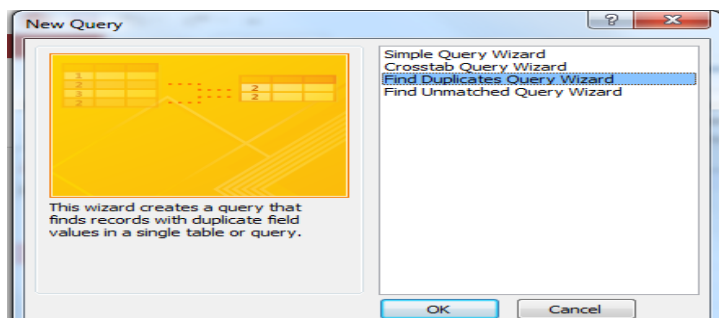
7. Data Cleaning using Access: Finding and Removing Duplicates

7(a) Using Query Wizard to Check if Duplicates Exist

We will check if the combination of STORE and WEEK is repeated in the table OJMOVEMENT.

- Click CREATE in the top ribbon
- Click on any table in the left pane
- Double click on Query Wizard, then click Find Duplicates Query Wizard → OK
- Click Table: ojmovement → Next
- Select Store and Week and click Next

You will get the data sheet view of a table that gives the STORE, WEEK and BRAND for all the cases where the combination of STORE and WEEK is duplicated. You can click on the design view icon at the bottom right and use make table query to save this file.



Note: If you select STORE, WEEK and BRAND, there are no duplicates.

7(b) Removing Duplicates

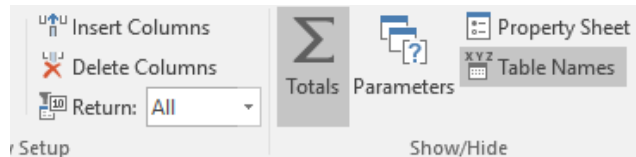
We can have two distinct situations:

- (1) We only want to remove duplicates wherever they occur. Then, we can simply instruct Access to keep the first case where a given field or combination of fields occurs.
- (2) We want to go through the table of duplicates prepared using the Find Duplicates Query Wizard and select the cases we want to remove.

Case 1. We have a table with one field and want to remove duplicates from that one field.

Example: Suppose we want to start with OJMOVEMENT, keep only the field STORE, and remove all duplicates for STORE.

- Click Create → Query Design → OJMOVEMENT → Add → Close
- Double click to select STORE
- In the data sheet view, you will have 33,643 cases, which is same as the number of cases in OJMOVEMENT.
- Click on Σ in the top ribbon.

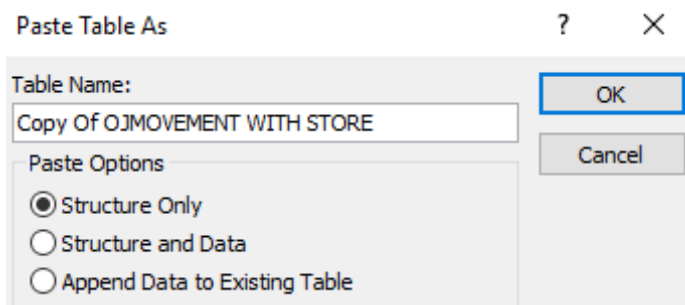


- Now, in the data sheet view, you will have only 82 cases, which give the unique values of STORE.
- You can save this table using “make table “ from design view. In this table, the duplicates for STORE are removed.

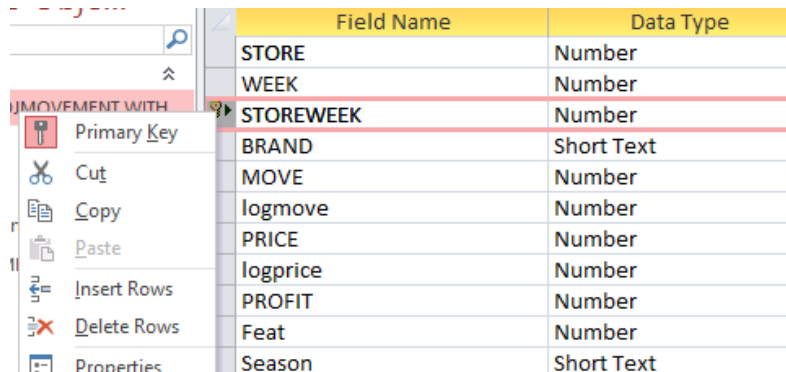
Case 2. We have a table with multiple fields, and we want to remove duplicates based on one field.

Example: Suppose we want to start with OJMOVEMENT WITH STORE, and remove duplicates for one field: STOREWEEK.

- In the list of tables in the left pane, right click OJMOVEMENT WITH STORE → copy → paste
- Paste to copy of OJMOVEMENT WITH STORE and select “structure only.”



- Open copy of OJMOVEMENT WITH STORE with design view. Right click at the left of STOREWEEK and select primary key. Save the change in design.



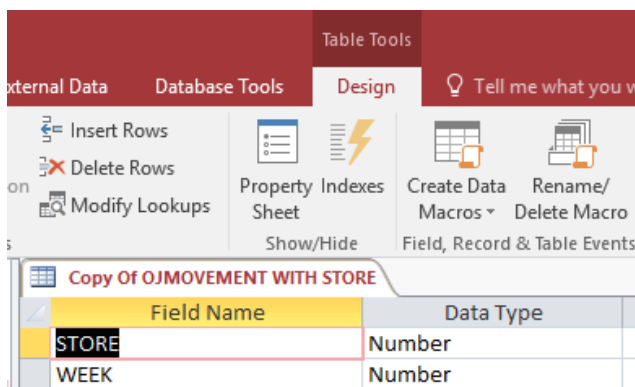
Field Name	Data Type
STORE	Number
WEEK	Number
STOREWEEK	Number
BRAND	Short Text
MOVE	Number
logmove	Number
PRICE	Number
logprice	Number
PROFIT	Number
Feat	Number
Season	Short Text

- Click Create → Query Design. Select OJMOVEMENT WITH STORE → Add → Close.
- Double click on * in the list of variables.
- Append to copy of OJMOVEMENT WITH STORE and click yes when prompted.
- You will get a message that you cannot append all cases. Click Yes
- Copy of OJMOVEMENT WITH STORE is now OJMOVEMENT WITH STORE with duplicates of STOREWEEK removed.
- For each unique value of STOREWEEK, only the first case is retained.

Case 3. We have a table with multiple fields, and we want to remove duplicates based on a combination of multiple fields.

Example: Suppose we want to start with OJMOVEMENT WITH STORE, and remove duplicates for the combination of the fields STORE and WEEK. (The procedure described can be used for a combination of up to ten fields.)

- In the list of tables in the left pane, right click OJMOVEMENT WITH STORE → copy → paste
- Paste to copy of OJMOVEMENT WITH STORE and select “structure only.”
- Right click to open Copy of OJMOVEMENT WITH STORE in design view.
- In the top ribbon, under Table Tools, click Indexes



Field Name	Data Type
STORE	Number
WEEK	Number

- In the dialog box, type ID1 in index, and select STORE and WEEK in fields.

Indexes: Copy Of OJMOVEMENT WITH STORE		
Index Name	Field Name	Sort Order
ID1	STORE	Ascending
	WEEK	Ascending

Index Properties

The name of the field to be indexed.

- Right click at the left of ID1. A box called “Index Properties” appears. In this box, use the drop down menu to change No for Primary to Yes

Indexes: Copy Of OJMOVEMENT WITH STORE		
Index Name	Field Name	Sort Order
ID1	STORE	Ascending
	WEEK	Ascending

Index Properties

Primary	Yes
Unique	Yes
Ignore Nulls	No

If Yes, this index is the primary key.

- Save the change in the design of Copy of OJMOVEMENT WITH STORE
- Click Create ➔ Query Design. Select OJMOVEMENT WITH STORE ➔ Add ➔ Close.
- Double click on * in the list of variables.
- Append to copy of OJMOVEMENT WITH STORE and click yes when prompted.
- You will get a message that you cannot append all cases. Click Yes
- Copy of OJMOVEMENT WITH STORE is now OJMOVEMENT WITH STORE with duplicates of the combination of STORE and WEEK removed.
- For each unique combination of STORE and WEEK, only the first case is retained.

Note:

- In both Case 2 and Case 3, we first copied the table into a table that only had the structure of the original table.
- We next opened the copy in design view and selected the field or combination of fields that should not be duplicated as primary key. (For multiple fields, we used Index.)
- Finally, we appended the original table to the copy of the table. When appending, Access only allows unique values of the primary key to be appended. This removes duplicates.

Case 4. Removing selected cases

In cases 1, 2 and 3, we simply retained the first time a field or combination of fields occurs in a sample. In some cases, we may want to look at the duplicates and select cases to remove. As the process is similar to that of selecting a random sample, only a sketch of the method is provided.

- Make sure table has an ID field. You can always add one using design view and auto number.
- Use find duplicates query wizard to prepare a table of duplicates. Include ID as a field in this table.
- From the table of duplicates, prepare a table that of cases you want to remove. Add a field called flag that is 1 for all cases in this table. Let us call this table **remove**. The field ID in this table is a list of cases you want to remove from the original table.

You can now remove the duplicates in three different ways.

Method 1 (by hand): Open the table in data sheet view. Remove the cases you want by right click at the left of the row and selecting **Delete Row**. This method is appropriate only when the sample is small. The next two methods are appropriate for large samples.

Method 1 (using SQL):

- Click Create → Query design → Add table of interest → Close
- Select the fields you want in your final table. Make sure ID is included.
- **Remember:** Make sure you have a table called “remove” that has a field called ID that is the list of cases to remove.
- In the criteria box under ID, enter Not In (select ID from remove).
- This removes the cases you want to remove.
- This method usually takes a long time to run if you want to remove a large number of cases.

Method 2 (not using SQL):

- Use Database Tools → Relationships to join Table of interest with remove.
- Use ID to join such that you include ALL records from Table of interest and only those records from remove where the joined fields are equal. Save the relationship.
- **Remember:** Make sure you have a table called “remove” that has a field called ID that is the list of cases to remove.
- Create → Query Design → Add Table of interest and remove → Close
- From Table of interest, select variables you need including ID. From remove, select Flag.
- In the criteria box under Flag, type Is NULL
- Use make table to save the table. This table has no longer has the duplicates.

8. Data Cleaning with Access: Finding and Removing Missing Cases

To show how to find and remove missing cases, we will use the file country export military.xlsx. This file has three worksheets obtained from World Data 2016 published by the World Bank.

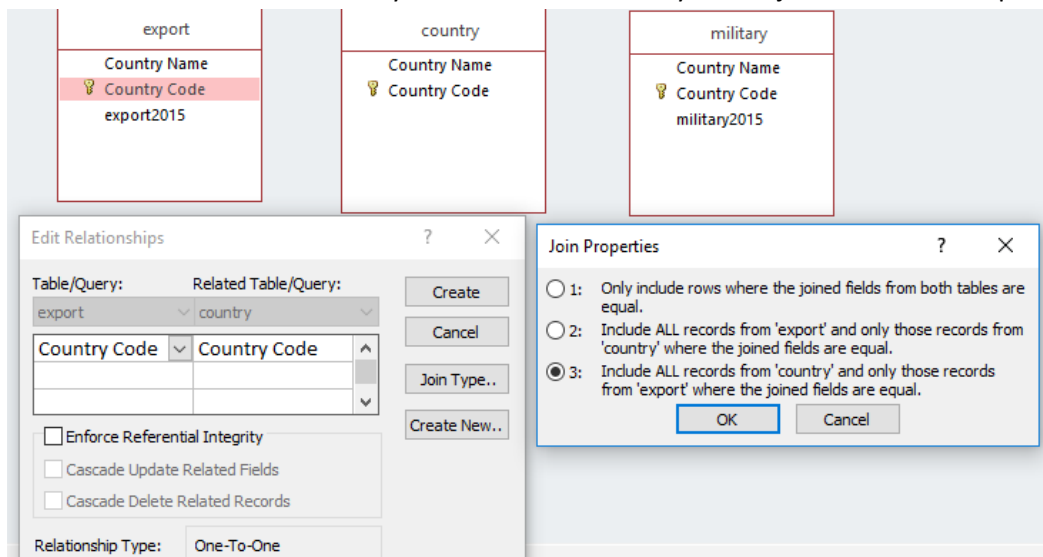
- The worksheet country lists the names of all countries and the country code for each country. There are no missing cases in this worksheet.
- The worksheet export lists country names, country codes, and exports in 2015 as percent of GDP for the countries for which such data are available.
- The worksheet military lists country names, country codes, and military expenditures in 2015 as percent of GDP for the countries for which such data are available.

From these three worksheets, we will prepare four tables:

- Table of all cases where we have both export and military data.
- Table of all cases where we have export but do not have military.
- Table of all cases where we have military but not export.
- Table of all cases where we have neither military nor export.

Step 1: Import data and prepare master data file

- We first import the worksheet country. In this case, we have to specify that the first row contains column headings. Select Country Code as primary key.
- Import the worksheets export and military as tables. For each table, we select Country Code as the primary key.
- Click Database tools → Relationships → Show Table → country → add → export → add → military → add → close
- Join country and export using Country Code such that we include all cases from country, and cases from export where joined fields are equal. Similarly, joined country and military such that we include all cases from country and cases from military where joined fields are equal.





- Save relationships.

Step 2. Find and remove missing cases using Is Null and Is Not Null

- Click create → query design → country → add → export → add → military → add → close
- Select country name and country code from country, export2015 from export, and military2015 from military.
- In criteria boxes for both export2015 and military2015, type in Is Not Null

Field:	Country Name	Country Code	export2015	military2015
Table:	country	country	export	military
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Is Not Null	Is Not Null
or:				

The data sheet view shows 144 cases where we have data for both Export and Military, that is, there are no missing cases. Use make table → provide name → ! to save the table.

- In criteria box for Export2015, type Is Not Null, and in criteria box for Military2015 type Is Null. The data sheet view shows 36 cases where we have data for Export2015, but Military2015 is missing. Use make table → provide name → ! to save table.

Field:	Country Name	Country Code	export2015	military2015
Table:	country	country	export	military
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Is Not Null	Is Null
or:				

- Similarly, in criteria box for Export2015, type Is Null, and in criteria box for Military2015 type Is Not Null. The data sheet view shows 34 cases where we have data for Military2015, but Export2015 is missing. Use make table → provide name → ! to save table.
- Finally, type Is Null in criteria boxes for Export2015 and Military2015. This gives the 50 cases for which both Export2015 and Military2015 are missing.

Field:	Country Name	Country Code	export2015	military2015
Table:	country	country	export	military
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Is Null	Is Null
or:				

9. Data Cleaning: Using Access to Find Unmatched Cases

We will use Query Wizard to check if two tables have mismatched cases. This method can only check mismatches for **one field**. However, you can create a new field that combines two or more fields (like STOREWEEK) and check if there is a mismatch, that is, a certain value is present in one table and not the other.

The data set called 255survey.xls collected from undergraduate students in 2012. The dataset has two worksheets, 255survey1, and 255 survey2.

255survey1: Last name, first name, brand of laptop owned, brand of laptop respondent wished to own.

255survey2: First name, last name, first brand of smartphone (if any) the respondent owned.

These are class participation data where each student was supposed to submit a survey once. Each combination of last name and first name unique, that is, not repeated. While the names of the students are actual names, I randomly changed the brands owned to preserve confidentiality.

Open 255survey.xls in Access and import the two worksheets 255survey1 and 255survey2 as tables.

- When importing each worksheet, check the box that the **“First row contains column headings.”**
- Choose ID1 as the primary key for 255survey1 and ID2 as the primary key for 255survey2.

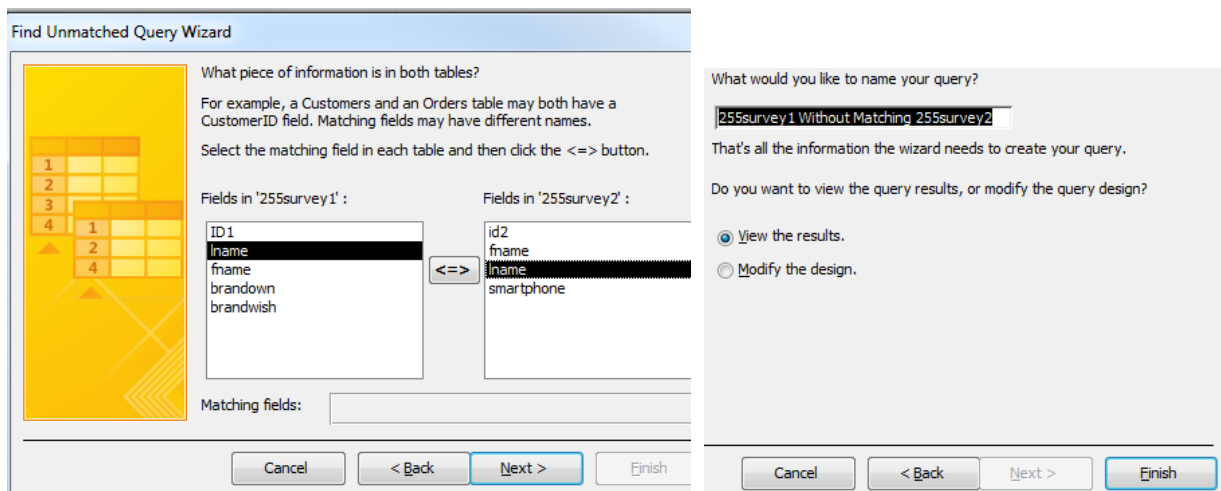
Case 1: Click on table 255survey1 → Query Wizard → Find Unmatched Query Wizard → OK

Click 255survey1 → Next → 255survey2 → Next

Click on lname on both sides → Next

Select ID1, fname and lname as fields you wish to see, and click Next

Click Finish to view results. The results appear as a table, which can be saved from design view.



All Access Objects

Search...

Tables

- 255survey1
- 255survey2
- Table1

Queries

- 255survey1 Without Matching 255survey2
- Find duplicates for 255survey1
- Find duplicates for 255survey2

ID1	fname	lname
38	Elliott	Brianna
64	Ebby	Kahen Kashani
84	OMAR	ALBANAWI
122	justin	murray
149	Xiaonan	Jia
159	Duan	Guochen
165	Feng	Xiaopei
168	ozemary	feliz
169		
173	Matthew	Dumoff

Case 2: Click on table 255survey2 → Query Wizard → Find Unmatched Query Wizard → OK

Click 255survey2 → Next → 255survey1 → Next

Click on lname on both sides → Next

Select ID2, fname and lname as fields you wish to see, and click Next

Click Finish to view results

All Access Objects

Search...

Tables

- 255survey1
- 255survey2
- Table1

Queries

- 255survey1 Without Matching 255survey2
- 255survey2 Without Matching 255survey1
- Find duplicates for 255survey1
- Find duplicates for 255survey2

id2	fname	lname
1	Omar	AL Banawi
12	Annalise	Brod
20	Unwoo	Chun
31	Guochen	Duan
36	Brianna	Elliott
100	James	Mullen
106	Stephanie	Pagano
130	Justin	Sohn

Case 3. From Table 255survey1, use Make-Table Query to create a new table survey1 that includes

ID1, fname, lname, fullname, brandown, brandwish

FULLNAME is computed as fullname: [fname]+" "+[lname]

" ", that is, quotation marks with a space between them, creates a space between fname and lname.

Field:	ID1	fname	lname	fullname: [fname]+" "+[lname]	brandown	brandwish
Table:	255survey1	255survey1	255survey1		255survey1	255survey1
Sort:						
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:						
or:						

Similarly, from Table 255survey2, make table survey2 that includes

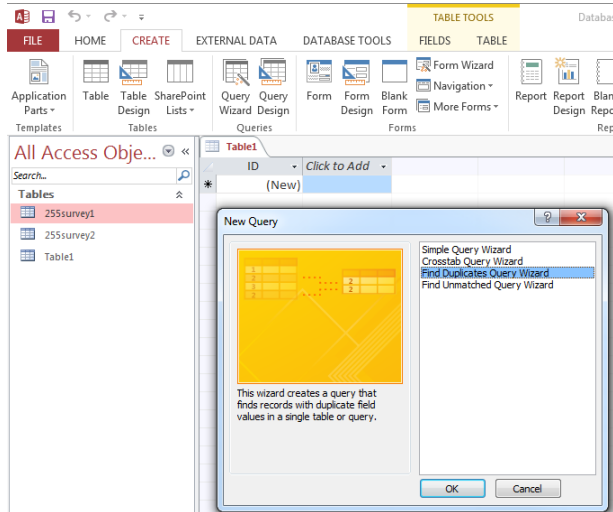
ID2, fname, lname, fullname, smartphone. If you run the find unmatched cases query using fullname, you will get more mismatched cases because of different spellings of first name entered.

10. More Examples of Data Cleaning with Access

Example 1: We will again use the data set 255survey.xls to check if there are duplicates, that is, whether a student made multiple submissions.

Click on the table 255survey1.

Click CREATE → Query Wizard → Finding Duplicates Query Wizard → OK

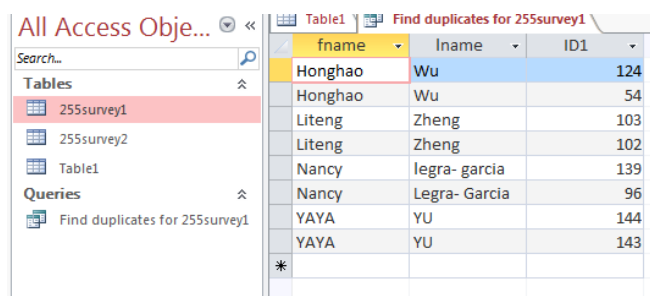


Click 255survey1 → Next

Select fname, lname, Next

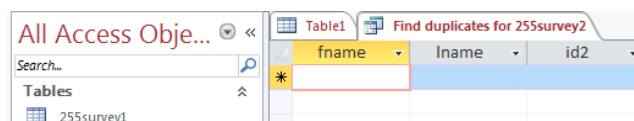
Select ID1 as the Additional Query Field, and click Finish

The duplicates in 255survey1 are shown as a table.



fname	lname	ID1
Honghao	Wu	124
Honghao	Wu	54
Liteng	Zheng	103
Liteng	Zheng	102
Nancy	legra- garcia	139
Nancy	Legra- Garcia	96
YAYA	YU	144
YAYA	YU	143

Proceeding similarly, we find that there are no duplicates in 255survey2, as shown below.



fname	lname	id2
Honghao	Wu	54

We open the table 255survey1 in Datasheet View and delete the first time the student appears from the worksheet by right clicking at the left of the row and selecting **delete record**.

Example 2. Import the data set zipcode.xls into Access. This dataset has two worksheets:

Auto: The two fields are zip (zip code), and newcar (the number of new car dealerships in the zip code).

Acclegal: The two fields are zip (zip code), and acclegal (the number of Accounting/Legal offices in the zip code).

For each worksheet, select **zip** as the **primary key**. In each worksheet, no zip code is repeated. However, there are zip codes where we have data for auto but not acclegal and vice versa.

We will:

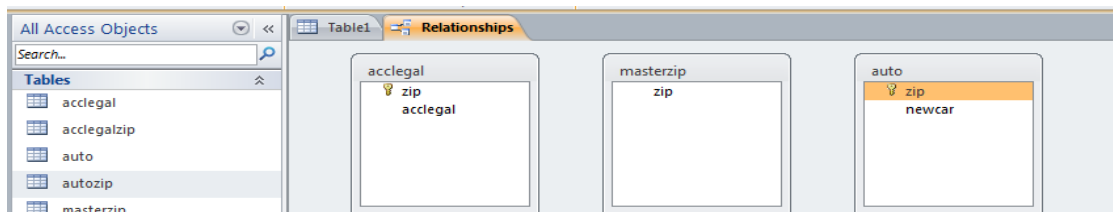
- Create a table where we have the all the zip codes, number of new car dealerships if available, and number of accounting/legal offices if available.
- Create a table where both the numbers are available.
- Create a table where the number of new car dealerships is available, but the number of accounting/legal offices is not available.

Step 1. First, create a master list of zip codes called **masterzip** as follows.

- From table **auto**, use make-table query to create a table called **autozip** that only contains one field, zip
- From table **acclegal**, use make-table query to create a table called **acclegalzip** that only contains one field, **zip**.
- Append table **acclegalzip** to table **autozip**.
- The table **autozip** now has all zip codes, possibly with duplicates.
- From table **autozip**, use make-table query to create a table called **masterzip** that has one field zip (zip codes) where duplicates are removed by clicking Σ

Step 2. Click Database Tools → Relationships

Click acclegal → Add → masterzip → Add → auto → Add

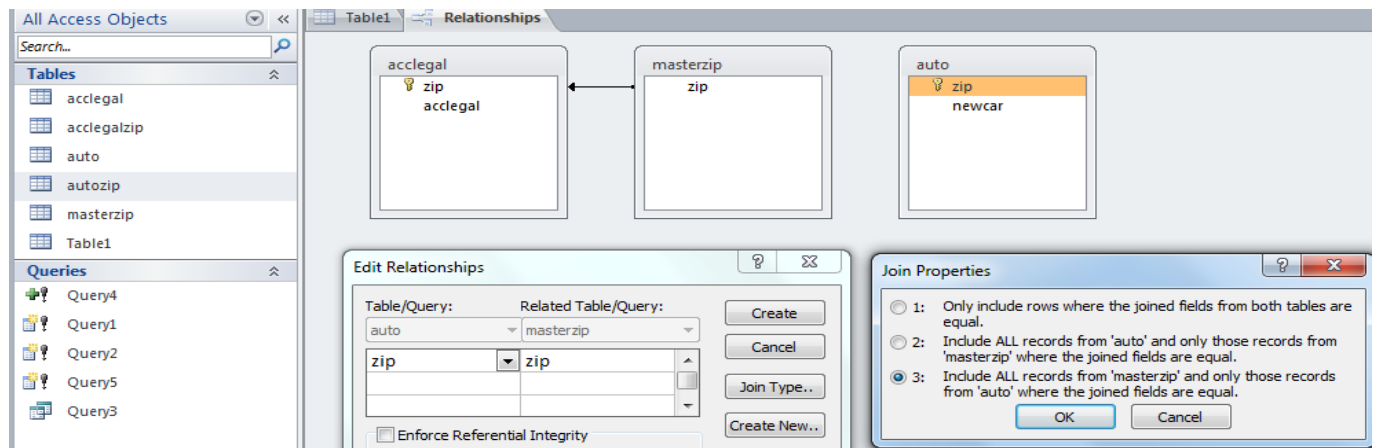


Drag **zip** from **acclegal** to **zip** in **masterzip**.

Click Join Type → Include ALL records from 'masterzip' and only those records from 'acclegal' where the joined fields are equal → OK → Create

Drag zip from auto to zip in masterzip.

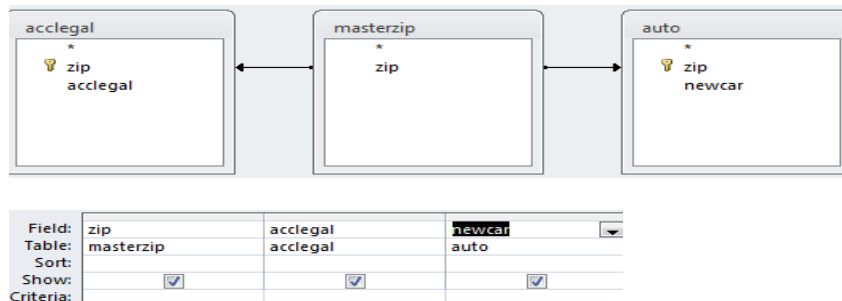
Click Join Type → Include ALL records from 'masterzip' and only those records from 'auto' where the joined fields are equal → OK → Create



Save the changes in relationship.

Click CREATE → Query Design → acclegal → Add → masterzip → Add → auto → Add → Close

Double click to select: zip from masterzip, acclegal from acclegal, newcar from auto.



Click Make-Table, name table **ALL ZIP**, click !

The table ALL ZIP contains data for all zip codes. Some cases may have missing data.

For fields acclegal and newcar:

- Type **Is Not Null** in both criteria boxes. Make table called **ALL ZIP NOT NULL**.
- Type **Is Not Null** in criteria box for acclegal and **Is Null** in criteria box for newcar. Make table **AUTO NULL**.
- Type **Is Null** in criteria box for acclegal and **Is Not Null** in criteria box for newcar. Make table **ACCLEGAL NULL**.
- Finally, type **Is Null** for both acclegal and auto criteria boxes and make table. You get a table with 0 rows. This means that every zip code in masterzip has data for at least one of acclegal and newcar fields.