



Future Skills Training Program

Data Science and Analytics

Topic:- Basics of Python



Dr. Sunil Joshi

Agenda

- ☐ What is Python...?
- ☐ Differences between program and scripting language
- ☐ History of Python
- ☐ Scope of Python
- ☐ Why do people use Python?
- ☐ Installing Python IDE
- ☐ Who uses python today
- ☐ What can I do with python
- ☐ A Sample Code
- ☐ Python code execution
- ☐ Running Python

What is Python...?

- ❑ Python is a general purpose programming language that is often applied in scripting roles.
- ❑ So, Python is programming language as well as scripting language.
- ❑ Python is also called as Interpreted language

Differences between program and scripting language

Program

- a **program is executed** (*i.e. the source is first compiled, and the result of that compilation is expected*)
- A "program" in general, is a **sequence of instructions** written so that a computer can perform certain task.

Scripting

- a **script is interpreted**
- A "script" is code written in a scripting language. A scripting language is nothing but a **type of programming language in which we can write code to control another** software application.

History

- ❑ Invented in the Netherlands, early 90s by Guido van Rossum
- ❑ Python was conceived in the late 1980s and its implementation was started in December 1989
- ❑ Guido Van Rossum is fan of 'Monty Python's Flying Circus', this is a famous TV show in Netherlands
- ❑ Named after Monty Python
- ❑ Open sourced from the beginning

Why was python created?

"My original motivation for creating Python was the perceived need for a higher level language in the Amoeba [Operating Systems] project.

I realized that the development of system administration utilities in C was taking too long. Moreover, doing these things in the Bourne shell wouldn't work for a variety of reasons. ...

So, there was a need for a language that would bridge the gap between C and the shell"

- Guido Van Rossum

Scope of Python

- Science
 - Bioinformatics
- System Administration
 - Unix
 - Web logic
 - Web sphere
- Web Application Development
 - CGI
 - Jython – Servlets
- Testing scripts

Why do people use Python...?

The following primary factors cited by Python users seem to be these:

- **Python is object-oriented**

Structure supports such concepts as polymorphism, operation overloading, and multiple inheritance.

- **Indentation**

Indentation is one of the greatest feature in Python.

- **It's free (open source)**

Downloading and installing Python is free and easy
Source code is easily accessible



□ It's powerful

- Dynamic typing
- Built-in types and tools
- Library utilities
- Third party utilities (e.g. Numeric, NumPy, SciPy)
- Automatic memory management

□ It's portable

- Python runs virtually every major platform used today
- As long as you have a compatible Python interpreter installed Python programs will run in exactly the same manner irrespective of platform.

□ It's mixable

- Python can be linked to components written in other languages easily
- Linking to fast, compiled code is useful to computationally intensive problems
- - Python/C integration is quite common

□ It's easy to use

- No intermediate compile and link steps as in C/ C++
- Python programs are compiled automatically to an intermediate form called *bytecode*, which the interpreter then reads
- This gives Python the development speed of an interpreter without the performance loss inherent in purely interpreted languages

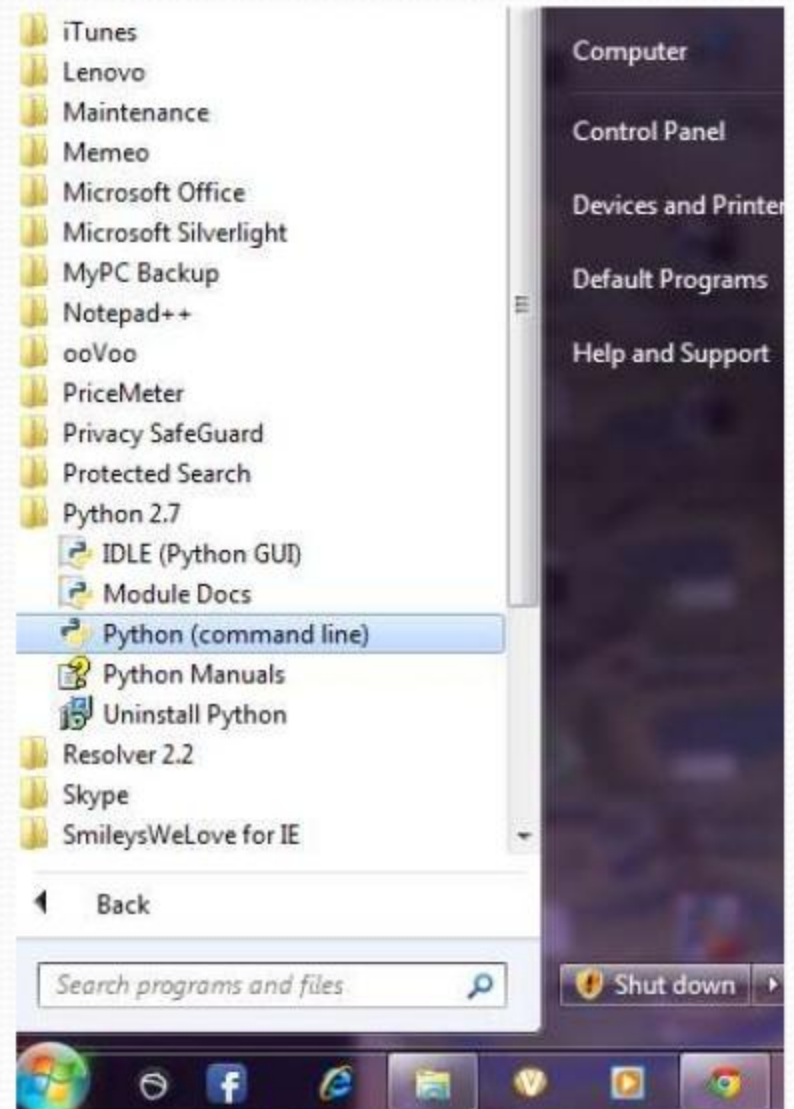
□ It's easy to learn

- Structure and syntax are pretty intuitive and easy to grasp

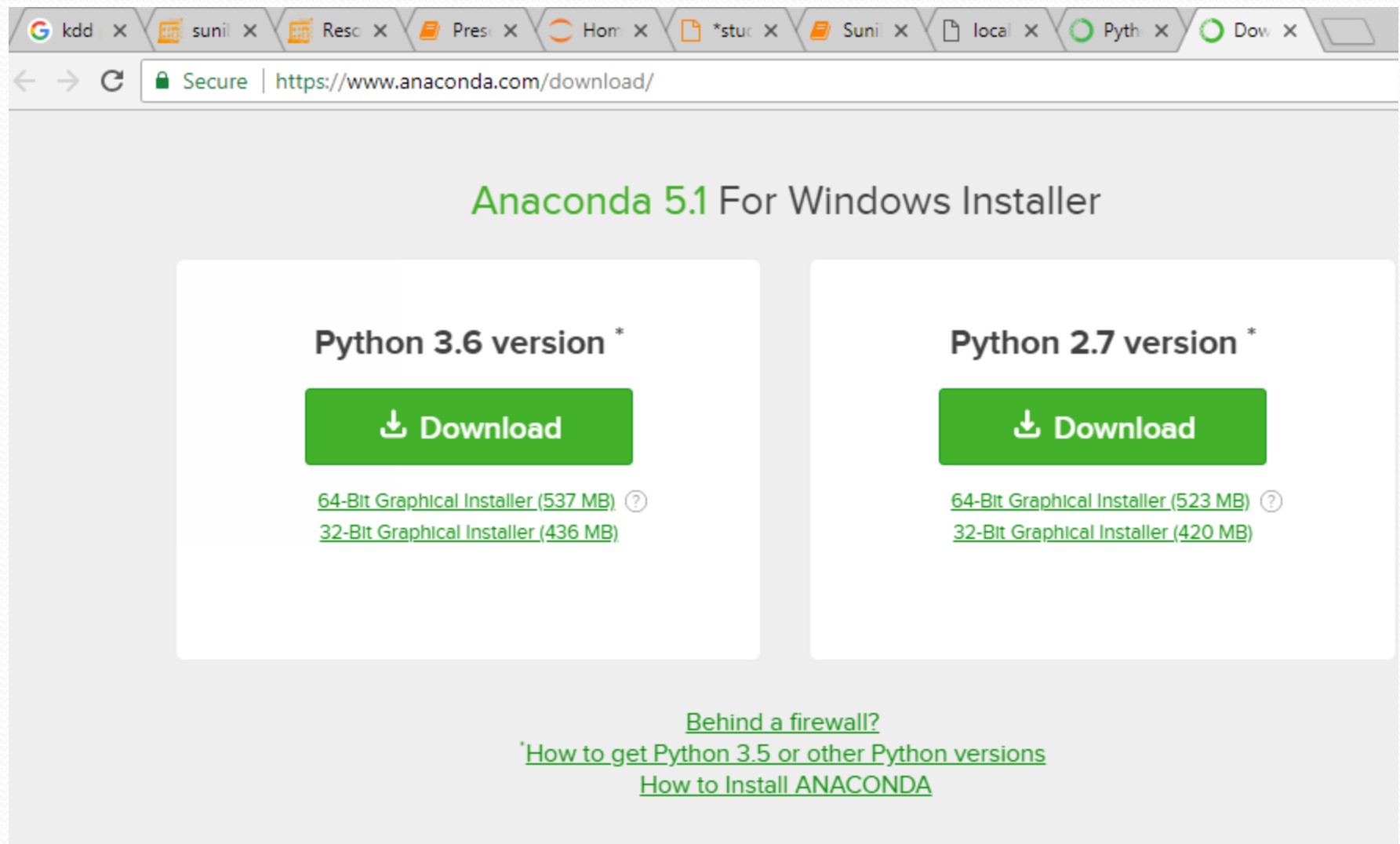
Installing Python

- Python is pre-installed on most Unix systems, including Linux and MAC OS X
- But for in Windows Operating Systems , user can download from the <https://www.python.org/downloads/>
 - from the above link download latest version of python IDE and install, recent version is 3.4.1 but most of them uses version 2.7.7 only

- After installing the Python Ver#2.7.7, go to start menu then click on python 2.7 in that one you can select python (command line) it is prompt with >>>



How to Install Anaconda



Browser tabs: kdd, sunil, Resc, Pres, Hom, *stuc, Suni, local, Pyth, Dow

Secure | <https://www.anaconda.com/download/>

Anaconda 5.1 For Windows Installer

Python 3.6 version *

[Download](#)


[64-Bit Graphical Installer \(537 MB\)](#) ?
[32-Bit Graphical Installer \(436 MB\)](#)


Python 2.7 version *


[Download](#)


[64-Bit Graphical Installer \(523 MB\)](#) ?
[32-Bit Graphical Installer \(420 MB\)](#)


[Behind a firewall?](#)
[How to get Python 3.5 or other Python versions](#)
[How to Install ANACONDA](#)

 Home

 Environments

 Projects (beta)

 Learning

 Community

Documentation

Developer Blog

Applications on

base (root)

Channels



jupyterlab

0.31.5

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Launch



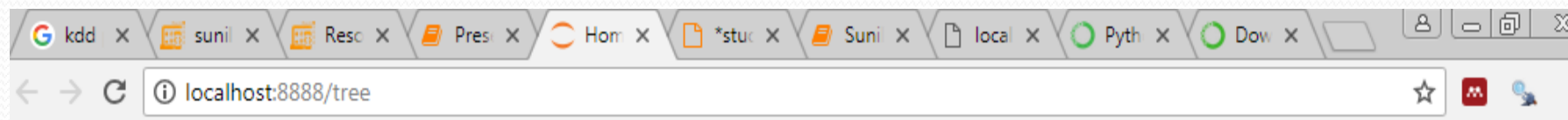
notebook

5.4.0

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Launch





Logout

Files Running Clusters

Select items to perform actions on them.

0 ▾

/

<input type="checkbox"/>	Contacts	
<input type="checkbox"/>	Desktop	
<input type="checkbox"/>	Documents	
<input type="checkbox"/>	Downloads	
<input type="checkbox"/>	Favorites	6 months ago
<input type="checkbox"/>	Links	5 months ago
<input type="checkbox"/>	Music	3 months ago
<input type="checkbox"/>	Pictures	2 months ago
<input type="checkbox"/>	Saved Games	a month ago
<input type="checkbox"/>	Searches	4 months ago
<input type="checkbox"/>

Upload

New ▾

↺

Notebook:

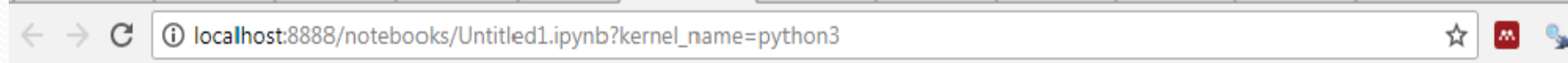
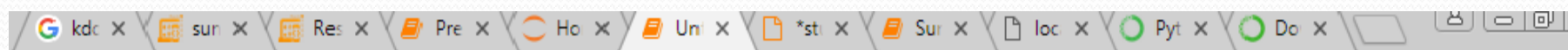
Python 3

Create a new notebook with Python

Text File

Folder

Terminal



Untitled1

Last Checkpoint: a few seconds ago (unsaved changes)



Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted



Python 3



In []:

The screenshot shows a web browser window with multiple tabs. The active tab is titled "Untitled1" and the address bar shows the URL "localhost:8888/notebooks/Untitled1.ipynb?kernel_name=python3". The Jupyter Notebook interface is displayed, featuring a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", "Widgets", and "Help". Below the menu bar is a toolbar with various icons. The "Run" button, represented by a play icon, is circled in red. The main area of the notebook contains a code cell with the following text:

```
In [1]: # To print Message use print function
        # To Execute Press Shift and Enter
        print("Hello World")
```

Below the code cell, the output "Hello World" is displayed. At the bottom of the notebook, there is an empty code cell with the prompt "In []:".

Who uses python today...

- Python is being applied in real revenue-generating products by real companies. For instance:
- Google makes extensive use of Python in its web search system, and employs Python's creator.
- Intel, Cisco, Hewlett-Packard, Seagate, Qualcomm, and IBM use Python for hardware testing.
- ESRI uses Python as an end-user customization tool for its popular GIS mapping products.
- The YouTube video sharing service is largely written in Python

What can I do with Python...?

- ☐ System programming
- ☐ Graphical User Interface Programming
- ☐ Internet Scripting
- ☐ Component Integration
- ☐ Database Programming
- ☐ Gaming, Images, XML , Robot and more

A Sample Code

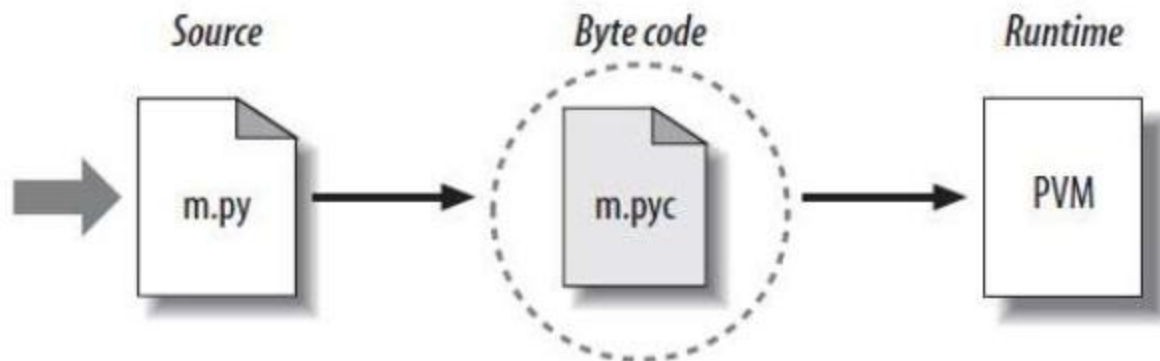
```
x = 34 - 23          # A comment.  
y = "Hello"          # Another one.  
z = 3.45  
if z == 3.45 or y == "Hello":  
    x = x + 1  
    y = y + "World"  # String concat.  
print x  
print y
```

Enough to understand the code

- Indentation matters to code meaning
 - Block structure indicated by indentation
- First assignment to a variable creates it
 - Variable types don't need to be declared.
 - Python figures out the variable types on its own.
- Assignment is `=` and comparison is `==`
- For numbers `+` `-` `*` `/` `%` are as expected
 - Special use of `+` for string concatenation and `%` for string formatting (as in C's `printf`)
- Logical operators are words (`and`, `or`, `not`) *not* symbols
- The basic printing command is `print`

Python Code Execution

- Python's traditional runtime execution model: source code of any type is translated to byte code, which is then run by the Python Virtual Machine. Your code is automatically compiled, but then it is interpreted.



Source code extension is **.py**

Byte code extension is **.pyc** (compiled python code)

Running Python

Once you're inside the Python interpreter, type in commands at will.

- **Examples:**

```
>>> print 'Hello world'
```

```
Hello world
```

Relevant output is displayed on subsequent lines without the >>> symbol

```
>>> x = [0,1,2]
```

Quantities stored in memory are not displayed by default

```
>>> x
```

If a quantity is stored in memory, typing its name will display it

```
[0,1,2]
```

```
>>> 2+3
```



?

suniljoshi.cse@satengg.org



Thank You