Chapter 11: Working with Lists of GIS Data

- Many geoprocessing tasks first require that you perform an initial gathering of data before getting to the main body of the geoprocessing tasks
- This is often accomplished through one or more list methods found in Arcpy
- Python provides you with the ability to batch process data through scripting. This helps you automate workflows and increases the efficiency of you processing
- For example, you may need to iterate through all feature classes on disk and perform a specific action on each dataset
- Arcpy provides a number of functions that can be used to iterate through lists of data e.g. ListFeatureClasses()
- These methods work on many different types of GIS data
- Functions provide for getting lists of fields, indexes, datasets, feature classes,…

ListFields(dataset, wild_card, field_type)	Returns a list of fields found in the input value
ListIndexes(dataset, wild_card)	Returns a list of attribute indexes found in the input value
ListDatasets(wild_card, feature_type)	Returns the datasets in the current workspace
ListFeatureClasses(wild_card, feature_type)	Returns the feature classes in the current workspace
ListFiles(wild_card)	Returns the files in the current workspace
ListRasters(wild_card, raster_type)	Returns a list of rasters found in the current workspace
ListTables(wild_card, table_type)	Returns a list of tables found in the current workspace
ListWorkspaces(wild_card, workspace_type)	Returns a list of workspaces found in the current workspace
ListVersions(sde_workspace)	Returns a list of versions the connected user has permission to use

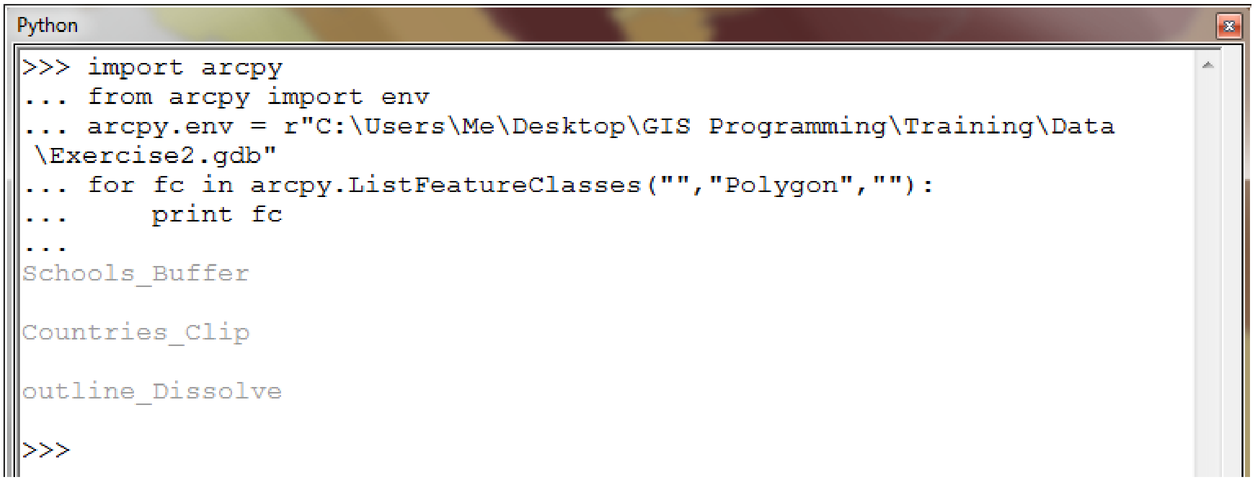
- The result of each of these functions is a Python list, which is a list of values.
- A list in scripting can contain any type of data, such as a string, which could be, for example, a path to a dataset, field, or row from a table.
- Once the list has been created with the values you want, you can loop through it in your script to work with each individual value.
- YOU MUST always REMEMBER to set your workspace before calling any list function otherwise the function would not know which dataset the information is coming from.
- Some list parameters are similar
- Some require input dataset
    - ListFields
    - ListIndexes
- The rest do not require an input dataset

    - They typically create lists of data based on a workspace
- All methods have the ability to filter using a wildcard and a type delimiter

Example:

ListFeatureClasses ({wild_card}, {feature_type}, {feature_dataset})

List all Polygon layer in the specified geodatabase

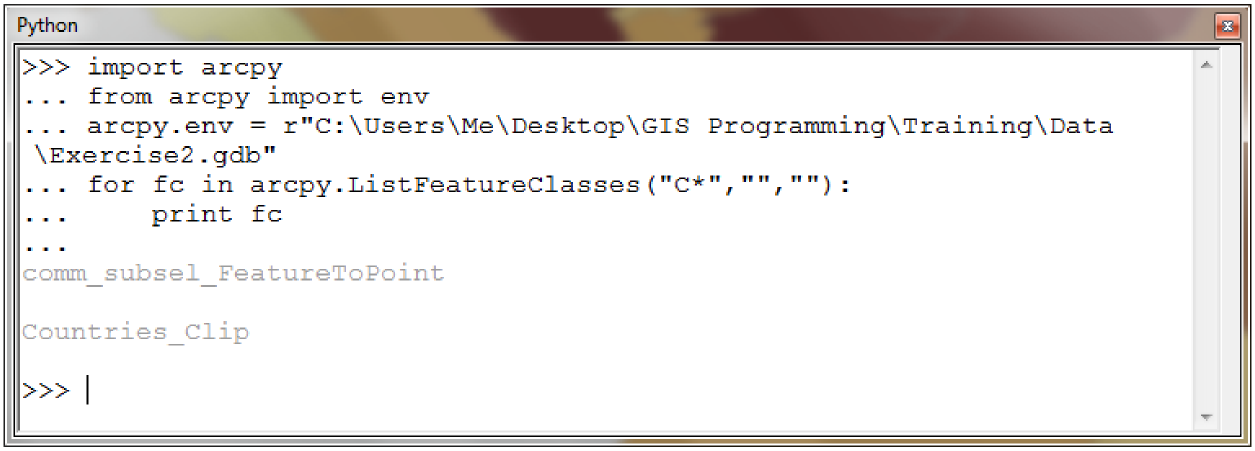


```
>>> import arcpy
... from arcpy import env
... arcpy.env = r"C:\Users\Me\Desktop\GIS Programming\Training\Data
 \Exercise2.gdb"
... for fc in arcpy.ListFeatureClasses("","Polygon",""):
...     print fc
...
Schools_Buffer

Countries_Clip

outline_Dissolve

>>>
```

- http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#/ListFeatureClasses/000v0000001n000000/

Example

Using a wildcard to select all feature classes that start with the name 'C'

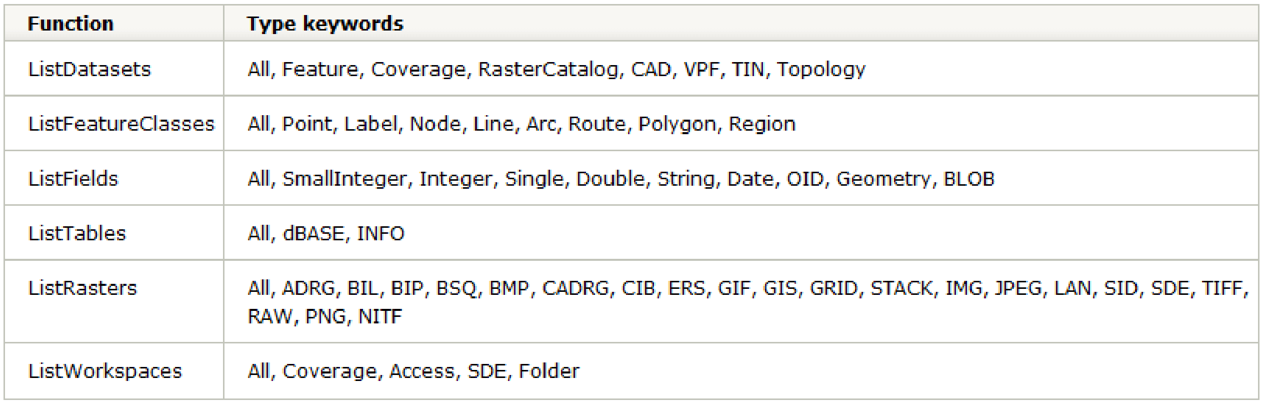


```
>>> import arcpy
... from arcpy import env
... arcpy.env = r"C:\Users\Me\Desktop\GIS Programming\Training\Data
 \Exercise2.gdb"
... for fc in arcpy.ListFeatureClasses("C*","",""):
...     print fc
...
comm_subsel_FeatureToPoint

Countries_Clip

>>> |
```

Task:

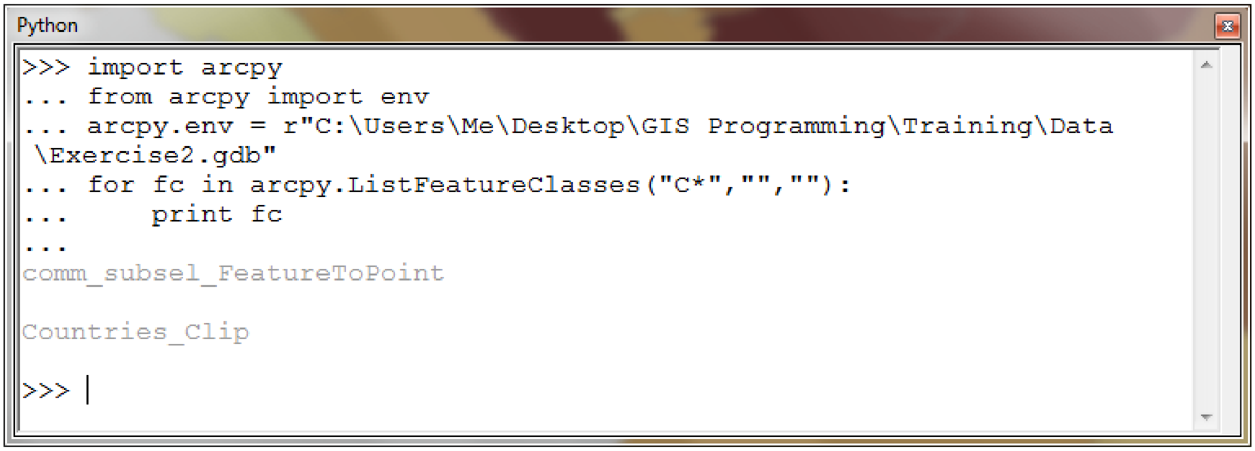
Write code to filter all feature classes in a geodatabase that start with "C" and are of Polygon Type

- Type can also be used to filter output



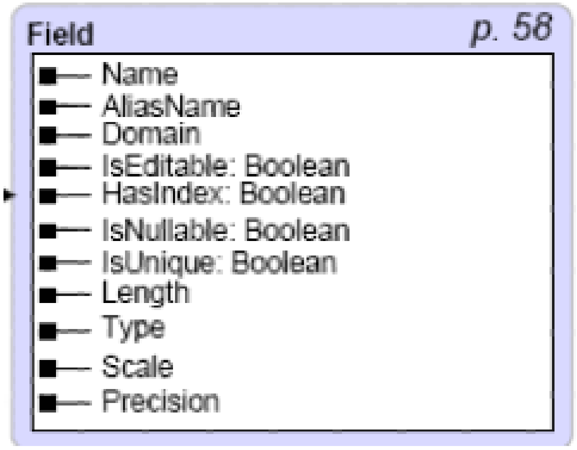
Function	Type keywords
ListDatasets	All, Feature, Coverage, RasterCatalog, CAD, VPF, TIN, Topology
ListFeatureClasses	All, Point, Label, Node, Line, Arc, Route, Polygon, Region
ListFields	All, SmallInteger, Integer, Single, Double, String, Date, OID, Geometry, BLOB
ListTables	All, dBASE, INFO
ListRasters	All, ADRG, BIL, BIP, BSQ, BMP, CADRG, CIB, ERS, GIF, GIS, GRID, STACK, IMG, JPEG, LAN, SID, SDE, TIFF, RAW, PNG, NITF
ListWorkspaces	All, Coverage, Access, SDE, Folder

- Once you have generated the list, you then typically proceed to iterate through the list and perform some type of geoprocessing
- For…loops in Python are often used for this purpose
- You may want to:
    - Display contents
    - Retrieve the number of items
    - Retrieve items by index number
    - Append items,…



```
>>> import arcpy
... from arcpy import env
... arcpy.env = r"C:\Users\Me\Desktop\GIS Programming\Training\Data
 \Exercise2.gdb"
... for fc in arcpy.ListFeatureClasses("C*","",""):
...     print fc
...
comm_subsel_FeatureToPoint

Countries_Clip

>>> |
```

- The ListFields function lists the fields in a feature class, shapefile, or table
- ListFields returns a list of field objects
- Field objects contain various read only properties



- The ListTables function returns a list of tables in a workspace
- Table types can be DBASE,INFO or ALL
- Returned values are outputted as strings