PYTHON

PYTHON PROGRAMMING

By
ADITYA PRABHAKARA

Introducing Myself

Aditya S P (sp.aditya@gmail.com)
Freelance trainer and technologist

Boring Stuff about me:

- 14+ years of experience in development and training
- Started with Java, moved to Android and now working on Big Data Technologies

Interesting Things about me:

- Actually Nothing !

Getting to know you

Show of hands please!

- Any freshers in this group?
- What is the general development experience of this group
  - 0-2 years, 0-5 years, 5 and above
- What programming area are you currently working on?
  - Java, Web Stack, Analytics, Big data, any other
- Why are you learning python programming?
  - Sys admin, Web development, Data Analytics, IoT, any other

Agenda

- Python programming
- Advanced Python
- Object Oriented Programming in Python

PYTHON

Course Objectives

- At ease with python programming
- Pythonic way of coding
- Learn OOP in python

Chapter: Introduction

Python

High Level

Interpreted

Dynamic Programming language

Multi-paradigm language

- OO
- Functional
- Procedural
- Imperative

The idea of Python started in 1980 and the implementation began by 1990

- Author: Guido Von Rossum

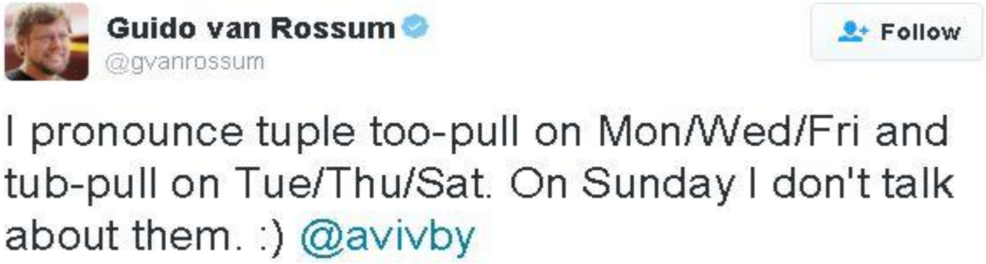
PYTHON

Guido van Rossum

In Guido van Rossum’s words

Over six years ago, in December 1989, I was looking for a "hobby" programming project that would keep me occupied during the week around Christmas. My office ... would be closed, but I had a home computer, and not much else on my hands. I decided to write an interpreter for the new scripting language I had been thinking about lately: a descendant of ABC that would appeal to Unix/C hackers. I chose Python as a working title for the project, being in a slightly irreverent mood (and a big fan of Monty Python's Flying Circus).

Python ‘s **Benevolent Dictator For Life** (BDFL)



Guido van Rossum
@gvanrossum

I pronounce tuple too-pull on Mon/Wed/Fri and tub-pull on Tue/Thu/Sat. On Sunday I don't talk about them. :) @avivby

A funny Guy : Check his answer on asking how to pronounce “tuple” – a data structure in Python

PYTHON

Zen of Python

A set of 20 aphorisms. My favs below

- Beautiful is better than ugly.
- Simple is better than complex.
- Complex is better than complicated.
- Readability counts.
- Errors should never pass silently.
- In the face of ambiguity, refuse the temptation to guess.
- If the implementation is hard to explain, it's a bad idea.
- If the implementation is easy to explain, it may be a good idea.

--Tim Peters

Define aphorism:

- a pithy observation which contains a general truth.
- a concise statement of a scientific principle, typically by a classical author.

Why should I love python?

Python is very concise

```
public class HelloWorld
{
        public static void main
 (String[] args)        {

        System.out.println(“Hello
world!”);
        }
}
```

```
print “Hello world!”
```

Why should I love python?

Python is as readable as a pseudo code.

Readability is very important because very often code is read more than its written

```
fileHandle = open(“somefile.txt”, “r”)

for line in fileHandle:
    print line
```

Why should I love python?

Eliminates a lot of syntax overheads.

- No dreaded semi-colon at the end “;”
- No flower brackets “}” for blocks

```
var1 = 100

if var1>100:
    print “Not a century”
    print var1
else:
    print “It’s a century”
    print var1

print “Bye!”
```

Why should I love python?

Follows duck typing

- Lot less coding.
- Forces a clean understanding of code before using it.
- Can be dangerous too, depending on what kind of a programmer you are!

```
def f(someobj):
    someobj.quack()
```

```
If a equals 10:
   f(b)

#suppose b cannot quack and a is
#very rarely 10. you will not see
#an error as long as a is never 10
```

Why should I love python?

Python comes with batteries included.

Python Standard Library

Simply put, Python has a large standard library. Some examples

- HTTP Protocols
- Database access
- IPC
- File system access

Python Package Index

- Python Package Index is a repository for python libraries
- Currently has over 89000 packages
- These can be installed through pip

Version of Python for this course

- We will be using version 2.7.12
- The latest version is 3.7
- Python 3 and Python 2 are radically different at a many places.
  - Implies all the libraries have to be ported to Python 3
  - Due the compatibility issues Python 2 versions continue to find favour
  - In a couple of years time, we would see everything in Python 3
- Incremental learning from 2.x to 3.x is not very difficult

Chapter: Setup

PYTHON

Python setup or installation

Step 1
- Download from https://www.python.org/downloads/

Step 2
- Run the installer and click through

Step 3
- Run IDLE and the python shell should come up

Check the installation guide provided (You will not need this)