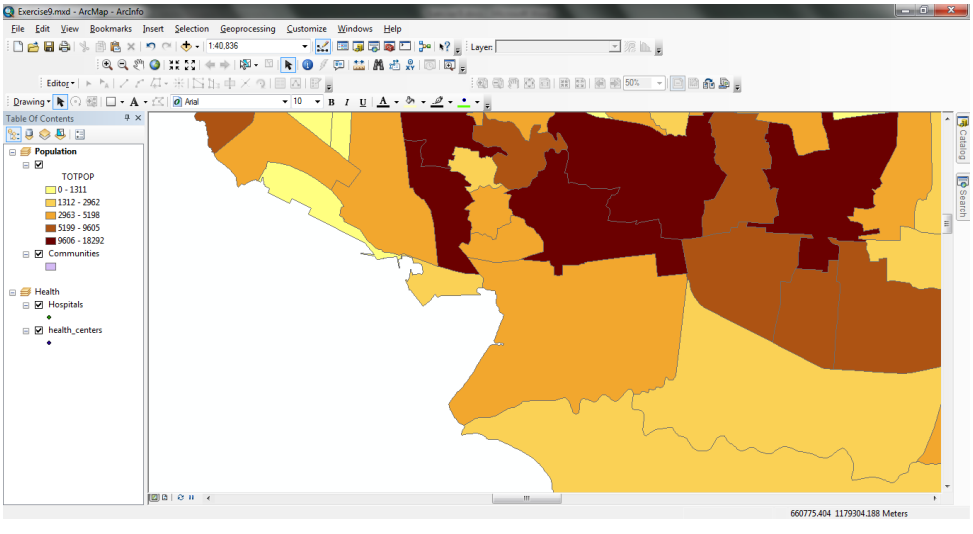
Exercise: Working with Layers

The ArcPy Mapping module, released with ArcGIS 10 provides a number of capabilities including the management of map documents and layers, printing, exporting, and ArcGIS Server publishing as well as map automation and the creation of PDF map books. This exercise will cover working with layers including adding and removing layers, and inserting, moving, and updating layers. At the end of this exercise you will have learned the following:

- How to add layer files (.lyr) to a data frame
- How to remove layer files (.lyr) from a data frame
- Use InsertLayer() to more specifically control the positioning of a layer added to a data frame
- Move layers to a new position within the data frame

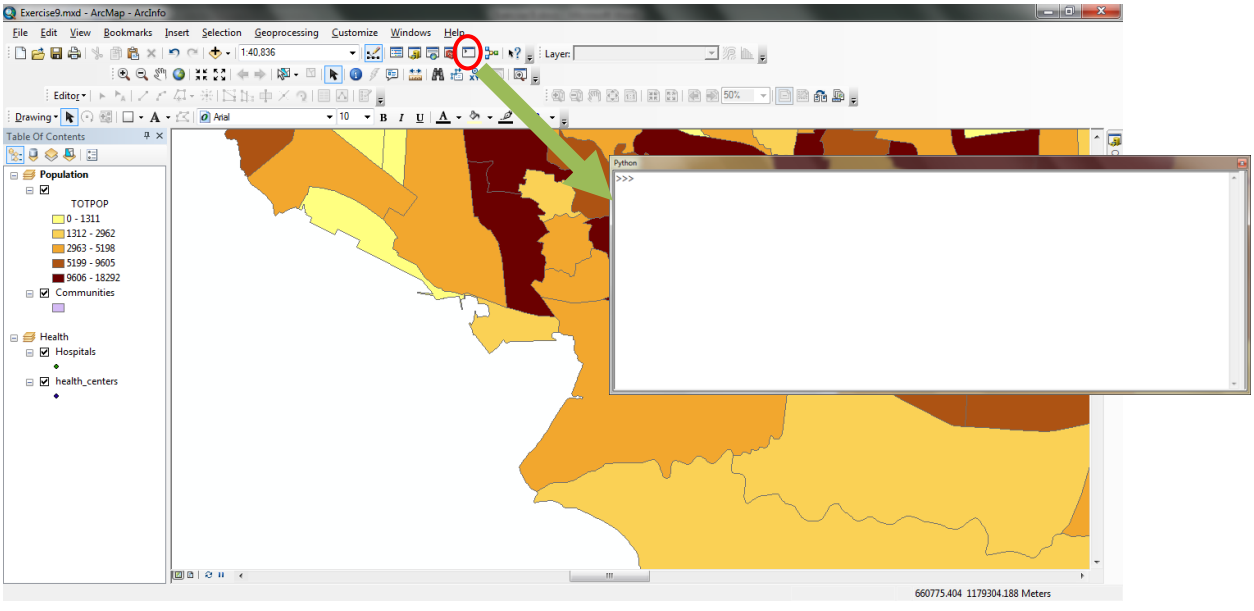
Step 1: Open the Crime.mxd Map Document file

- Start ArcMap and load Exercise9.mxd



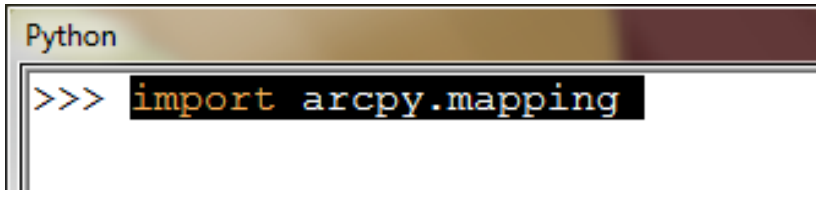
Step 2: Open the Python Window

- Click the Python Window button to open the editor.



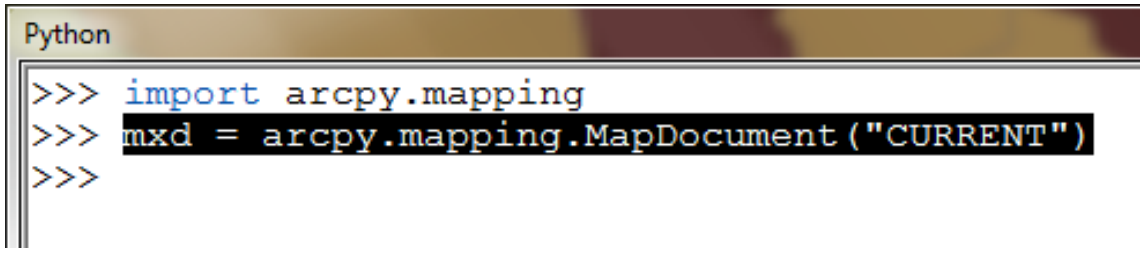
Step 3: Reference the ArcPy Module

- In the Python Window type the code you see below. This will reference the ArcPy Mapping module.



Step 4: Reference the Current Map Document File

- Add the code you see below which will reference the currently active map document file.



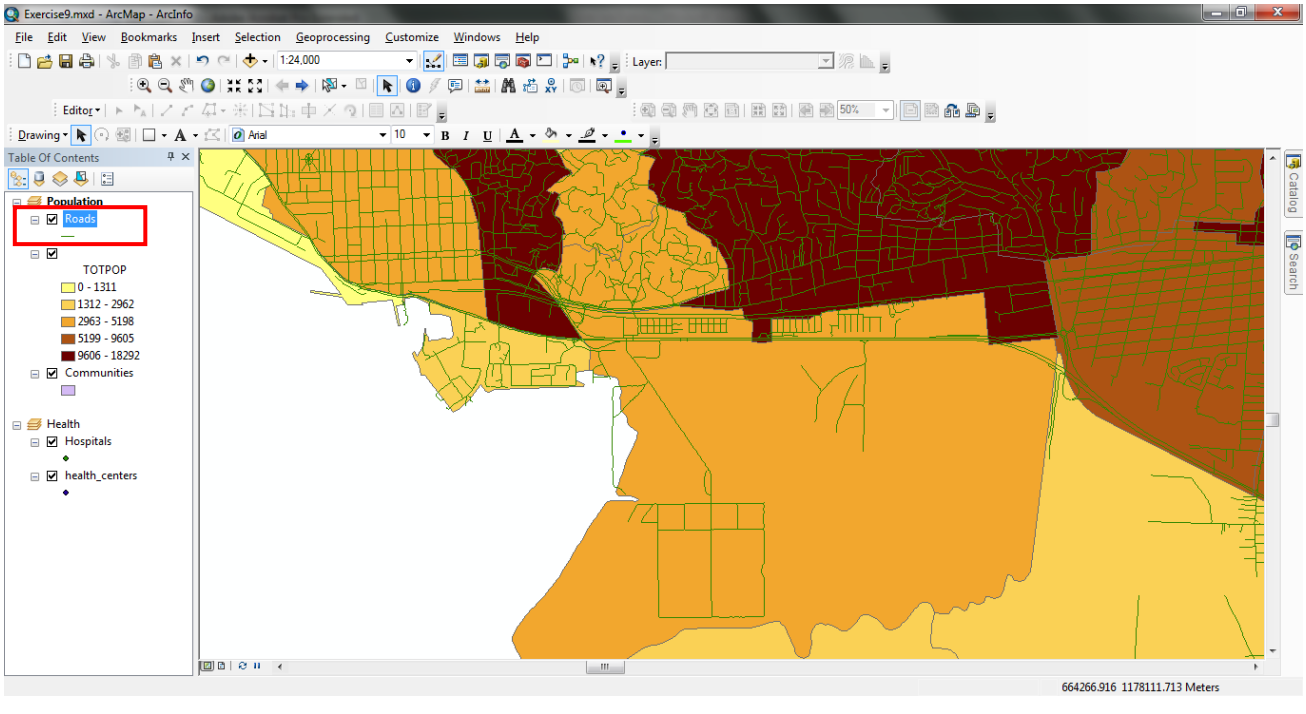
Step 5: Add a Layer File to the Population Data Frame

- Now that you have a reference to the current map document file let's generate a list of layers and print them out to the screen.
- Add the following code to the Python Window and then we'll discuss what each line accomplishes.

```
Python
>>> import arcpy.mapping
>>> mxd = arcpy.mapping.MapDocument("CURRENT")
>>> df = arcpy.mapping.ListDataFrames(mxd,"Population")[0]
>>> addLayer = arcpy.mapping.Layer(r"C:\Users\Me\Desktop\GIS Programming\Training\Data\Roads.lyr")
>>> arcpy.mapping.AddLayer(df,addLayer,"AUTO_ARRANGE")
```

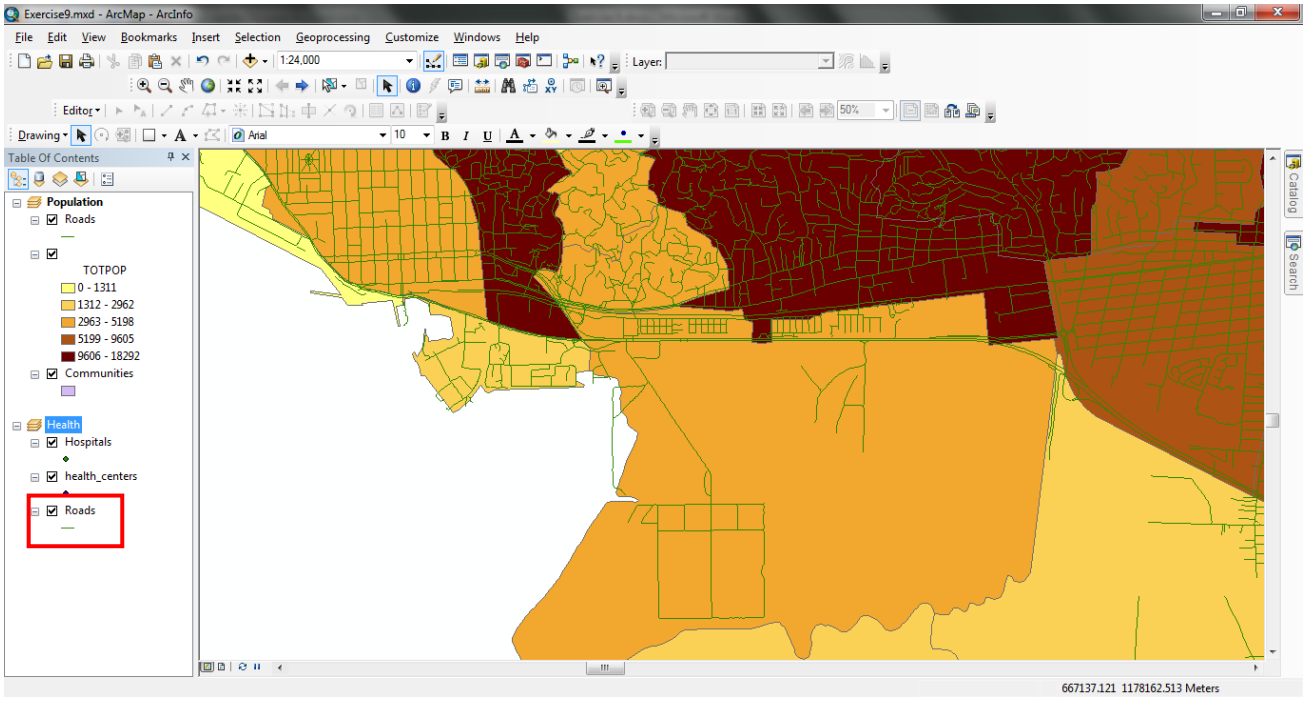
What we've done is first obtain a list of data frames filtered by the title "Population" which will return only a single data frame. Next, we create a new instance of the Layer object called "addLayer" which contains a reference to a layer file stored on disk in your data folder. Layers can be added from layer files, the same data frame, a different data frame, or a data frame in another map document file. Finally, we call the "AddLayer()" method passing in our data frame (df), the new Layer instance (addLayer) and specified that we want ArcMap to auto arrange the new layer.

- Run the code by hitting the enter key and you should see a new layer in the Population data frame in ArcMap as seen in the figure below.



Step 6: Removing Layers

In this next step you are going to learn how to remove layers from data frames in a map document file. You are going to remove the Roads Layer that you just added to the Population data frame along with any other Roads layer in any data frame of the map document. Follow the previous steps to also add the Roads layer to the Health data frame. Following this, you ArcMap window should now be looking similar the image below.

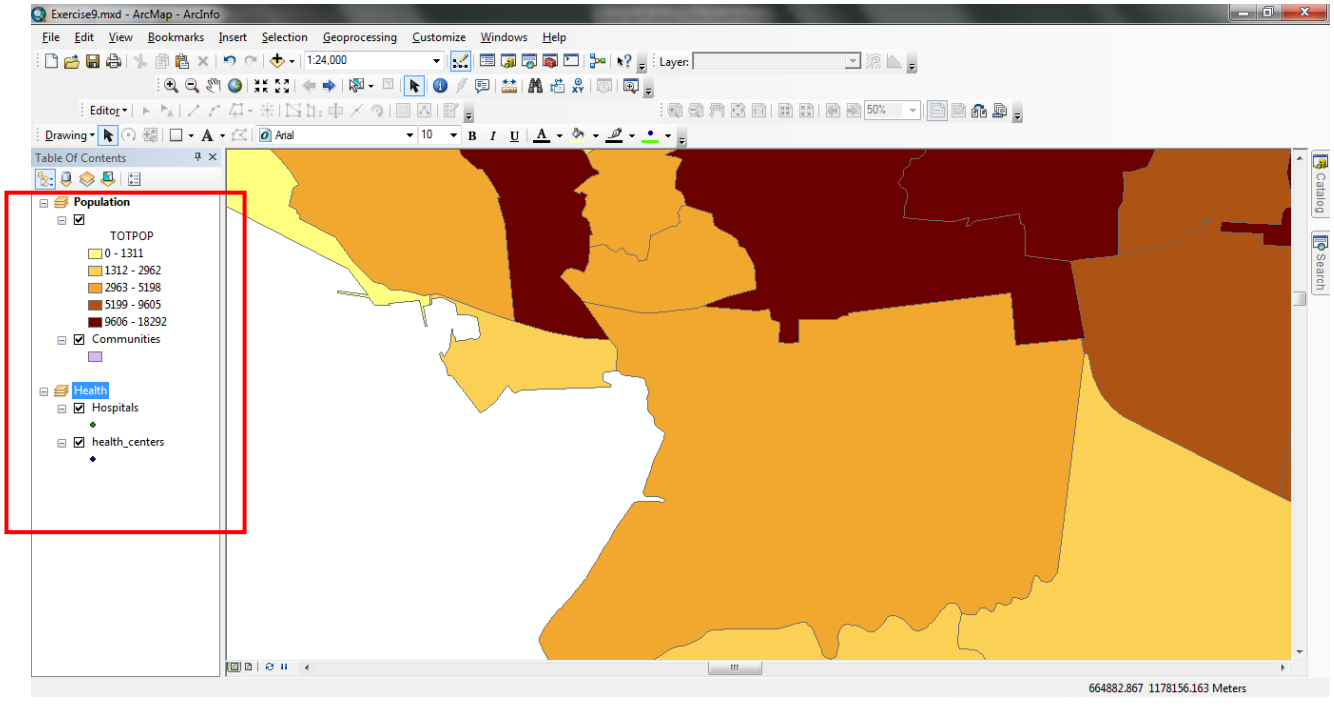


- Right click inside the Python Window and select “Clear All” which should result in the window being cleared of all code and output as seen below.

- Add the following code block which will loop through each data frame, obtain a list of all layers in the data frame, test for a layer name of “Roads” and remove the layer.

```
Python
>>> import arcpy.mapping
>>> mxd = arcpy.mapping.MapDocument("CURRENT")
>>> for df in arcpy.mapping.ListDataFrames(mxd):
...     for lyr in arcpy.mapping.ListLayers(mxd,"",df):
...         if lyr.name == "Roads":
...             arcpy.mapping.RemoveLayer(df,lyr)
```

- Running the code should remove the Roads layers in each of the data frames in the current map document.



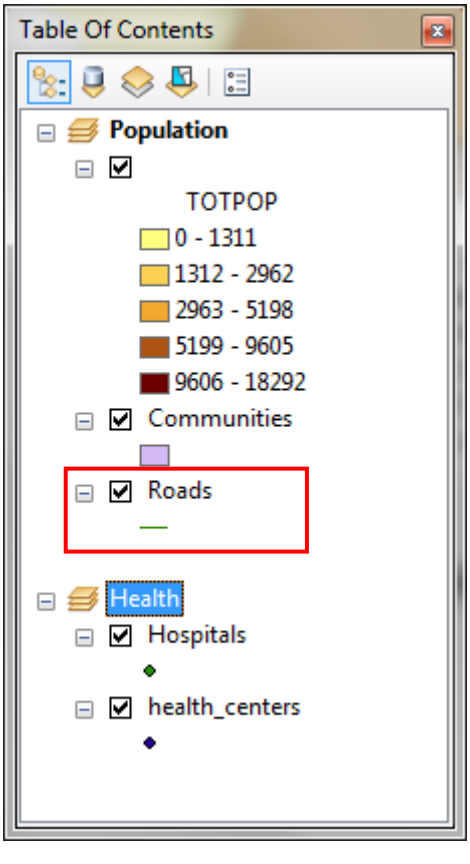
Step 7: Use InsertLayer for Precise Positioning of a Layer

The InsertLayer() method allows for more precise positioning of a new layer into a data frame or group layer. It uses a reference layer to specify a location and the layer is added either before or after the reference layer as specified in your code. Since InsertLayer() requires the use of a reference layer you can't use this method on an empty data frame. In this step you're going to add the Roads.lyr file back into the Population data frame using the Communities layer as the reference layer. You'll add the Roads file below the communities file.

- Clear the Python Window
- Add the following code block, run it, and then we'll discuss the code.

```
Python
>>> import arcpy.mapping
... mxd = arcpy.mapping.MapDocument("CURRENT")
... df = arcpy.mapping.ListDataFrames(mxd,"Population")[0]
... refLayer = arcpy.mapping.ListLayers(mxd,"Communities",df)[0]
... insertLayer = arcpy.mapping.Layer(r"C:\Users\Me\Desktop\GIS Programming
 \Training\Data\Roads.lyr")
... arcpy.mapping.InsertLayer(df,refLayer,insertLayer,"AFTER")
...
```

This script first gets the Population data frame (df), creates a new instance of the Layer class (refLayer) which points to the Communities layer. This layer will become our reference layer. Next, we create a second instance of the Layer class (insertLayer) which points to the Roads.lyr file which you've seen earlier in this exercise. Finally, we call the InsertLayer() method passing in the data frame (df), reference layer (reflayer), layer to insert (insertLayer), and specify that we want the insert layer to be added after the reference layer. After running the script you should see that the Roads.lyr file has been added as seen in the figure below.



Step 8: Moving Layers within a Data Frame

In this step you'll use the MoveLayer() method to reposition the Roads layer within the data frame.

- Clear the Python Window
- Type in the following block of code

```
Python
>>> import arcpy.mapping
... mxd = arcpy.mapping.MapDocument("CURRENT")
... df = arcpy.mapping.ListDataFrames(mxd,"Population")[0]
... for lyr in arcpy.mapping.ListLayers(mxd,"",df):
...     if lyr.name == "Roads":
...         moveLayer = lyr
...     if lyr.name == "Communities":
...         refLayer = lyr
... arcpy.mapping.MoveLayer(df,refLayer,moveLayer,"BEFORE")
...
```

- Run the code by hitting the ENTER key

This script gets the Population data frame (df), and loops through all the layers in the data frame. If the layer name is “Roads” it is assigned to the ‘moveLayer’ variable and if the layer name is “Communities” it is assigned to the ‘refLayer’ variable. Just as with InsertLayer(), MoveLayer() requires a reference layer. Finally, we call MoveLayer() passing in the data frame, reference layer, move layer, and instructions to place the move layer before the reference layer. Assuming you don’t have any errors when you run the script you should see that the Roads layer has been moved above the Communities layer.