

Python Programming for Arcgis 1

Daniel Sheehan

dsheehan@mit.edu, gishelp@mit.edu

9:30-12:30

January 31, 2013

This class was originally developed by David Quinn and taught by David and Daniel in IAP 2010 and 2011.

Outline

- Introduction to Python and Arcgis
- Programming Principles and Modules
- Model Builder
- Read and Writing data

Python

Python is a language that lets you work more quickly and integrate your systems more effectively ¹

Documentation at <http://docs.python.org> and look for Python 2.7 (used in Arcgis 10.1)

¹ <http://www.python.org>

Python + Arcgis

- Python can interact with Arcgis and be used to repeat many types of analyses.
- Why Python?
- It is an integral part of Arcgis
- Easy to read syntax
- Large user community
- Useful for scripts to control other programs

How does Python work with Arcgis

- At Arcgis 10.1
 - Fully integrated into Arcgis
 - Largely Geoprocessing functions
 - Automated mapping is not possible, yet

Logistics

- We will be using the IDLE programming environment
- Windows: START -> Programs -> Arcgis -> Python 2.7 -> IDLE
- We are using Arcgis 10.1 on lab computers and assume that you are using 10.1 if you are using your own laptop

Programming concepts

- Variables
- Control Structures (IF statements and FOR loops)
- Functions

Python is case sensitive and reads whitespace for defining programming blocks – use space bar, not tabs.

The Print Function and Strings

```
# this is a comment  
print "hello world"
```

""" Alternative
Commenting
Style """

The Print function and Strings

```
# this is a comment  
print "hello world"
```

```
# this is a variable that contains a string  
name = "Daniel"  
print name
```

Integers and Floats

```
# declare variables
```

```
int_sample = 10
```

```
float_sample = 10.0
```

```
# printing variables
```

```
# cast non-string variable as a string using str()
```

```
print "The value of this integer is: " + str(int_sample)
```

```
print "The value of this float is: " + str(float_sample)
```

if statement

```
x = 2
```

```
# Condition checks if statement is true
```

```
if x == 1:
```

```
    print 'x is 1!'
```

if / elif / else statement

```
x = 2
```

```
# Condition checks if statement is true
```

```
if x == 1:
```

```
    print 'x is 1!'
```

```
elif x == 2:
```

```
    print 'x is 2!'
```

```
else:
```

```
    print 'x is not known'
```

for loop

```
for i in range(3):
```

```
    # convention is to use 4 spaces to indent
```

```
    # python reads whitespace at the beginning of a line
```

```
    print i
```

Python, like most programming languages, uses arrays that are zero based.

while loop

```
# define j
```

```
j = 1
```

```
# 'while' less than some condition
```

```
while j < 3:
```

```
    print j
```

```
    # increment j
```

```
    j += 1
```

Three ways to access a folder

Accessing a folder

```
path = "C:\\folderName\\"
```

```
path = "C:/folderName/"
```

```
path = r"C:\folderName\"
```

Importing Modules

Use the *import* command:

```
# count the number of files in a directory
```

```
import os  
files = os.listdir(path)  
len(files)
```

A module is a list of Python programs that can be accessed. Commonly used modules are *os*, *sys*, *glob*.

glob

```
import glob # use the glob module
path =
    "C:\\users\\dsheehan\\2012_work\\JPAL\\"
# loop through all files
for i in glob.glob(path + "*"):
    print i
```

Try replacing '*' with '*.shp'

Importing the Arcgis module

At 10.0 and 10.1

```
import arcpy
```

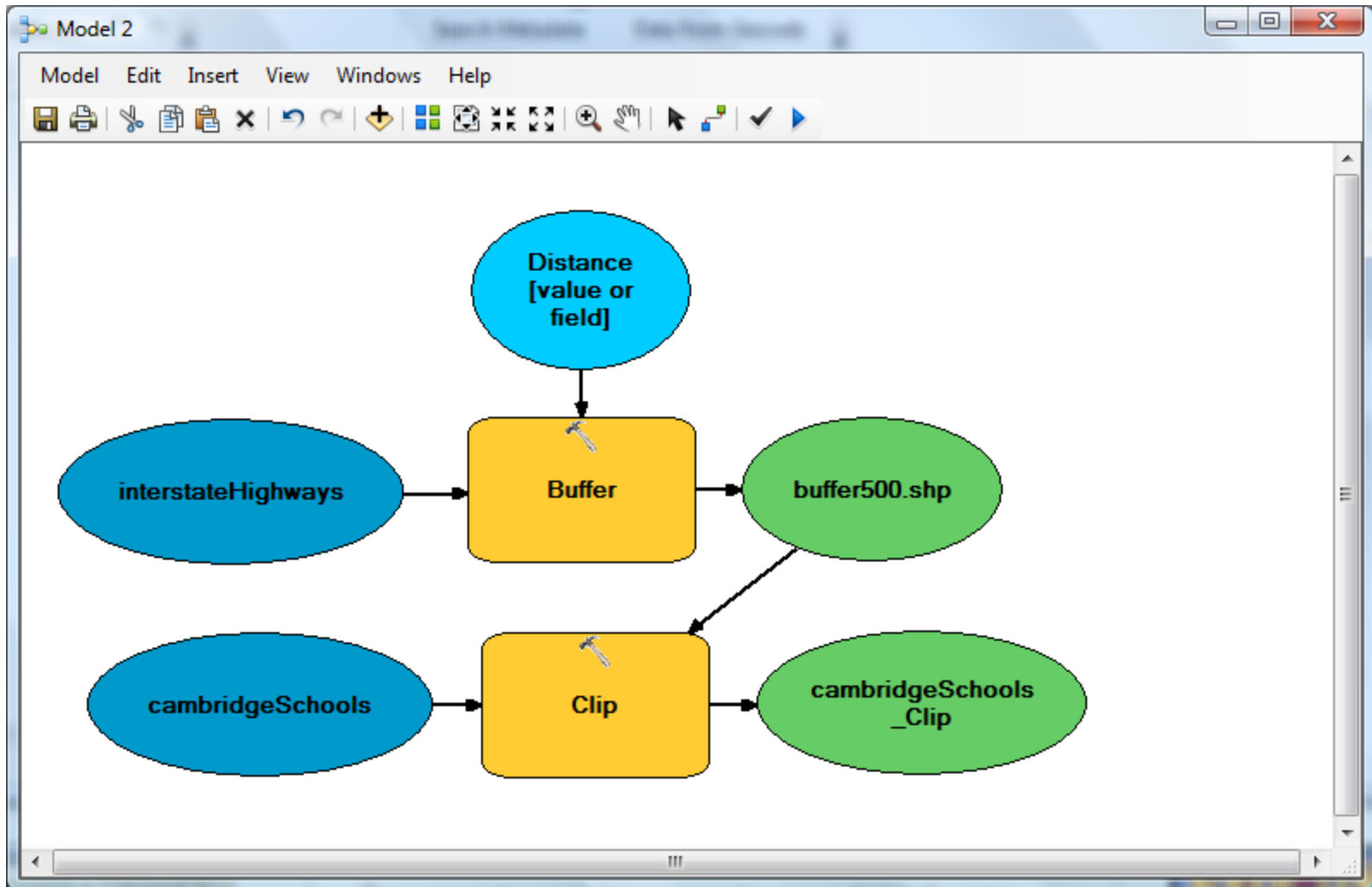
At 9.3:

```
import arcgisscripting
```

Exercise 1: Reading folder contents

- Download zip file from course site:
- Using the glob module, print out:
 - a list of all of the files
 - a list of shapefiles

Model Builder



Exercise 2: ModelBuilder

Using ModelBuilder:

- Buffer interstateHighways.shp (500 meters)
 - Units of data is meters
- Clip cambridgeSchools.shp with buffer
- Export model as 'Python'

Catching exceptions

Try:

<your code>

except:

```
print arcpy.GetMessages()
```

```
raise
```

Overwriting files

- `from arcipy import env`
- `env.overwriteOutput = True`

Exercise 3: Convert ModelBuilder Code into a loop

- Using the code from ModelBuilder
- Identify relative filepaths and restructure code
- Iterate through this loop 2 times, buffering 500 meters, 1000 meters
- Intersect cambridgeSchools.shp with buffer and make 2 new shapefiles

Writing to a text file

Create a file ('w' means create a new file, 'a' appends to an existing file, will create it if it doesn't already exist)

```
f = open("C:\\users\\dsheehan\\test.txt", 'w')
```

Write to a file

```
f.write("Contents of file" + "\n")
```

```
f.flush() # flushes buffer
```

```
f.close() # closes file
```

Exercise 4: File Manipulation

Create a folder called “temp_folder”:

- Make 5 text files called File1.txt, File2.txt, etc.
- Write a string in each file