

A PYTHON PRIMER FOR THE GIS PROFESSIONAL

Lee Frederick
City of Milwaukee ITMD-GIS

A Python Primer for the GIS Professional

Topics

- What is a script? Why automate a task?
- What is Python? What can Python do? Why use it?
- What is ArcPy?
- How do you write and run a Python script?
- Tips to get started writing your own Python scripts
- Online resources

A Python Primer for the GIS Professional

What is a script?

- a set of instructions, or **code**, written in a human-readable language that tells a piece of software what to do
- instructions are converted by an **interpreter** into a machine-readable language that the software can understand
- scripts are typically used to **automate a task or workflow**

A Python Primer for the GIS Professional

Why automate a task with a script?

- a task is repeated on a regular or semi-regular basis
- the results of a task must be exactly the same each time the task is performed
- a task must be performed outside of normal working hours

A Python Primer for the GIS Professional

Why automate a task with a script? scripts can...

- provide detailed documentation about a task's inputs, processes, and outputs
- be shared or reused for other similar tasks
- minimize or eliminate input errors
- boost productivity: free up time to do other things...

A Python Primer for the GIS Professional

What is Python?

- an object-oriented programming language that is open source & non-proprietary:
 - unrestricted use & distribution
 - not tied to a single OS, platform, application, or vendor
- also referred to as a dynamic or **scripting** language
- part of the default ArcGIS installation: if you are an ArcGIS Desktop 10.X user, you probably already have Python and **ArcPy** at your disposal

A Python Primer for the GIS Professional

Why use Python?

- popularity: large community of users offers a strong base of documentation & support
- access ArcGIS functionality along with built-in & third-party Python functionality:
 - combine both GIS & non-GIS processes into a complete scripted workflow or task
 - single programming language
 - single development environment
- because Esri said so...

A Python Primer for the GIS Professional

What can Python do?

- run (almost) any ArcGIS tool
- built-in functionality, as well as thousands of available third-party site packages, allows Python to do (almost) anything else that you can imagine...

*A Python Primer
for the GIS Professional*

What can Python do? General computing tasks:

- copy, move, rename, or delete files
- write to or read from files
- compose & send emails
- compress files

A Python Primer for the GIS Professional

What can Python do? ETL tasks:

- 'extract-transform-load'
- download and upload files
- import and export files & database objects
- 'scrub' data (string manipulation)
- QA/QC
- modify schemas
- convert from one data storage format to another

*A Python Primer
for the GIS Professional*

What can Python do? GIS tasks:

- general geoprocessing
- general data management
- spatial analysis
- refresh data in publication environments
- automate production of maps
- run Modelbuilder models (or replace models altogether)
- geodatabase administration

A Python Primer for the GIS Professional

What is ArcPy?

- An ArcGIS-specific **site package** developed by Esri
- primary role is to extend Desktop functionality into a Python scripting environment
- when imported as a **module** into a Python script, ArcPy provides access to ArcGIS extensions and ArcToolbox tools for Python scripting
- (almost) anything that can be done ‘manually’ in ArcMap or ArcCatalog can be performed via ArcPy

A Python Primer for the GIS Professional

Where do you write a Python script? IDEs:

- ‘integrated development environment’
- a single interface where code can be written, tested, edited, and executed
- provides language-specific color coding, debugging, syntax & formatting help, etc.
- examples:
 - **PyScripter**
 - IDLE (included with ArcGIS)
 - Python Tools for Visual Studio

▪ many others...

A Python Primer for the GIS Professional

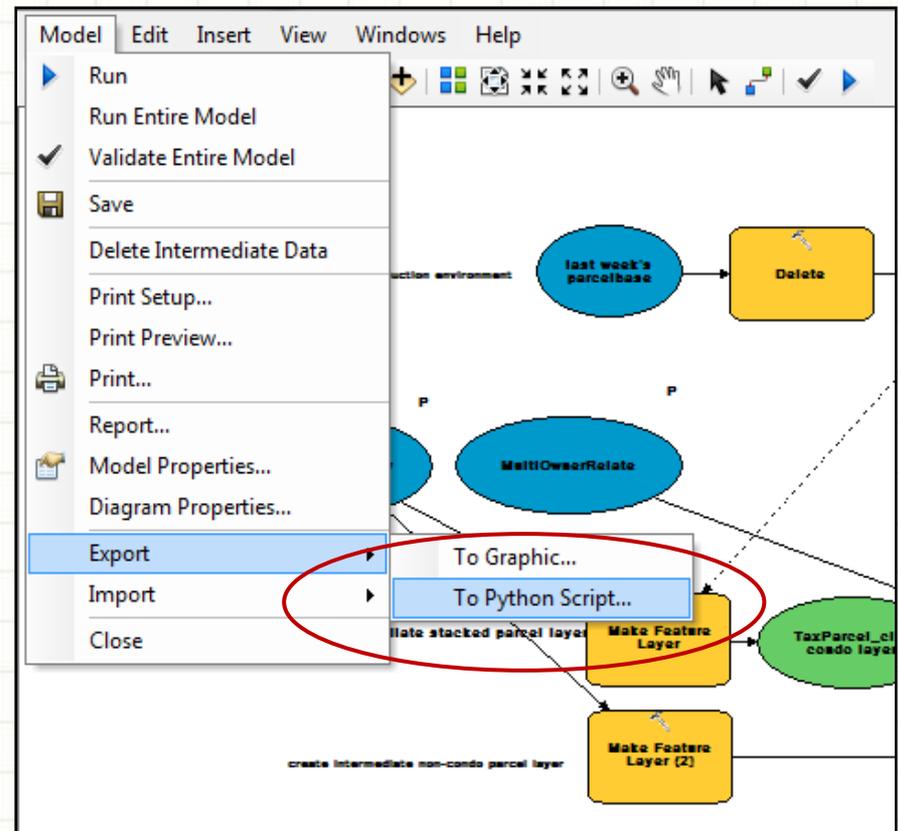
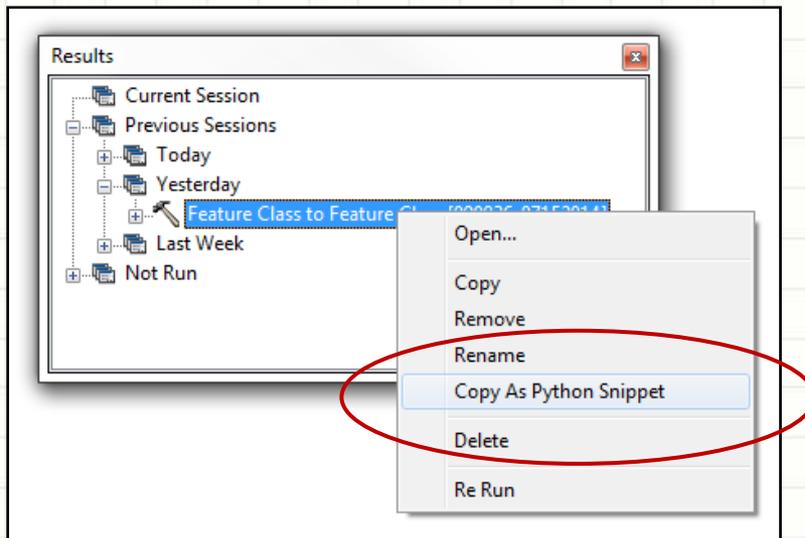
Where do you write a Python script? code editors:

- language-specific color coding, syntax & formatting help, etc.
- examples:
 - **Notepad ++**
 - EditPad Lite
 - many others...

A Python Primer for the GIS Professional

How do you write a Python script? Get code from...

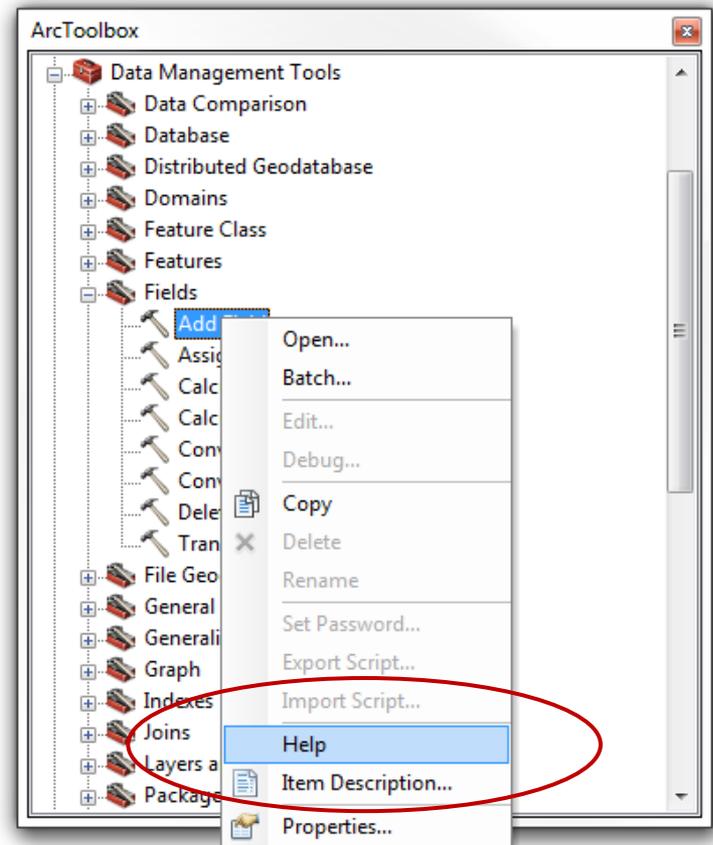
- geoprocessing results after running an ArcToolbox tool
- Modelbuilder export



A Python Primer for the GIS Professional

How do you write a Python script? Get code from...

- ArcToolbox documentation
- online developer communities



A Python Primer for the GIS Professional

How do you write a Python script? The basics...

- gain access to the ArcPy **module** using the **import** statement:

```
import arcpy
```

- call ArcPy **classes** or **functions** (tools) from the module

```
import arcpy
```

```
arcpy.FeatureClassToFeatureClass_conversion(arguments )
```

- be sure to include all input **arguments** or **parameters** that the tool expects, in the expected order

A Python Primer for the GIS Professional

How do you write a Python script? The basics...

- pay close attention to syntax:

- case sensitivity

`arcpy.FeatureClassToFeatureClass_conversion(arguments)`

is **NOT** the same as...

`Arcpy.FeatureclassToFeatureclass_conversion(arguments)`

- **strings** vs. **variable names** and **statements**

`"my_file_gdb"` and `'import'`

are **NOT** the same as...

`my_file_gdb` and `import`

- save your script text file with a **.py** extension

A Python Primer for the GIS Professional

How do you write a Python script? A basic GIS example:

```
# Imports shapefile features into a new file geodatabase
# Arguments:
IN_DIR = "C:\\Users\\lfrede\\Desktop"
IN_SHP = IN_DIR + "\\my_shapefile.shp"
OUT_DIR = IN_DIR
OUT_GDB_NAME = "my_file_gdb"
OUT_GDB = OUT_DIR + "\\OUT_GDB_NAME + ".gdb"
OUT_FC_NAME = "my_fc"
# Processes:
# import module(s)
import arcpy
# create a new file geodatabase
arcpy.CreateFileGDB_management( OUT_DIR, OUT_GDB_NAME, "" )
# create a feature class from input shapefile features in new file geodatabase
arcpy.FeatureClassToFeatureClass_conversion( IN_SHP, OUT_GDB, OUT_FC_NAME, "", "", "" )
```


A Python Primer for the GIS Professional

How do you run a Python script?

- from Windows Explorer (double-click on .py file)
- within a custom ArcGIS toolbox as a 'script tool'
- Task Scheduler

A Python Primer for the GIS Professional

Get started: Just jump into it!

- become familiar with Python & ArcPy basics within the context of your own work using your own data
- identify a task or workflow that involves GIS data and break it down into smaller steps
- develop & test short, simple practice scripts (one script per step or group of steps) using small 'disposable' input & output datasets
- gradually become more comfortable and confident by building on small 'victories'

A Python Primer for the GIS Professional

Get started: trial & error...

- make mistakes and learn from the error messages returned by Python
- learn from other Python users:
 - search the web for error messages to see how other users reached a solution
 - find code on the web and adapt it to your own needs without guilt, but give credit where credit is due

*A Python Primer
for the GIS Professional*

**Get started: develop scripts with portability,
reusability & modularity in mind**

- Refined, organized, and self-explanatory scripts are easily...
 - read and understood
 - shared
 - repurposed
 - integrated into other scripted tasks
- Refine: replace repeated argument strings with ‘constants’ & ‘variables’ that represent arguments

A Python Primer for the GIS Professional

Get started: develop scripts with portability, reusability & modularity in mind

- Organize: divide code into ordered & labeled sections for readability & ‘transparency’

Who/what/when/why/how

Input argument constants

Functions

Input argument variables

Processes

- Explain: what does your code do?

explain your code...

...using brief comments...

...throughout your script

A Python Primer for the GIS Professional

Online resources: Esri

- [ArcGIS Desktop overview of Python](#)
- [ArcGIS Desktop overview of ArcPy](#)
- [ArcGIS Desktop geoprocessing tool references](#)
- [ArcGIS Python Recipes](#)
- [ArcGIS Resources: Python](#)
- [Free Esri Training web course: Python for Everyone](#)
- [Free Esri Training web course: Using Python in ArcGIS Desktop](#)
- [GIS developer Python discussions at GeoNet, The Esri Community](#)

A Python Primer for the GIS Professional

Online resources: Python.org

- [Python Beginner's Guide for Non-Programmers](#)
- [Python Tutorial](#)
- [Python Style Guide](#)
- [Python Documentation](#)
- [Python Language Reference](#)
- [Python Standard Library](#)
- [Python Package Index](#)

A Python Primer for the GIS Professional

Online resources: developer communities

- [GIS + Python discussions at Stack Exchange](#)
- [GIS + Python discussions at Stack Overflow](#)

*A Python Primer
for the GIS Professional*

Questions?