Data Analysis: A Practical Introduction for Absolute Beginners

Lab 5: Data Joins

Learning Objectives

- Perform Inner Joins, Full Outer Joins, Left Joins, and Right Joins by hand in Excel.

Data Set

There are **two different data sets** for this lab:
Mod3Lab2a.csv
Mod3Lab2b.csv

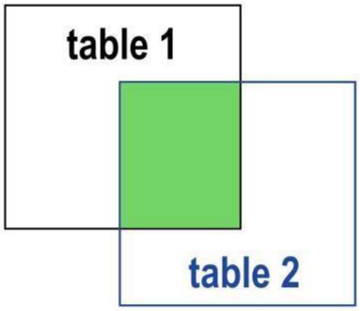
What You 'll Need

To complete the lab, you will need the online version of Microsoft Excel.

Overview

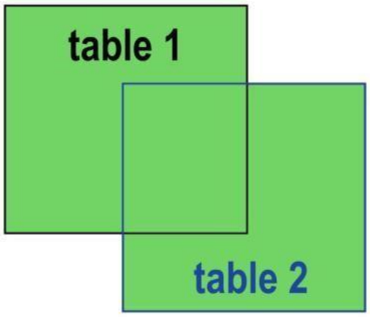
In data analysis, a “join” is a combination two (or more) tables into a single table, based on related columns in the tables. In this lab, we’ll practice manually performing four different types of data join in Excel. Before we get started, let’s run through a quick refresher on the definitions. Here are four of the main types of join:

Inner Join: From two tables of data, this type of join returns a new table that *only* includes matching values from both tables. Any row or column that only shows up in one table (instead of both) will be ignored and not added to the join.



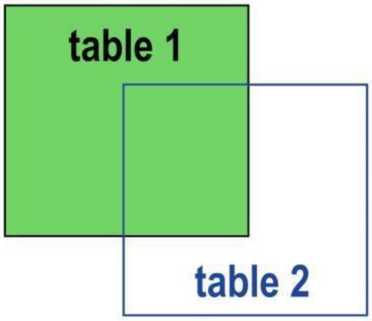
Inner Join

Full Outer Join: Also known as simply an Outer Join, this is basically the exact opposite of an Inner Join: It returns every single row *and* column from both tables, even if they don't match the rows/columns in the other table. If there are rows without matching values, the new join table will return a "null" or "NA" entry for those cells (or simply be empty). A Full Outer Join will have the largest possible number of results.



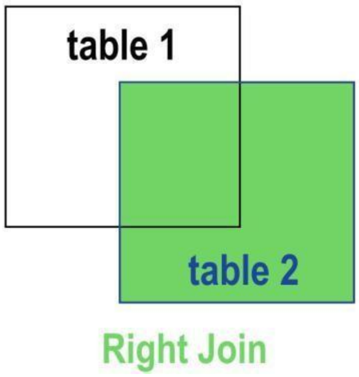
Full Outer Join

Left Join: A special type of Outer Join where the "left" table (i.e. the table that's listed first) is favored. That means it returns *every* row and column from the left table, but only matching rows from the right table.



Left Join

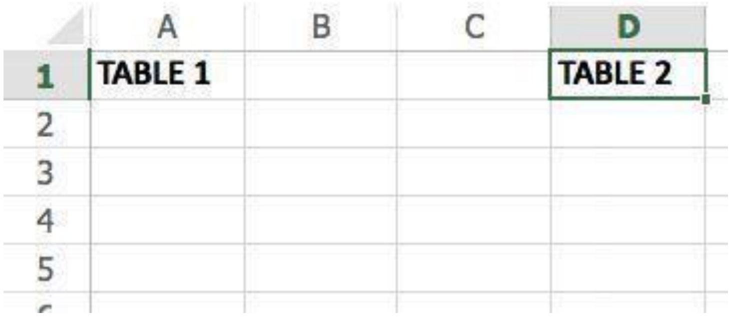
Right Join: A special type of Outer Join where the "right" table (i.e. the table that's listed second) is favored. It returns *every* row and column from the right table, but only matching rows from the left table.



Exercise 1: Inner Joins

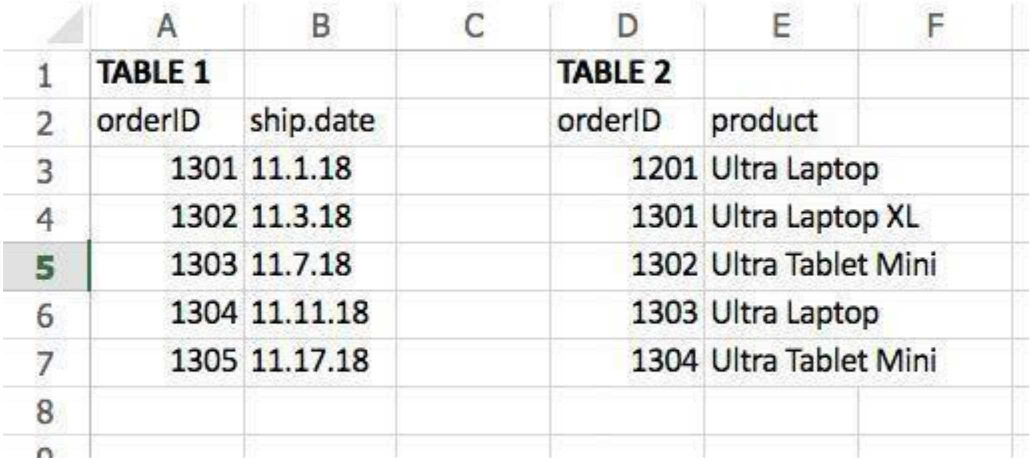
We’ll start by performing an Inner Join on the two data sets.

1. In a new blank worksheet in Excel Online, type **TABLE 1**in cell A1, and type **TABLE 2**in cell D1. This’ll help keep your data sets straight.



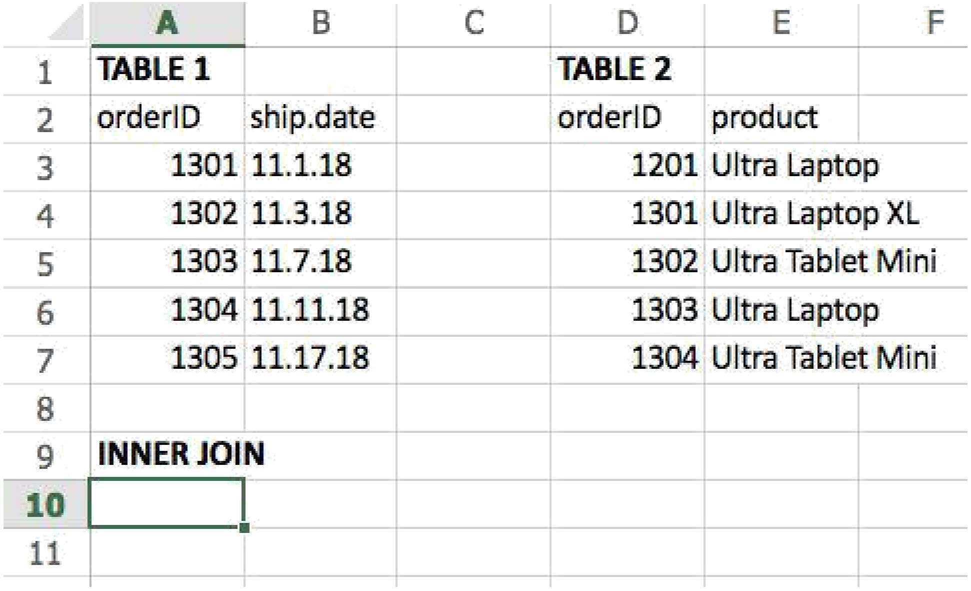
	A	B	C	D
1	TABLE 1			TABLE 2
2				
3				
4				
5				

2. Open both data sets in Excel. To do this, copy and paste the data from the first data set (Mod3Lab2a.csv) below TABLE 1, and copy and paste the second data set (Mod3Lab2b.csv) below TABLE 2.



	A	B	C	D	E	F
1	TABLE 1			TABLE 2		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						

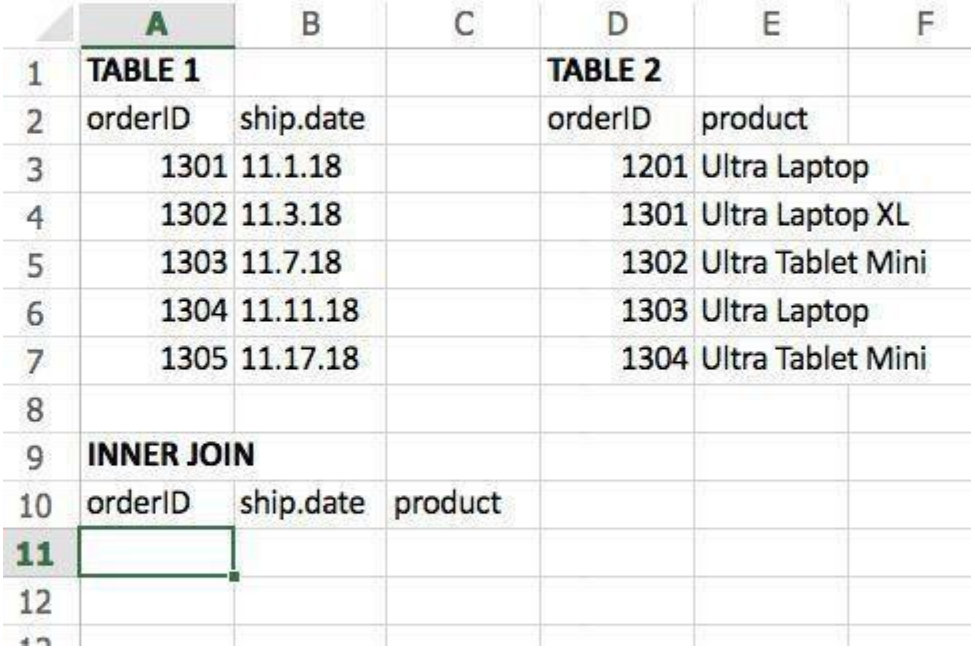
3. Down in cell A9, type **INNER JOIN**.



	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						
9	**INNER JOIN**					
10						
11						

4. Below that is where you'll create your new table to represent the Inner Join. Now it's time to think through what an Inner Join will look like with these two small data sets. Your Inner Join should be a new table that *only*includes matching values from both tables, ignoring any values that only show up in one table.

Your two tables both contain "orderID" values — that's the common variable between Table 1 and Table 2. Table 1 pairs these order IDs with their ship dates, while Table 2 pairs each order ID with a product name. Since you're "joining" the two tables, your new table should have columns for order ID, ship date, *and*product. Type those in, like so:

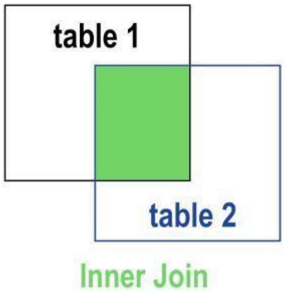


	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						
9	**INNER JOIN**					
10	orderID	ship.date	product			
11						
12						

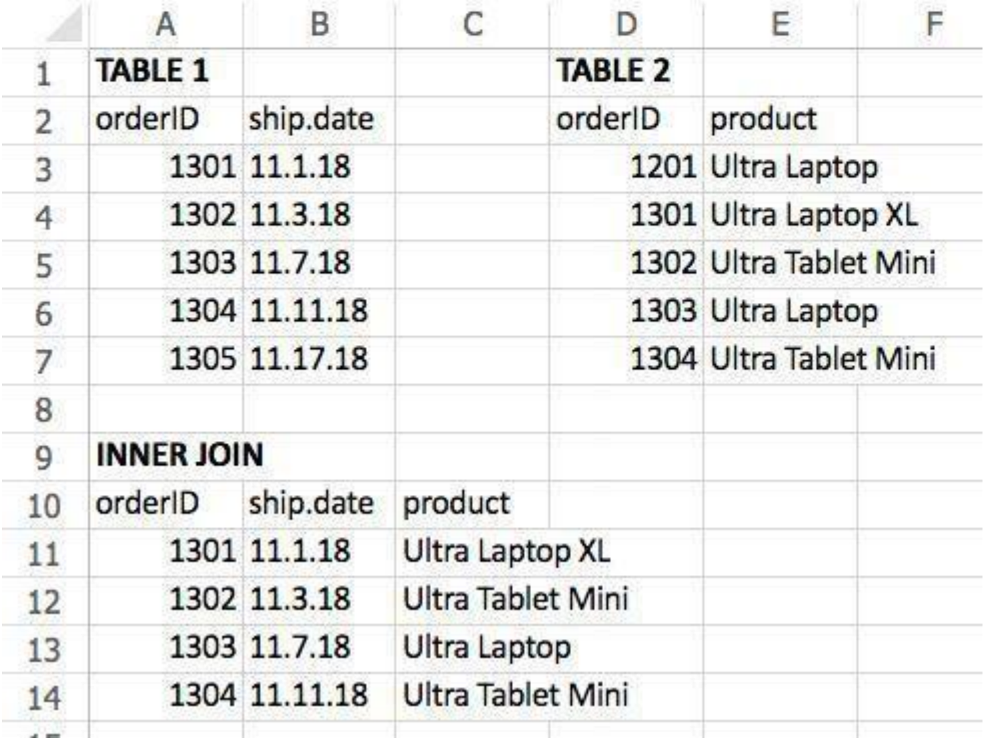
For example, order ID number 1301 was shipped on 11.1.18 (according to Table 1), and that same order included the Ultra Laptop XL product (according to Table 2, though note that order 1301 is in the *second* row of Table 2). The idea behind the join is that you want to have both these pieces of information in a single handy table.

5. Now figure out which data to include in the new Inner Join table. It's not as simple as just copying all the data for both tables into the new one. Why not?

Check out those order ID numbers. Notice how both tables include the ID numbers 1301, 1302, 1303, and 1304 — but Table 1 includes 1305 as well, which is *not*in Table 2. Similarly, there's an order ID number 1201 in Table 2 that doesn't show up in Table 1. Since an Inner Join should ignore values that only show up in one table, you *don't*want to include 1305 or 1201 in the new table. Before looking at the next step below, see if you can copy and paste the correct data from Tables 1 and 2 to form the new Inner Join table. Here's a hint, using that graphic from earlier:



6. Got it? Here's what your Inner Join should look like:



	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						
9	**INNER JOIN**					
10	orderID	ship.date	product			
11	1301	11.1.18	Ultra Laptop XL			
12	1302	11.3.18	Ultra Tablet Mini			
13	1303	11.7.18	Ultra Laptop			
14	1304	11.11.18	Ultra Tablet Mini			

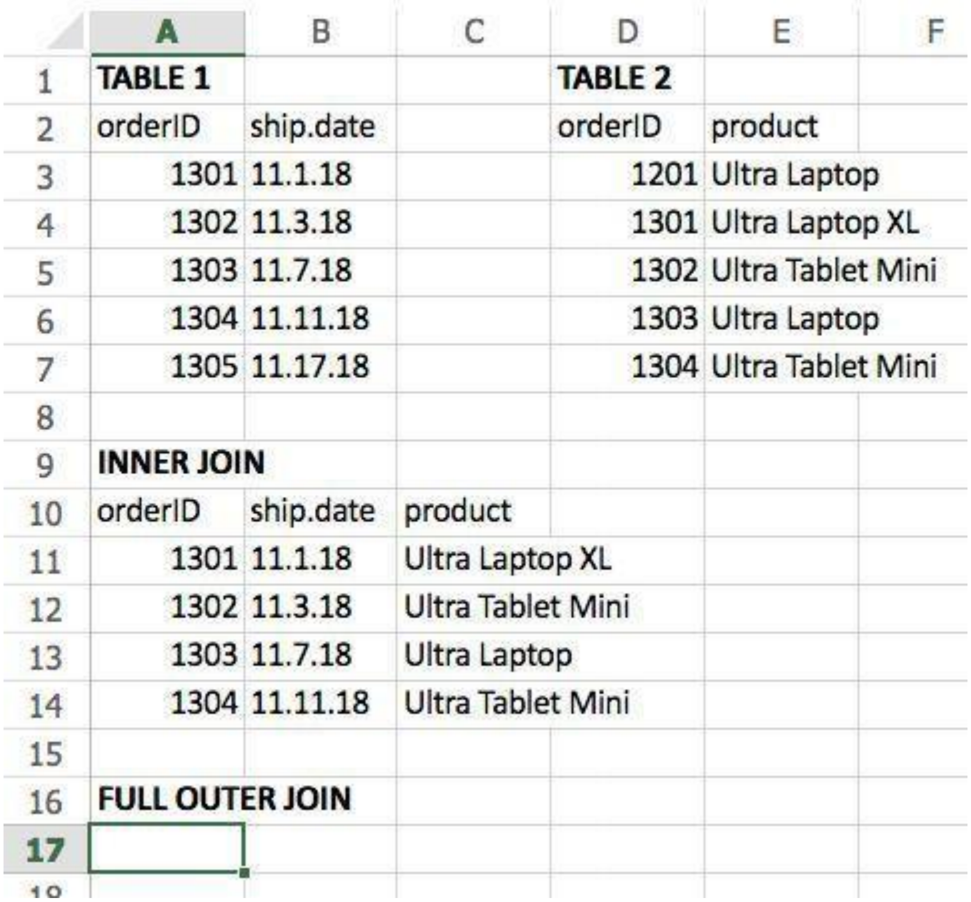
We only included the info for orders 1301, 1302, 1303, and 1304 because the two original tables had those order numbers in common. Orders 1305 (in the bottom row of Table 1) and 1201 (in the top row of Table 2) get ignored.

Obviously this can get a lot more complex with larger data sets, but that's the basic rundown of an Inner Join!

Exercise 2: Full Outer Joins

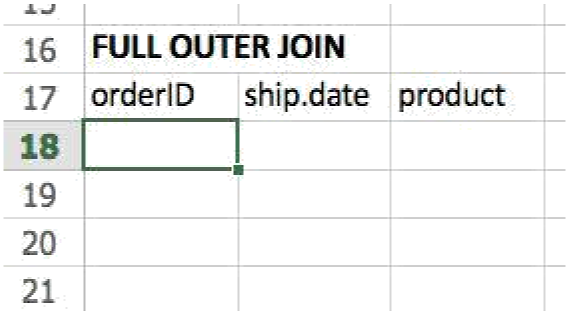
Now we'll use those same two data sets to perform a Full Outer Join. It's a bit more straightforward than the Inner Join.

1. In the same Excel worksheet you used in Exercise 1, set up a new table for your Full Outer Join.



	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						
9	**INNER JOIN**					
10	orderID	ship.date	product			
11	1301	11.1.18	Ultra Laptop XL			
12	1302	11.3.18	Ultra Tablet Mini			
13	1303	11.7.18	Ultra Laptop			
14	1304	11.11.18	Ultra Tablet Mini			
15						
16	**FULL OUTER JOIN**					
17						

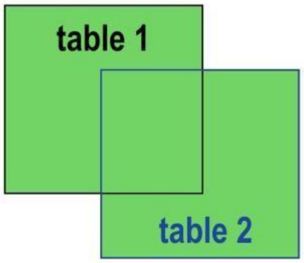
2. Once again, you're "joining" the two tables, so your new table should have columns for order ID, ship date, and product.



16	FULL OUTER JOIN		
17	orderID	ship.date	product
18			
19			
20			
21			

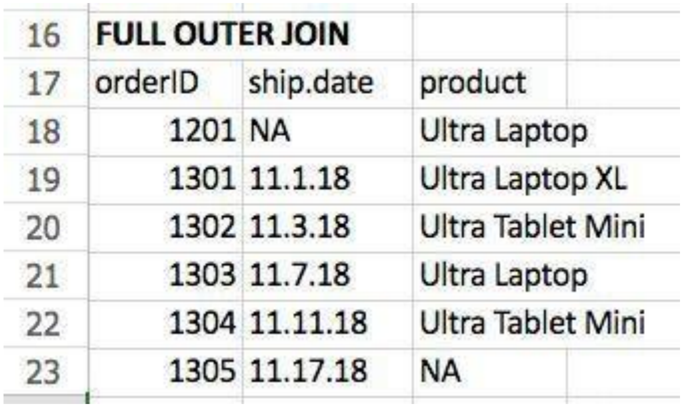
3. Think it through. A Full Outer Join includes all columns and rows from both tables, with "NA" (null) values for any cell that only shows up in one table.

   You want to "fully" join both tables this time, which means you *do* want to include those rogue orders 1305 from Table 1 and 1201 from Table 2. Try it out for yourself before looking down at the answer below. Here's the graphic again:



Full Outer Join

4. Here's what your Full Outer Join table should look like:



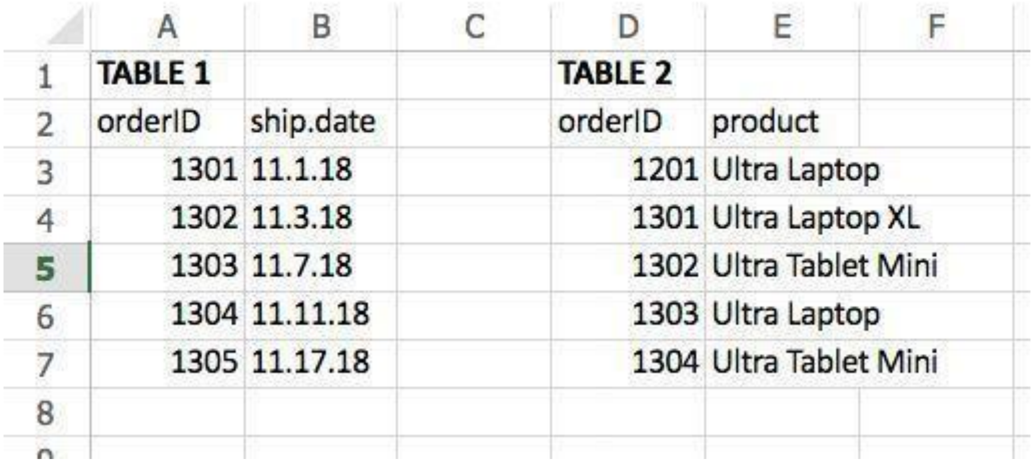
16	FULL OUTER JOIN		
17	orderID	ship.date	product
18	1201	NA	Ultra Laptop
19	1301	11.1.18	Ultra Laptop XL
20	1302	11.3.18	Ultra Tablet Mini
21	1303	11.7.18	Ultra Laptop
22	1304	11.11.18	Ultra Tablet Mini
23	1305	11.17.18	NA

   Notice how there's no ship date for order 1201 (because that order doesn't show up in Table 1), and there's no product name for order 1305 (because that one doesn't show up in Table 2). Instead, we add an "NA" or "null" value in those two spots.

Exercise 3: Left Joins

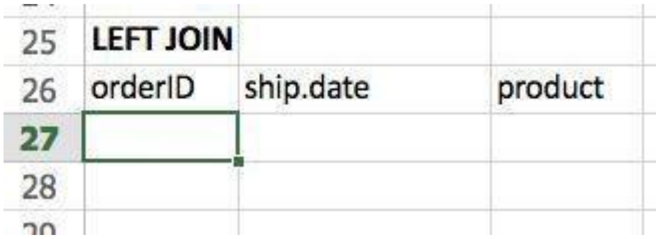
Next up, we'll run through a Left Join on our two original data sets.

1. Here are the original data sets again, just as a refresher:



	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						

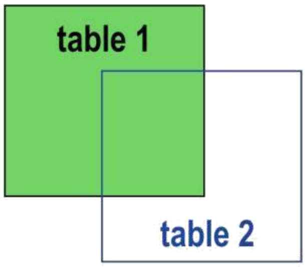
2. In the same Excel worksheet you used in the first two exercises, set up a new table for your Left Join (down below the joins from the other exercises). Once again, you'll want to include all three variables in your new table because you're joining the two originals.



25	**LEFT JOIN**		
26	orderID	ship.date	product
27			
28			

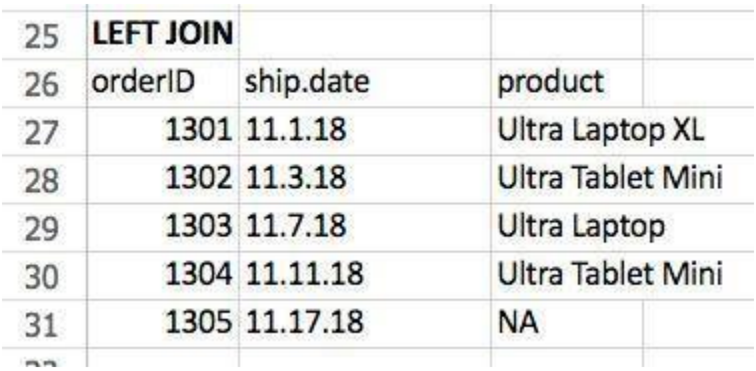
3. Which data should you include in the new table? In a Left Join, the "left" table (i.e. the table that's listed first) is favored. That's Table 1 in this case. Your new table should return *every* row and column from Table 1, but only matching rows from Table 2. It might be helpful to think of it this way: You're treating Table 1 like a Full Outer Join, but you're treating Table 2 like an Inner Join.

Try to find the answer yourself before looking ahead at the next step. Here's the graphic again for a Left Join:



Left Join

4. Ready? Here's what your Left Join table should look like:



25	**LEFT JOIN**			
26	orderID	ship.date	product	
27	1301	11.1.18	Ultra Laptop XL	
28	1302	11.3.18	Ultra Tablet Mini	
29	1303	11.7.18	Ultra Laptop	
30	1304	11.11.18	Ultra Tablet Mini	
31	1305	11.17.18	NA	

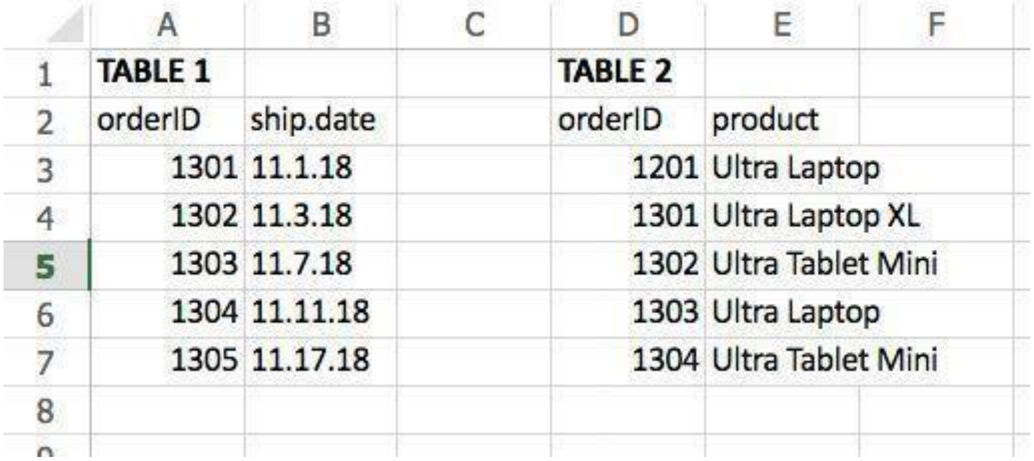
Notice that we included all five orders from Table 1 (the “left” table), including that weird order 1305 that didn’t show up in Table 2. But we skipped the extra order from Table 2 (1201) because the “right” table is *not* favored.

5. Name the worksheet as Exercise
   - Save the File Mod3Lab2

Exercise 4: Right Joins

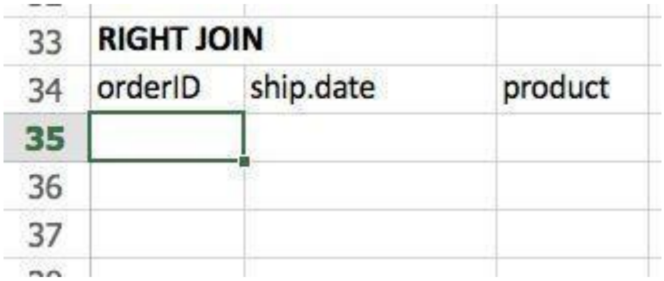
One more! For this last exercise, we’ll perform a Right Join on the same two data sets we’ve been looking at.

1. Here are the original data sets again:



	A	B	C	D	E	F
1	**TABLE 1**			**TABLE 2**		
2	orderID	ship.date		orderID	product	
3	1301	11.1.18		1201	Ultra Laptop	
4	1302	11.3.18		1301	Ultra Laptop XL	
5	1303	11.7.18		1302	Ultra Tablet Mini	
6	1304	11.11.18		1303	Ultra Laptop	
7	1305	11.17.18		1304	Ultra Tablet Mini	
8						

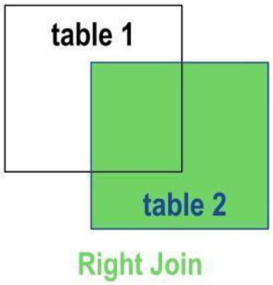
2. In the same Excel worksheet you’ve been using for the other exercises, set up another new table for the Right Join (down below the tables from the other exercises). Once again, you’ll want to include all three variables in your new table because you’re joining the two originals.



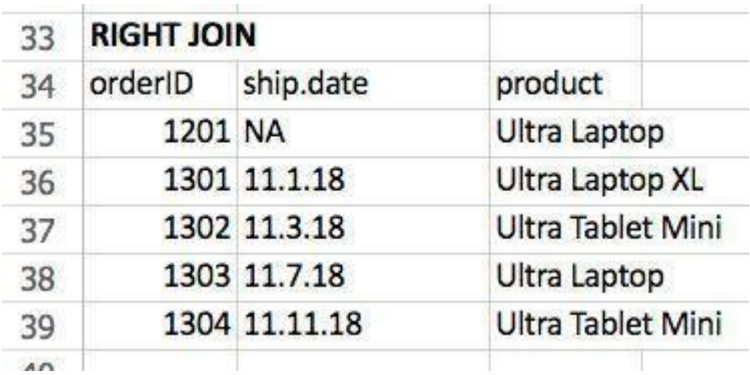
33	**RIGHT JOIN**		
34	orderID	ship.date	product
35			
36			
37			

3. Time to think it through again. In a Right Join, the "right" table (i.e. the table that's listed second) is favored. That's Table 2 this time around. The Right Join returns *every* row and column from the second table, but only matching rows from the first table. So in this case, it's kind of like you're doing an Inner Join on Table 1, and a Full Outer Join on Table 2.

4. You know the drill: Try to copy and paste the correct data from the original tables to create your new Right Join. Don't look ahead at the next step until you've tried it. Here's the graphic showing a Right Join:



5. And here's what your Right Join should look like:



33	**RIGHT JOIN**		
34	orderID	ship.date	product
35	1201	NA	Ultra Laptop
36	1301	11.1.18	Ultra Laptop XL
37	1302	11.3.18	Ultra Tablet Mini
38	1303	11.7.18	Ultra Laptop
39	1304	11.11.18	Ultra Tablet Mini

This time, we included every row from Table 2, including that rogue order number 1201, which doesn't have a ship date because it didn't show up in Table 1. But we did *not* include the extra order from Table 1 (1305) because the "left" table is not favored this time.