

Python Workshop

Built-In Data Structures

C.D. Langevin

U.S. Geological Survey
Reston, Virginia, USA

USGS National Groundwater Workshop, August 2012

Outline

- 1 Concepts
- 2 Data Structures
 - Numbers
 - Strings
 - Lists
 - Dictionaries
 - Tuples
 - Others
 - Shared References
 - Zip
- 3 Numpy Arrays
- 4 Summary
- 5 Embedding Code

Concepts

- Data Structure – A way to store and organize data in a computer
- Mutability – An *immutable* object is one whose property or state cannot be changed. Whereas, *mutable* objects can change state.

Numbers

- integer (int)
- float/double precision (float)
- long integer (long)
- complex (complex)

```
In [2]: type(2)  
Out[2]: <type 'int'>
```

```
In [3]: type(2.0)  
Out[3]: <type 'float'>
```

```
In [4]: type(2**100)  
Out[4]: <type 'long'>
```

```
In [5]: type(2j)  
Out[5]: <type 'complex'>
```

Strings

- An immutable sequence of characters.

```
In [69]: s = 'modflow'
```

```
In [70]: s.upper()
```

```
Out[70]: 'MODFLOW'
```

```
In [71]: s.capitalize()
```

```
Out[71]: 'Modflow'
```

```
In [72]: s[0]
```

```
Out[72]: 'm'
```

```
In [73]: s[-1]
```

```
Out[73]: 'w'
```

```
In [74]: s[0:4]
```

```
Out[74]: 'modf'
```

```
In [75]: len(s)
```

```
Out[75]: 7
```

```
In [76]: 'Modflow' + '-88'
```

```
Out[76]: 'Modflow-88'
```

Lists

- A mutable collection of objects.
- List members are accessed using a zero-based indexing scheme.

```
In [79]: l = []
```

```
In [80]: l.append('first')
```

```
In [81]: l.append(2)
```

```
In [82]: l.append(3.0)
```

```
In [83]: l  
Out[83]: ['first', 2, 3.0]
```

```
In [84]: l[0]  
Out[84]: 'first'
```

```
In [85]: l[1]  
Out[85]: 2
```

```
In [86]: l[2]  
Out[86]: 3.0
```

```
In [87]: len(l)  
Out[87]: 3
```

List Methods

- append
- count
- extend
- index
- insert
- pop
- remove
- reverse
- sort

```
In [115]: l
Out[115]: ['mf.dis', 'mf.bas', 'mf.pcg', 'mf.lpf']
```

```
In [116]: l =
['mf.dis', 'mf.bas', 'mf.lpf', 'mf.pcg']
```

```
In [117]: l.index('mf.bas')
Out[117]: 1
```

```
In [118]: l.remove('mf.pcg')
```

```
In [119]: l.append('mf.sip')
```

```
In [120]: l
Out[120]: ['mf.dis', 'mf.bas', 'mf.lpf', 'mf.sip']
```

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Numbers
Strings
Lists
Dictionaries
Tuples
Others
Shared References
Zip

Dictionaries

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Numbers
Strings
Lists
Dictionaries
Tuples
Others
Shared References
Zip

Tuples

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Numbers
Strings
Lists
Dictionaries
Tuples
Others
Shared References
Zip

Others

Sets
Boolean
None

Shared References

- In-Place Changes
- Shared References and Equality

See pages 116-121 in Learning Python, Third Edition

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Numbers
Strings
Lists
Dictionaries
Tuples
Others
Shared References
Zip

Zip

- What is Zip?

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

What is Numpy

Creating an Array

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Working with Arrays

Concepts
Data Structures
Numpy Arrays
Summary
Embedding Code

Summary

Code In Frame

Here is some python code. Note that the [fragile] keyword is required on the begin frame line. See the following site for additional details: <http://robfelty.com/2008/09/22/beamer-fragile-frames>

```
class GridNodeIterator(object):
    def __init__(self, grid):
        self.index = 0
        return

    def __iter__(self):
        return self

    def next(self):
        if self.index == self.nodes - 1:
            raise StopIteration()
        nodeobj = self.get_nodeobj(self.index)
        self.index += 1
        return nodeobj
```