Python Workshop
Process Flow

Joseph D. Hughes

U.S. Geological Survey
Florida Water Science Center, Tampa, Florida USA

USGS National Groundwater Workshop, August 2012

Outline

Overview

- Much of what is useful to do in Python is reading files, manipulating the data, and writing out results in another format
- Python and Numpy provide ways to read and write ASCII and binary files. We will focus on ASCII files

USGS
science for a changing world

Background Information

Process flow control resources:

http://docs.python.org/tutorial/controlflow.html

while, continue, and break

ProcessFlowExamples.py

Import data from an external file and iterate over data using **while** and **print** last entry.

```
1   import numpy as np
2   #--load flow data
3   q = np.genfromtxt( 'USInflow.dat', skip_header=1 )
4   #--determine sizes
5   ntimes, ncol = q.shape[0], q.shape[1]
6   print 'Number of times:   {0:10d}\nNumber of entries: {1:10d}'.format( ntimes, ncol )
7   #--while iteration with break and continue
8   ipos = 0
9   while True:
10      #--only print last line
11      if ipos+1 < ntimes:
12          #--increment ipos by one
13          ipos += 1
14          continue
15      #--print data
16      print '{0:25s}: {1}, {2}, {3}'.format( 'while iteration', ipos+1, q[ipos,0], q[ipos,1] )
17      #--terminate after printing last element
18      break
```

`range` iterator and `continue`

`ProcessFlowExamples.py`

Import data from an external file and iterate over data using **`range`** and **`print`** last entry.

```
1   import numpy as np
2   #--load flow data
3   q = np.genfromtxt( 'USInflow.dat', skip_header=1 )
4   #--determine sizes
5   ntimes, ncol = q.shape[0], q.shape[1]
```



```
19  #--range iterator - range creates a list
20  for ipos in range(0,ntimes):
21      #--only print last line
22      if ipos+1 < ntimes:
23          continue
24      #--print data
25      print '{0:25s}: {1}, {2}, {3}'.format( 'range iteration', ipos+1, q[ipos,0], q[ipos,1] )
```

`xrange` iterator and `continue`

`ProcessFlowExamples.py`

Import data from an external file and iterate over data using **`xrange`** and **`print`** last entry.

```
1   import numpy as np
2   #--load flow data
3   q = np.genfromtxt( 'USInflow.dat', skip_header=1 )
4   #--determine sizes
5   ntimes, ncol = q.shape[0], q.shape[1]
```

⋮

```
26  #--xrange iterator - xrange is a generator
27  for ipos in xrange(0,ntimes):
28      #--only print last line
29      if ipos+1 < ntimes:
30          continue
31      #--print data
32      print '{0:25s}: {1}, {2}, {3}'.format( 'xrange iteration', ipos+1, q[ipos,0], q[ipos,1] )
```

`in` iterator and `continue`

`ProcessFlowExamples.py`

Import data from an external file and iterate over data using **`in`** iterator and **`print`** last entry.

```
1   import numpy as np
2   #--load flow data
3   q = np.genfromtxt( 'USInflow.dat', skip_header=1 )
4   #--determine sizes
5   ntimes, ncol = q.shape[0], q.shape[1]
```

⋮

```
33  #--element iterator
34  ipos = 0
35  for t in q:
36      #--increment ipos by one
37      ipos += 1
38      #--only print last line
39      if ipos < ntimes:
40          continue
41      #--print data
42      print '{0:25s}: {1}, {2}, {3}'.format( 'element iteration', ipos, t[0], t[1] )
```

enumerate iterator and continue

ProcessFlowExamples.py

Import data from an external file and iterate over data using enumerate iterator and **print** last entry.

```
1   import numpy as np
2   #--load flow data
3   q = np.genfromtxt( 'USInflow.dat', skip_header=1 )
4   #--determine sizes
5   ntimes, ncol = q.shape[0], q.shape[1]
```

⋮

```
43  #--enumeration iterator
44  for ipos,t in enumerate(q):
45      #--only print last line
46      if ipos+1 < ntimes:
47          continue
48      print '{0:25s}: {1}, {2}, {3}'.format( 'enumeration iteration', ipos+1, t[0], t[1] )
```

ProcessFlowExamples.py output