Lab 8: Importing, Joining Tables

***What You'll Learn*:** This lesson introduces importing text files into an ArcMap table, combining rows, and navigating tricky joins. The work is organized as two small projects, the first with step-by-step instructions, the second less so, as most operations have been introduced previously, and we expect you to be familiar with them by now. This project requires synthesis of what you've learned in this and previous labs.

Data are in the L8\ directory, with the \Project1 subdirectory containinting a */wr48.shp* file and a *states.shp* file, both in NAD83 geographic coordinates, and the \Project2 subdirectory containing a *Cal.shp* file in WGS84 geographic coordinates.

What You'll Produce: Three maps, one of U.S. NASS data, one of California county income, and one of California counties with parks or forests.

Project 1

This project introduces something quite common, joining ASCII tabular data with a shapefile. Here, we will combine a text file on corn production for US counties with a county shapefile, but there are many types of tabular data that are available as text files, summarized on a county basis, including population, voting, education, income, crime, air pollution, and many other social, political, and environmental data.

Here we import a text file, convert it to a ArcMap compatible table, and edit the table, deleting columns, creating join items, and combining rows before joining it with a polygon shape file. These are all common operations when ingesting tabular data.

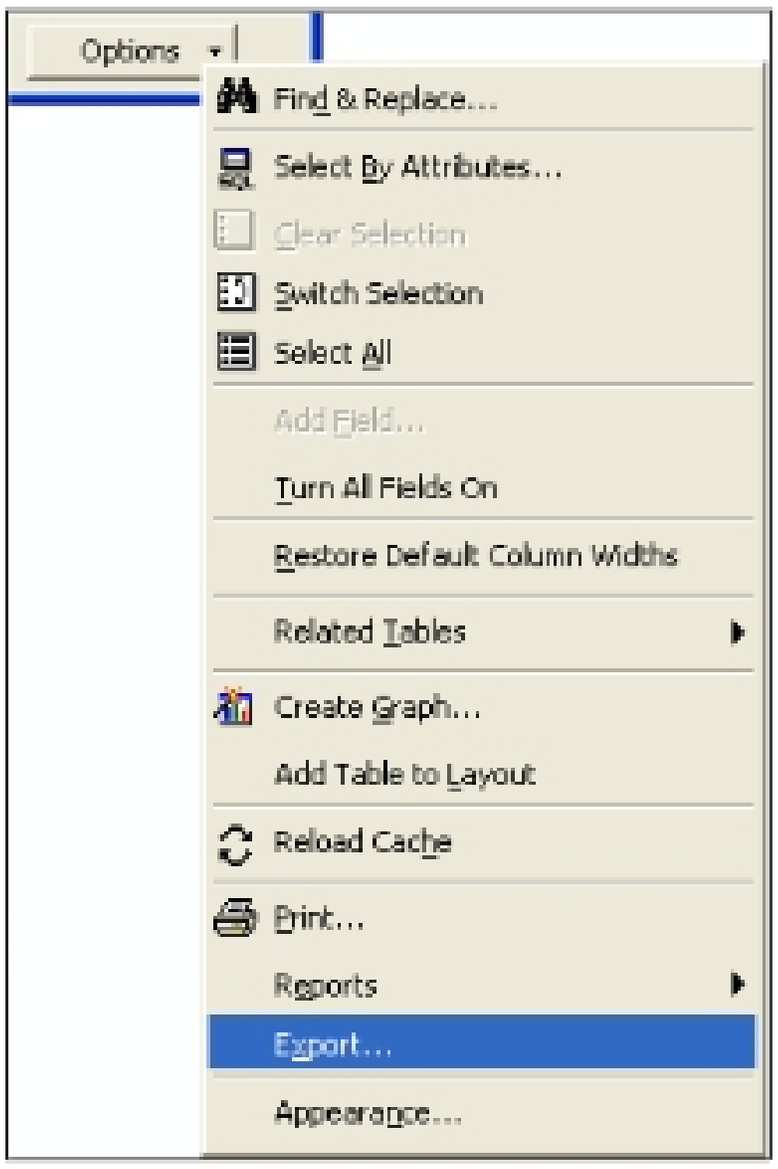
Start ArcMap, and add */wr48.shp* from the L8\Project1\ subdirectory.

Now add the text file *cnty26.csv* to this data view, and open the table for viewing. ***(Video: L8_1_add_texttable.mov)***
This file contains 1996 seed corn production, in bushels, for counties in the United States. These data were downloaded from the National Agricultural Statistical Service website, www.nass.usda.gov/, and we're most interested in the columns:

Stfips: the state Federal Information Processing System (FIPS) code
CoFips: county FIPS code
Harvested: the acres harvested for a given yield category in a county
Yield: Bushels per acre harvested for the yield category
Production: Total bushels produced (yield times harvested) for the given yield level.

Unfortunately, we can't directly edit the .csv file, so we must convert it to a dbf file.

In the Attributes of *cnty26.csv* window, left click on the **Options** button in the lower-right portion of the table window frame, then **Export** in the dropdown menu, and save all records to the Project1 subdirectory, naming it something like "*raw_corn_dat*".



Remove the *cnty26.csv* from the data frame, to reduce clutter, and add and open *raw_corn_dat.dbf* table in the data frame.

Delete all the columns except the following: OID, Stfips, CoFips, Harvested, Yield, Production.

Delete columns by right clicking in the column heading, and selecting **Delete** near the bottom of the dropdown menu. Left click on the Yes button in response to the warning about deleting outside of an edit session.

We now want to join these data with the county shapefile, *lwr48.shp*. Unfortunately, there are two problems. First, we don't have a ready-made key for the join. There is no column that maps cleanly from the *raw_corn_dat.dbf* file to the *lwr48.dbf* file.

Let's first fix this problem ***(Video: L8_2_create_index.mov)***.

Open the data table for the *lwr48* shapefile.

Notice that *lwr48.shp* also has the county and state FIPS codes, in the COUNTY and STATE columns, respectively. Each state has a unique FIPS code, and each county within a STATE has a unique code. If we combine the STATE-COUNTY codes, we can create a unique ID for each county in the country.

Add a column to the lwr48 data table (**Options > Add Field** in the "Attributes of lwr48" window).

Make this field a long integer, with at least 8 columns (precision), and name it something like sta_count.

Use the field calculator to assign sta_count a value according to the formula:

[STATE] * 10000 + [COUNTY]

Multiplying the STATE by 10000 and adding to COUNTY creates a unique 5-digit code, with the value for STATE in the first two digits, and the value for COUNTY in the next three digits.

Open the raw_corn_dat.dbf file, add a sta_count column similar to the one in lwr48, and create and value for a new column using the field calculator, according to:

[Stfips] *10000 + [CoFips]

Now, sort the raw_corn_dat table in ascending order, by right clicking/selecting in the sta_count column.

You should have a window that looks something like the figure at right:

Attributes of raw_corn_dat

OID	Stfips	CoFips	Harvested	Yield	Production	sta_count
28	1	1	2000	38	76000	10001
2649	1	1	1200	75	90000	10001
5250	1	1	900	33	30000	10001
7843	1	1	1600	76	121000	10001
38	1	3	11600	102	1180000	10003
2662	1	3	9200	99	914000	10003
5262	1	3	4600	68	315000	10003
7854	1	3	5300	112	593000	10003
48	1	5	4300	54	232000	10005
2672	1	5	3400	88	300000	10005
5272	1	5	1000	70	70000	10005
7864	1	5	2600	110	286000	10005
9	1	9	3400	94	320000	10009
2630	1	9	1800	94	169000	10009
5230	1	9	1800	83	150000	10009
7824	1	9	1800	119	214000	10009
2650	1	11	500	60	30000	10011
39	1	13	2600	68	177000	10013
2663	1	13	2900	77	223000	10013

Record: 0 Show: All Selected Records (0 out of

We can now see the second problem with this data set.

Note that there are multiply entries for sta_count, each state/county combination. This is because yield was reported at various levels for each county.

We must aggregate the rows before we join this table to the lwr48 shapefile. A join matches the rows by a key. If we don't somehow summarize the multiple rows that have the same sta_count value, then we can't be sure which will be chosen for the join – many to one joins are ambiguous.

Fortunately, ArcMap provides a tool for aggregating rows.

(Video: L8_3_summarize_rows.mov)