

Python Workshop

Senthil Kumaran

CollabNet

September 30, 2010

Agenda

- History
- Basics
- Control Flow
- Function
- Modules

History

What is Python?

- Python is a general purpose, object-oriented, high level, interpreted language
- Created in early 90's by Guido Van Rossum
- Simple, portable and powerful
- Free Software
- Influenced by ABC, ALGOL 68, C, Haskell, Icon, Lisp, Modula-3, Perl, Java

Why learn Python?

- Easier to learn than compiled languages like C/C++
- Fast development of POC code
- Cross Platform
- Batteries Included
- Great documentation
- Strong community support

Application Domains

- Web and Internet development
- Database Access
- Desktop GUIs
- Scientific and Numeric
- Education
- Network Programming
- Software Development
- Games and 3D Graphics

Versions of Python

- Python 2.5
- Python 2.6
- Python 3.0
- Why 2.5/2.6 and !3.0

Installing

- Download from <http://www.python.org/>
- Most GNU/Linux distributions have it already

Editing Python

- IDLE
- Emacs
- Vi/Vim
- Whatever editor you want

Python Interpreter

- Interactive session
- Use of interactive session

Exit python interpreter

- `quit()`
- Ctrl + D on NIX
- Ctrl + Z on Windows

Basics

Hello World

```
$ emacs hello_world.py
```

```
$ cat hello_world.py
```

```
#!/usr/bin/python
```

```
print ‘‘Hello World’’
```

```
$ chmod +x hello_world.py
```

```
$ ./hello_world.py
```

```
Hello World
```

```
$ python
```

```
Python 2.5.2 (r252:60911, Jan 4 2009, 17:40:26)
```

```
[GCC 4.3.2] on linux2
```

```
Type ‘‘help’’, ‘‘copyright’’, ‘‘credits’’ or ‘‘license’’ for more  
information.
```

```
>>> print ‘‘Hello World’’
```

```
Hello World
```

Indentation

- Is very important in Python
- No begin/end delimiters
- Comments start with #

Data Types

Integer Numbers

- Decimal - 1, 3, 87
- Octal - 01, 022
- Hexa - 0x1, 0x22
- Long - 1L, 456666343L

Floating Point

- 0.0, 400.34, 5e3, 45e5

Complex Numbers

- $-1+5j$, $5-6j$

Strings

```
print "Hello World" #Correct
```

```
print 'Hello World' #Correct
```

```
print "Hello World' #Wrong
```

```
print """  
This is line one  
This is line two  
This is line three  
"""
```

Tuple

- Immutable ordered sequence of items
- Assigning - `a = (1234, 1456, 1212)`
- Using tuples - can be used as a constant array
- Data can be accessed similar to an array - `a=(132,3232,323)`
- `a[1]` or `a[3]`

Lists

- List is a mutable ordered sequence of items (similar to tuple)
- Assigning - `a = (1234, 1456, 1212)`
- Using tuples - can be used as a constant array
- Data can be accessed similar to an array - `a=(132,3232,323)`
- `a[1]` or `a[3]`

Dictionaries

- Dictionaries are containers, which store items in a key/value pair
- Assigning - `d = 'x':24, 'y':33`
- Using Dict - They are used at a lot of places
- Data can be accessed by using the key - `d['x']`

Variables

- There is no prior declaration needed
- Variables are the references to the allocated memory
- Variables can refer to any data type (like Tuple, List, Dictionary, Int, String, Complex)
- References are share
- List, Dict etc are always shared

Index and slices

- String, List, Tuple, etc can be sliced to get a part of them
- Index - similar to array index, it refers to 1 position of data
- Slices - gives the data in the range
- Example - `a="Mepco Schlenk Engineering College"` `a[:3]` `a[4:11]` `a[4:]`
`a[-7:]` `a[:-8]` `a[:11:2]`

Control Flow

print

- Print is a simple statement for giving output similar to C's printf function
- Can be used to output to Console or a file
- Use - print "Hello World"

input

- Use `raw_input()` to take a string input from the user
- Used as

```
var = raw_input('Enter a String:')
```

- `Input()` is used to take a input without specifying the type

- If is a conditional statement, for simple “If then else” clause in English
- Header lines are always concluded with a “:” followed by intended block of statements
- Optionally it can be followed by an “else if” clause known as “elif” in python

```
if condition:
    Statement 1
    Statement 2
elif condition:
    Statements
else:
    statements
```

while

- While statement is used for repeatedly executing a block of code till the condition is true, also has an optional else clause
- Use wildly for infinite loop

```
while condition:  
    statements  
else:  
    statements
```


for

- It is a sequence iterator
- It works on Strings, lists, tuples, etc
- for target in iterable: statements

range

- They are used to generate and return integer sequence
- `range(5)` - `[0,1,2,3,4]`
- `range(1,5)` - `[1,2,3,4]`
- `range(0,8,2)` - `[0,2,4,6]`

Skip code

break

- Used to terminate a loop
- If nested it terminates the inner most loop
- Practically used for conditional loop termination with an if statement

continue

- Terminates the current iteration and executes next
- Practically used for conditional statements termination with an if statement

Some Helpful Functions

- `dir()`
- `help()`

Functions

What are functions?

- A Function is a group of statements that execute on request
- In Python Functions are Objects
- Defining a function

```
def name(parameters):  
    statement(s)  
    return var
```

Parameters

- Types of parameters
- Mandatory Parameters
- Optional parameters
- Default values
- Be careful when default value is a mutable object

```
def a(x,y=[]):  
    y.append(x)  
    print y  
print a(12)  
print a(34)
```

Modules

What are modules?

- How to load modules
- Effect on namespace
- Important modules
- `os.sys`

Thank You

Thank You!

<http://www.stylesen.org/>
stylesen@gmail.com