Part 3
Carrying out specialized tasks

Exercise 10

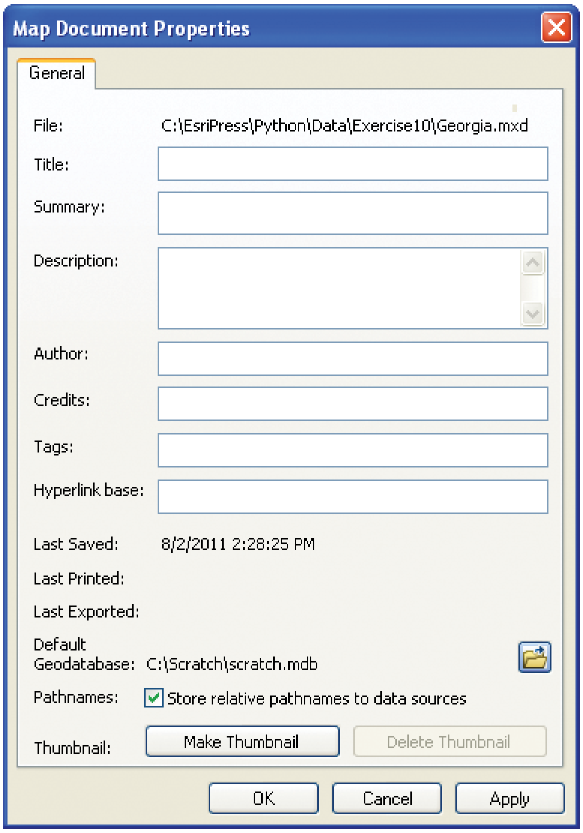
Map scripting

Open and save a map document

Map scripting can be used to open a map document and change any of the map document properties.

1 **Start ArcMap. Open the map document C:\EsriPress\Python\Data\Exercise10\Georgia.mxd.**

2 **On the ArcMap menu bar, click File > Map Document Properties.** Notice that most of the properties are blank, as shown in the figure. ➔



File: C:\EsriPress\Python\Data\Exercise10\Georgia.mxd

Last Saved: 8/2/2011 2:28:25 PM

Default Geodatabase: C:\Scratch\scratch.mdb

Pathnames: ☑ Store relative pathnames to data sources

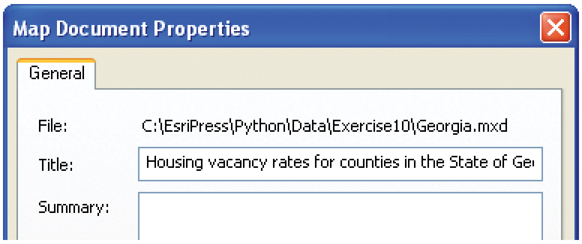
3 **Click OK to close the Map Document Properties dialog box.**

4 **On the Standard toolbar, click the Python button to open the Python window.**

5 Enter and run the following code:

```
    >>> mapdoc = arcpy.mapping.MapDocument("CURRENT")
    >>> mapdoc.title = "Housing vacancy rates for counties in the State of ➜
  ➜ Georgia, 2000"
    >>> mapdoc.save()
    >>> del mapdoc
```

6 On the ArcMap menu bar, click File > Map Document Properties.
Notice that the Title property has been modified, as shown in the figure.



File: C:\EsriPress\Python\Data\Exercise10\Georgia.mxd

Title: Housing vacancy rates for counties in the State of Ge

Summary:

7 Close ArcMap.

Work with data frames

Map scripting can be used to access and modify the properties of one or more data frames in a map document. In the first example, you will read the names of all the data frames.

1 Start PythonWin. Create a new Python script and save as dflist.py **to the C:\EsriPress\Python\Data\Exercise10\Results folder.**

2 Enter the following code:

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Austin_TX.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
listdf = arcpy.mapping.ListDataFrames(mapdoc)
for df in listdf:
    print df.name
del mapdoc
del listdf
```

3 Save and run the script.

Running the script prints the names of three data frames to the Interactive Window:

```
Facilities
Street Trees
Parks
```

You can also modify the data frame properties. In the next example, you will modify selected properties so they are identical for all data frames in the map document.

4 Start ArcMap. Open the map document C:\EsriPress\Python\Data\Exercise10\Austin_TX.mxd.

5 Briefly examine the layers and the properties of each data frame.

Notice that the extent is different for each, and the coordinate system for the Parks data frame is different from the other two. Next, you will use scripting to modify these properties and make them consistent.

6 Close ArcMap.

7 In PythonWin, create a new Python script and save as dfproperties.py **to the Results folder for exercise 10.**

8 Enter the following code:

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Austin_TX.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
dataset = "C:/EsriPress/Python/Data/Exercise10/Austin/base.shp"
spatialref = arcpy.Describe(dataset).spatialReference
extent = arcpy.Describe(dataset).extent
for df in arcpy.mapping.ListDataFrames(mapdoc):
    df.spatialReference = spatialref
    df.panToExtent(extent)
    df.scale = 15000
mapdoc.save()
del mapdoc
```

9 Save and run the script. Running this script sets the spatial reference, extent, and scale for all the data frames in the map document.

10 Start ArcMap. Open Austin_TX.mxd.

11 Confirm that the data frame properties have been modified.

Work with map layers

Map scripting can also be used to work with map layers. In the next example, you will create a list of all the layers in a map document and modify the properties of a specific layer.

1 In PythonWin, create a new Python script and save as maplayers.py **to the Results folder for exercise 10.**

2 Enter the following code:

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Austin_TX.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
for df in arcpy.mapping.ListDataFrames(mapdoc):
    print "Data frame " + df.name + " contains the following layers:"
    lyrlist = arcpy.mapping.ListLayers(mapdoc, "", df)
    for lyr in lyrlist:
        print lyr.name
del mapdoc
```

3 Save and run the script.

This prints the name of each data frame followed by the layers in each data frame:

```
Data frame Facilities contains the following layers:
addresses
facilities
sidewalks
base
Data frame Street Trees contains the following layers:
sidewalks
trees
buildings
base
Data frame Parks contains the following layers:
parks
base
```

Next, you will modify the properties of a specific layer in the map document.

4 **In ArcMap, make sure Austin_TX.mxd is still open. Activate the Parks data frame, notice how the labels of the parks layer are turned off, and turn off the parks layer.**

5 **Save Austin_TX.mxd.**

6 **Close ArcMap.**

7 **In PythonWin, create a new Python script and save as** lyrproperties.py **to the Results folder for exercise 10.**

8 **Enter the following code:**

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Austin_TX.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
lyrlist = arcpy.mapping.ListLayers(mapdoc)
for lyr in lyrlist:
    if lyr.name == "parks":
        print lyr.name
        lyr.visible = True
        lyr.showLabels = True
mapdoc.save()
del mapdoc
del lyrlist
```

9 **Save and run the script.**

10 **Start ArcMap. Open Austin_TX.mxd.**

11 **Confirm that the properties of the parks layer have been modified.**

12 **Close ArcMap.**

Work with page layout elements

Map scripting also makes it possible to work with page layout elements. In the next examples, you will create a list of all the elements of a page layout and modify one of the elements.

1 **In PythonWin, create a new Python script and save as** elemlist.py **to the Results folder for exercise 10.**

2. **Enter the following code:**

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Georgia.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
elemlist = arcpy.mapping.ListLayoutElements(mapdoc)
for elem in elemlist:
    print elem.name + " " + elem.type
del mapdoc
```

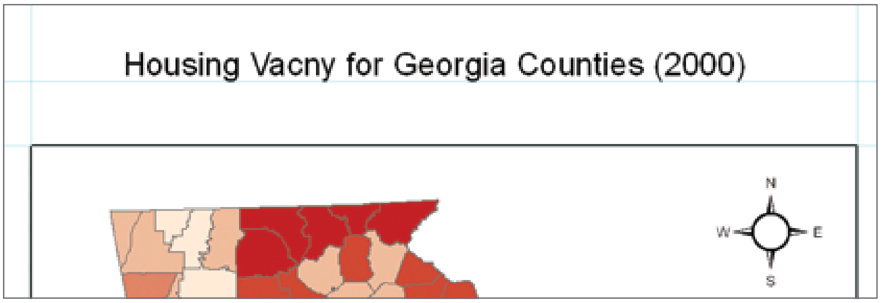
3. **Save and run the script.**

This prints the name and type of each page layout element, as follows:

```
Title TEXT_ELEMENT
Stepped Scale Line MAPSURROUND_ELEMENT
North Arrow MAPSURROUND_ELEMENT
Legend LEGEND_ELEMENT
Vacancy DATAFRAME_ELEMENT
```

Next, you will modify one of the elements.

4. **Start ArcMap. Open Georgia.mxd.** Notice that there is a typo in the title where the word "vacancy" is misspelled, as shown in the example in the figure.



Housing Vacny for Georgia Counties (2000)

5. **Close ArcMap.**

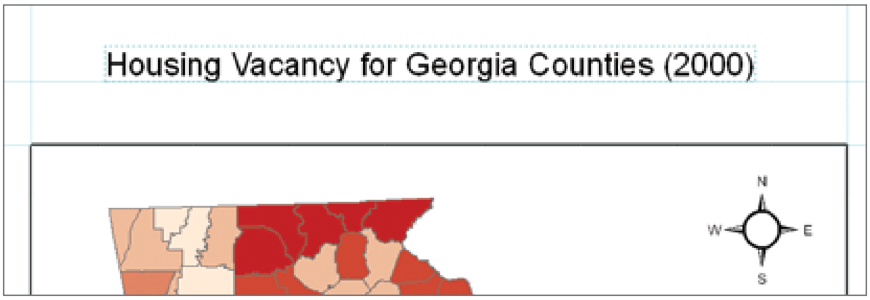
6. **In PythonWin, create a new Python script and save as** elemproperties.py **to the Results folder for exercise 10.**

7 Enter the following code:

```
import arcpy
mxd = "C:/EsriPress/Python/Data/Exercise10/Georgia.mxd"
mapdoc = arcpy.mapping.MapDocument(mxd)
elemlist = arcpy.mapping.ListLayoutElements(mapdoc)
title = elemlist[0]
title.text = "Housing Vacancy for Georgia Counties (2000)"
mapdoc.save()
del mapdoc
```

8 Save and run the script.

9 Start ArcMap. Open Georgia.mxd. Notice that the typo in the title has been corrected, as shown in the example in the figure.



Challenge exercise

Challenge 1

In ArcGIS Desktop Help, research the `AddLayer` function of the ArcPy mapping module and use it to write a script that adds the parks layer from the Parks data frame in Austin_TX.mxd to the other two data frames in the same map document.