

Exercise: Describing GIS Data

The Describe function provides you with the ability to get basic information about datasets. These datasets could include Feature Classes, Tables, Coverages, Layers, Workspaces, Rasters, and others. In this exercise you will use the Describe function in a Python script to return descriptive information about the GIS data in your project. At the end of this exercise you will have learned the following:

- How to use the Describe function to return information about a Feature Class
- Access Table and Dataset property groups associated with GIS data
- Access SpatialReference object to find coordinate system information

Step 1: Create a Describe Feature Class Script

In this step you will create a Python script that describes a feature class.

- Open PythonWin and create a new script.
- Insert comments including your name, date, and the purpose of this script.
- Save the script to a location of your choice with the name DescribeFC.py.
- Import the ArcPy module

```
import arcpy
```

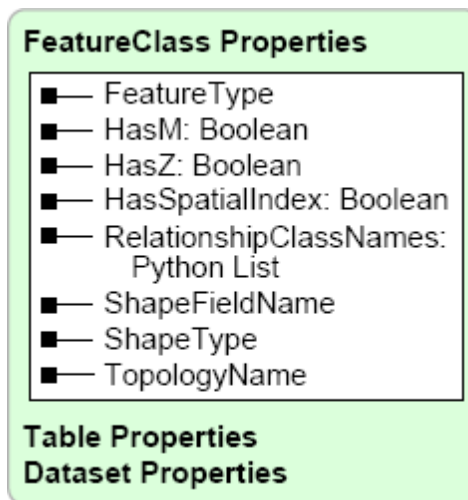
- Create a try block and set the workspace to a location of your choice

```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
```

- Call the Describe function on the Hospitals shapefile found in this workspace.

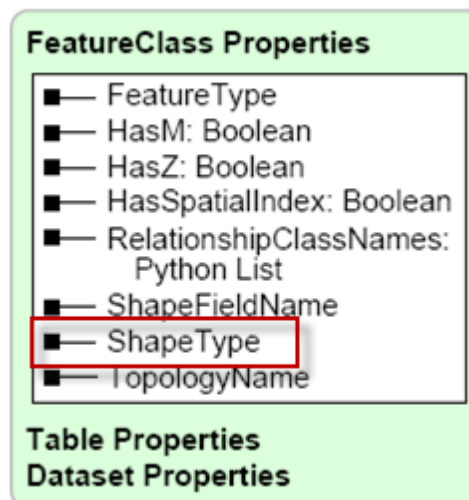
```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
    descFC = arcpy.Describe("Hospitals.shp")
```

In this case the Describe function will return a FeatureClass property group since we are describing a shapefile. You can see the properties in this group in the object model diagram below.

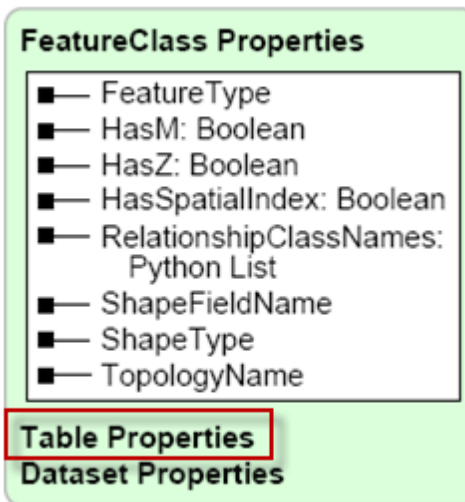


- Print the shape type (point, line, polygon) of the streets shapefile

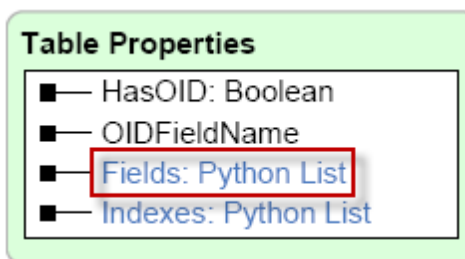
```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
    descFC = arcpy.Describe("Hospitals.shp")
    print "The shape type is: " + descFC.ShapeType
```



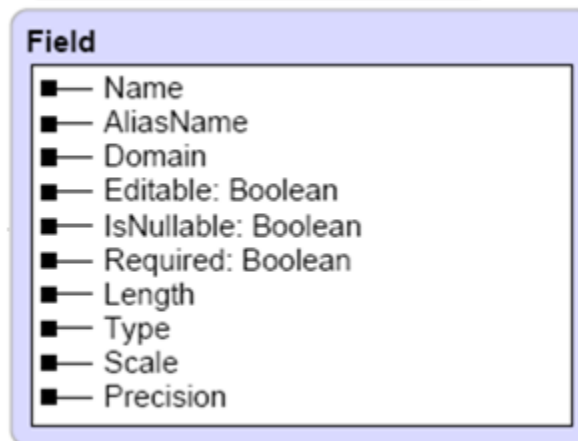
- Now we'll examine the fields associated with the Hospitals feature class. In addition to returning a FeatureClass property group, you also have access to a Table properties group which in this case will contain the fields associated with the feature class.



The Fields property on Table Properties returns a Python list containing one 'Field' object for each column/field in the feature class or table.



Each Field has a number of read only properties including the name, alias, length, type, scale, precision, and others as seen below.



- In this script we are going to print out the field name, type, and length as seen below. Note the use of a Python 'for' loop to process each 'Field' in the Python list. Add the follow code:

```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
    descFC = arcpy.Describe("Hospitals.shp")
    print "The shape type is: " + descFC.ShapeType
    flds = descFC.Fields
    for fld in flds:
        print "Field: " + fld.Name
        print "Type: " + fld.Type
        print "Length: " + str(fld.Length)
```

- Next, we'll print out the geographic extent of the layer through the use of the Extent object, returned by the Extent property on the Dataset property group which is also available through the Feature Class property group.

```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
    descFC = arcpy.Describe("Hospitals.shp")
    print "The shape type is: " + descFC.ShapeType
    flds = descFC.Fields
    for fld in flds:
        print "Field: " + fld.Name
        print "Type: " + fld.Type
        print "Length: " + str(fld.Length)
    ext = descFC.Extent
    print "XMin: %f" % (ext.xmin)
    print "YMin: %f" % (ext.ymin)
    print "XMax: %f" % (ext.xmax)
    print "YMax: %f" % (ext.ymax)
```

- Finally, add in the *except* statement to print out any error message.

```
import arcpy
try:
    arcpy.env.workspace = r"C:\Users\Me\Desktop\GIS Programming\Training\Data"
    descFC = arcpy.Describe("Hospitals.shp")
    print "The shape type is: " + descFC.ShapeType
    flds = descFC.Fields
    for fld in flds:
        print "Field: " + fld.Name
        print "Type: " + fld.Type
        print "Length: " + str(fld.Length)
    ext = descFC.Extent
    print "XMin: %f" % (ext.xmin)
    print "YMin: %f" % (ext.ymin)
    print "XMax: %f" % (ext.xmax)
    print "YMax: %f" % (ext.ymax)
except:
    print arcpy.GetMessages()
```

- When you run the script you should see output similar to what you see below.

```
[Dbg]>>> The shape type is: Point
Field: FID
Type: OID
Length: 4
Field: Shape
Type: Geometry
Length: 0
Field: FID_
Type: Integer
Length: 9
Field: AREA
Type: Double
Length: 13
Field: PERIMETER
Type: Double
Length: 13
Field: CARE_
Type: Double
Length: 10
Field: CARE_ID
Type: Double
Length: 10
Field: NAME
Type: String
Length: 18
Field: ADDRESS
Type: String
Length: 40
Field: NO_OF_BEDS
Type: SmallInteger
Length: 2
Field: NO DOCTORS
```

Task:

Write Python scripting to print the spatial reference and extent for a raster dataset

HINT: Remember dataset properties