

Python Workshop

Chaitanya Talnikar

Saket Choudhary

January 18, 2012

Named after this :

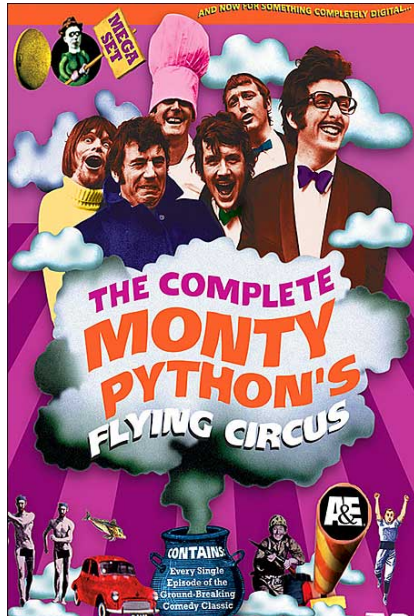


- Slide 1 was a joke !

- Slide 1 was a joke !
- Python : Conceived in late 1980s by Guido van Rossum as a successor to ABC programming language !

- Slide 1 was a joke !
- Python : Conceived in late 1980s by Guido van Rossum as a successor to ABC programming language !
- Philosophy: Language for readable code ! Remarkable power coupled with clear beautiful syntax !

- Slide 1 was a joke !
- Python : Conceived in late 1980s by Guido van Rossum as a successor to ABC programming language !
- Philosophy: Language for readable code ! Remarkable power coupled with clear beautiful syntax !
- Ideology behind Name : A Television Series *Monty Python's Flying Circus* !



- Readable Syntax, Clarity

```
print "Hi There !"
```


Python: The Power Pack

- Readable Syntax, Clarity
print "Hi There !"
- Intuitive object Orientedness

Python: The Power Pack

- Readable Syntax, Clarity
print "Hi There !"
- Intuitive object Orientedness
- Natural Expression of Procedural Code

Python: The Power Pack

- Readable Syntax, Clarity

```
print "Hi There !"
```

- Intuitive object Orientedness
- Natural Expression of Procedural Code
- Full Modularity, support for heirarchical packages

Python: The Power Pack

- Readable Syntax, Clarity

```
print "Hi There !"
```

- Intuitive object Orientedness
- Natural Expression of Procedural Code
- Full Modularity, support for heirarchical packages
- Exception-based Error Handling !

Python: The Power Pack

- Readable Syntax, Clarity

```
print "Hi There !"
```

- Intuitive object Orientedness
- Natural Expression of Procedural Code
- Full Modularity, support for heirarchical packages
- Exception-based Error Handling !
- Is Open Source !

Python can run everywhere, Literally !

- Windows, Mac, Linux
- Most of Linux versions have Python pre-installed for other dependences
- Python for Windows : <http://python.org/getit/>
- We will stick to Python2.7 for the session

- On an average Python code is smaller than JAVA/C++ codes by 3.5 times owing to Python's built-in datatypes and dynamic typing

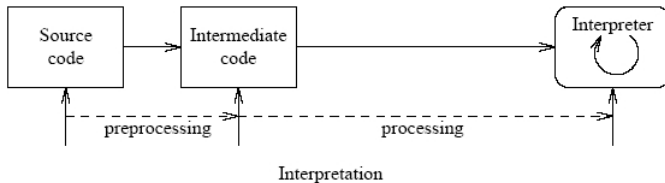
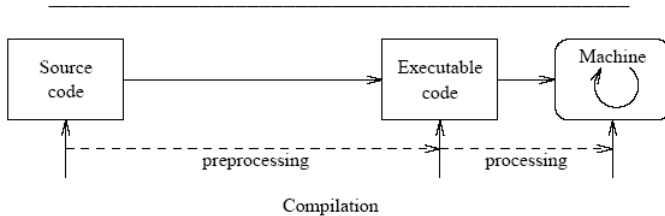
Python v/s Other Languages

- On an average Python code is smaller than JAVA/C++ codes by 3.5 times owing to Python's built-in datatypes and dynamic typing
- No Declarations of Arguments or variables !

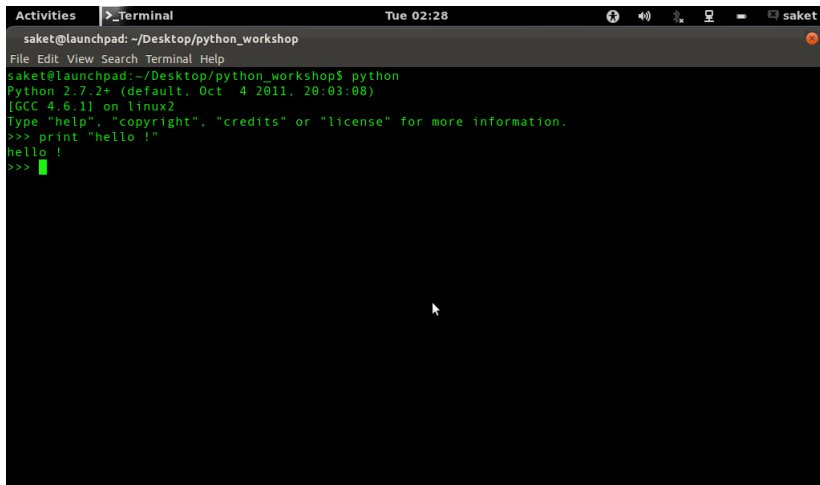
- On an average Python code is smaller than JAVA/C++ codes by 3.5 times owing to Python's built in datatypes and dynamic typing
- No Declarations of Arguments or variables !
- Dynamically declare and use variables !

- On an average Python code is smaller than JAVA/C++ codes by 3.5 times owing to Python's built-in datatypes and dynamic typing
- No Declarations of Arguments or variables !
- Dynamically declare and use variables !
- Python is interpreted while C/C++ are compiled.
 - Compiler : spends a lot of time analyzing and processing the program, faster program execution
 - Interpreter : relatively little time is spent analyzing and processing the program, slower program execution

Compiler v/s Interpreter



Python Interpreter Interactive



The screenshot shows a terminal window titled "saket" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content is as follows:

```
saket@launchpad: ~/Desktop/python_workshop
saket@launchpad:~/Desktop/python_workshop$ python
Python 2.7.2+ (default, Oct 4 2011, 20:03:08)
[GCC 4.6.1] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> print "hello !"
hello !
>>> █
```

Running Python Programs

- Python is a scripting language. No *edit-compile-link-run procedures*

Running Python Programs

- Python is a scripting language. No *edit-compile-link-run procedures*
- Python programmes are stored in files, called as Python scripts
saket@launchpad: `python filename.py`

Running Python Programs

- Python is a scripting language. No *edit-compile-link-run procedures*
- Python programmes are stored in files, called as Python scripts
saket@launchpad: `python filename.py`
- One of the ideology behind Python was to have a Language without braces . Cool (?)

- Variable declaration: The usual suspects:

- Strings
- int
- float
- bool
- complex
- files

- No declaration required

`x = 3`

`x ** 50`

- Data Structures:

- Tuples - Immutable(Fixed length)
("this", "cannot", "be" , "appended by anything")
- Lists
["this", "resembles", "traditional", "arrays"]

- Dictionaries

```
{"this": "corresponds to this", "and this": "to  
this"}
```

- set

```
set(["this", "has", "unique", "set of", "elements"])  
=> union, intersection, difference
```

- Maths module for your MA Courses !

```
import math  
math.sqrt(10)  
math.log(10,3)  
math.radians(x)
```

- Maths module for your MA Courses !

```
import math  
math.sqrt(10)  
math.log(10,3)  
math.radians(x)
```

- Arrays of the scientific world `from numpy import *`

```
a = array([1,2,3]) and not a = array(1,2,3)  
a  
array([1,2,3])  
zeros(3,4)  
array([[0., 0., 0., 0.], [0., 0., 0., 0.], [0.,  
0., 0., 0.]])
```

Blocks in Python

- Python doesn't use curly braces or any other symbol for programming blocks like for loop, if statement.

Blocks in Python

- Python doesn't use curly braces or any other symbol for programming blocks like for loop, if statement.
- Instead indentation is used in the form of spaces of tabs.

```
if a == 2:  
    print 'Hello'
```

Blocks in Python

- Python doesn't use curly braces or any other symbol for programming blocks like for loop, if statement.
- Instead indentation is used in the form of spaces of tabs.

```
if a == 2:  
    print 'Hello'
```

- Recommended standard: 4 spaces
PS: For more python style rules visit
<http://www.python.org/dev/peps/pep-0008/>

Loops and Conditionals

- Simple for loop for a variable i,
`for i in range(0,10):`
 `print i`

Loops and Conditionals

- Simple for loop for a variable i,

```
for i in range(0,10):  
    print i
```
- The range part can be replaced by any list.

Loops and Conditionals

- Simple for loop for a variable i,

```
for i in range(0,10):  
    print i
```
- The range part can be replaced by any list.
- if statement used for conditionals.

```
if a not in b:  
    print 'Hello'  
else:  
    print 'Bye'
```
- while loops can also be used the same way

Functions in python

- Simple to define. Multiple return values. Default parameters.

```
def sum(a, b=5):  
    return a+b  
print sum(2, 3)  
s = sum(4)
```

- Lambda expressions: can be used to quickly define functions.

```
g = lambda x: x**2  
g(4)
```

Lists and Tuples

- Tuples are just like lists, except their values cannot be changed.

```
a = (2, 3)  
print a[0]
```

- Items can be added to a list using append and deleted using remove

```
b = [2, 3]  
b.append(4)  
b.remove(2)
```

Lists and Tuples

- Tuples are just like lists, except their values cannot be changed.

```
a = (2, 3)  
print a[0]
```

- Items can be added to a list using append and deleted using remove

```
b = [2, 3]  
b.append(4)  
b.remove(2)
```

- The length of a list can be found out using len. Lists can store any type of object, you can even mix them len(b)
l = [2, 'Hi']

- Dicts hold a value given a key, they can be used to store records.

```
d = {'H2': 23, 'H3':17}  
print d['H2']  
d['H5'] = 30
```

- Dicts hold a value given a key, they can be used to store records.

```
d = {'H2': 23, 'H3':17}  
print d['H2']  
d['H5'] = 30
```

- To get a list of all keys or values

```
print d.keys()  
v = d.values()
```

- Dicts hold a value given a key, they can be used to store records.

```
d = {'H2': 23, 'H3':17}  
print d['H2']  
d['H5'] = 30
```

- To get a list of all keys or values

```
print d.keys()  
v = d.values()
```

- The lists can also be sorted.

```
v.sort()
```

- To open a file as read/write.
`f = open('test.txt', 'r+')`

- To open a file as read/write.
`f = open('test.txt', 'r+')`
- To read the file entirely/line by line

```
print f.read()  
print f.readline()
```

- To open a file as read/write.

```
f = open('test.txt', 'r+')
```

- To read the file entirely/line by line

```
print f.read()
```

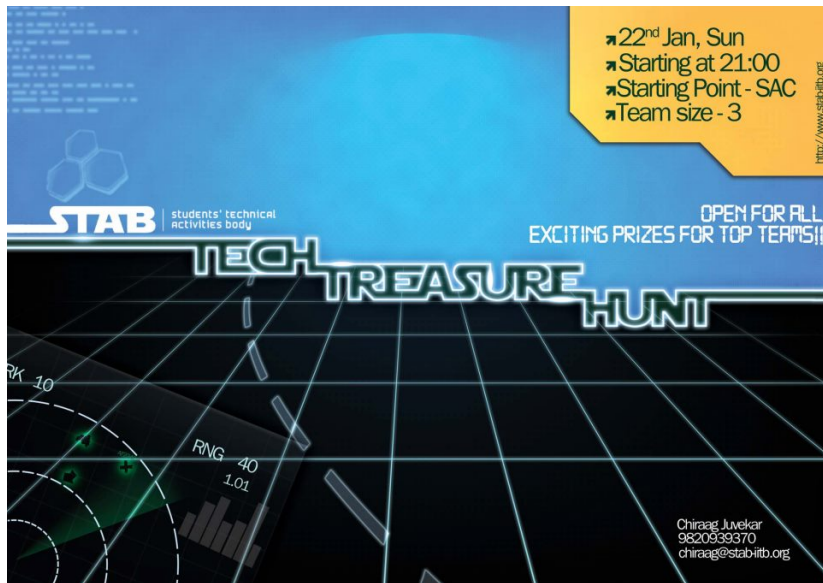
```
print f.readline()
```

- Writing to a file, note the text is written at the current file location of f

```
f.write("text")
```

```
f.close()
```

slides.close(): Are you ready to *hunt* ?

A poster for a 'Tech Treasure Hunt' event. The background is a dark blue grid with a glowing blue sphere in the upper center. In the bottom left, there's a radar-like graphic with green stars and labels 'RK 10', 'RNG 40', and '1.01'. The STAB logo is on the left, and event details are in a yellow box on the right. The title 'TECH TREASURE HUNT' is in large, glowing letters across the middle.

STAB | students' technical activities body

TECH TREASURE HUNT

OPEN FOR ALL
EXCITING PRIZES FOR TOP TEAMS!!

- 22nd Jan, Sun
- Starting at 21:00
- Starting Point - SAC
- Team size - 3

<http://www.stab-iitb.org>

Chiraag Juvekar
9820939370
chiraag@stab-iitb.org