



BITS Pilani

Pilani | Dubai | Goa | Hyderabad

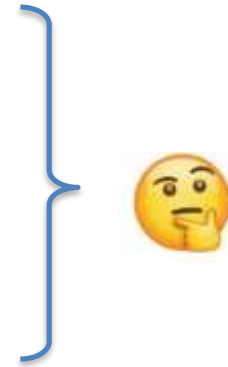
Introduction to Python for Data Science

DSECLPFDS

Parthasarathy

Agenda for CS #1

- 1) Ground Rules
- 2) Introduction to *DSECLPFDS*
 - Motivation & Objective of *DSECLPFDS*
 - Courseware
 - Books & Evaluation components
 - Pedagogy for DSECLPFDS ?
- 3) Course Schedule
- 4) Getting started with Module 1



Ground Rules!



- Mentally present – Observe!! Listen!!
- Keep your questions for the Q&A section / Discussion Forum
- Use the Discussion Forum in Canvas effectively
- Solve the exercises regularly!
- Go that “extra mile” 😊

$$1^{365} = 1$$

$$1.01^{365} = 37.8$$

Motivation for this course ?



Motivation

- As of now, Python is one of the most widely used programming languages in the Data Science field.
- Data Scientists just love Python! ❤️
- Python is easy to learn & has a great community for support!
- We would use Python for all the assignments / case-studies (For all the subjects in MTech DSE).

Course Objectives




What is this course about ?

- Introduce the fundamental programming concepts of Python
- Enable you to solve data problems using Python
- Act as a kick-start / bridge for participants of the MTech DSE programme who are *new* to Python.

What is this course *not* about ?

- Comprehensive, in-depth discussion about Python programming.
- Comprehensive, in-depth discussion about data analysis using Python and related packages, libraries, and tools.

- Available at : <https://bits-pilani.instructure.com/courses/1078/files/206566/download>


BITS Pilani
Pilani | Dubai | Goa | Hyderabad

Introduction to Python for Data Science

Table of Contents

- Course Objectives
- Modular Course Structure
- Text & References
- Additional Readings
- (Sample) Labs / Assignments

Course Objectives

CO1	Introduce students with fundamental programming concepts of Python
CO2	Enable students to solve data problems using Python

Text and References

T1	Charles Severance: Python for Everybody, Exploring Data in Python 3, Creative Commons, 2016
T2	Jake VanderPlas: Python Data Science Handbook, Essential Tools for Working with Data, O'Reilly Media, 2016
T3	Edouard Duchesnay: Statistics and Machine Learning in Python Release 0.2, 2018
T4	Wes McKinney: Python for Data Analysis, Agile Tools for Real World Data, O'Reilly Media, 2013

Additional Reading

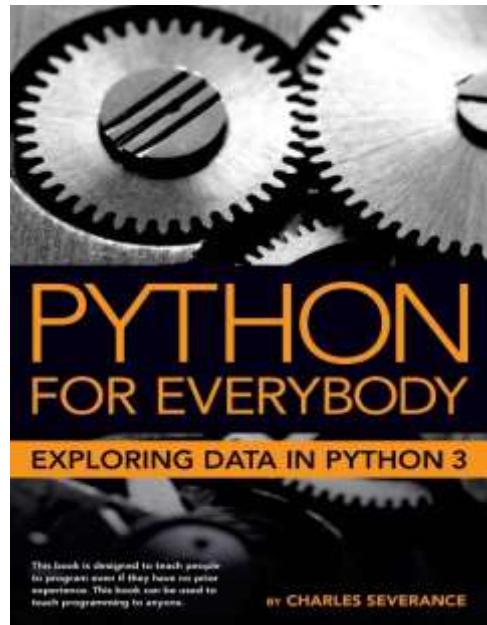
- [Python 3.* documentation](#)
- [Numpy Documentation](#)
- [Pandas Documentation](#)
- [Matplotlib documentation](#)
- [seaborn: statistical data visualization documentation](#)
- [Scikit-learn documentation](#)

1 | Page

Modular Content Structure		
Session	Topics	Reference
Saturday, 16 th Oct 2021		
1	Python Basics	
1.1	Setting up Python Environments Anaconda Distribution Spyder IDE Jupyter Notebooks Input / Output with Python	Python Documentation
1.2	Getting familiarity with basic code constructs Package imports Data Types & Type Casting Variables, Expressions & Statements	T1 : Ch 2, Class Notes
Sunday, 17 th Oct 2021		
2	Python Data Structures	
2.1	Immutable Data Structures Immutable Data Structures Strings Operations on String Familiarity with Tuples	T1 : Ch 6, 10, Class Notes
2.2	Mutable Data Structures List List operations Familiarity with Sets Dictionary operations	T1 : Ch 8, 9, Class Notes
3	Python Programming Constructs	
3.1	Expressions, Operations, and Decision Structures Boolean Expressions and Logical Operators Conditional and Alternative execution Chained and Nested execution Catching Exceptions with try and except	T1 : Ch 2, 3, Class Notes
Saturday, 23 rd Oct 2021		
3.2	Iterative Executions While loops Infinite loops, break, continue For loops Loop patterns	T1 : Ch 5, Class Notes
Self Study : Object Oriented Features supported by Python		

2 | Page

Text Books



[Charles Severance: Python for Everybody, Exploring Data in Python 3](#)



[Jake VanderPlas: Python Data Science Handbook](#)

eBooks of both are made available in Canvas

Note: These are the prescribed ones. Please feel free to explore any Python materials that suits you.

Evaluation Components



- This course is **NOT evaluated** !!
- You will **not** have any exams for this course 😊
- Nevertheless, there would be some exercises for you to try and hone your skills.
- Fast-Paced sessions!
 - As this is not a semester course and is only a bridge course, the courses will be medium to fast – paced.
 - Please use the recording / speed option in Impartus to align it to your pace 😊

Pedagogy for this Course



Step 01: Class Session

- **We** learn Fundamentals !
- Look at few examples for each concept.

Step 02: Explore

- **You** explore the additional notebooks. Get your hands dirty with Python
- Practice more examples for each concept.

Step 03: Doubts

- Put your queries in Discussion Forum.
- Peers and TA to answer ...

Non-Beginners: You can directly start with Step 02 and also use this phase for additional learning which might help in future ... You play an important role in Step 3 as well in answering your peer's queries.

Course Schedule



16/10/2021 – S1 (9AM)	17/10/2021 – S2 (9AM)	23/10/2021 – S3 (9AM)
<ul style="list-style-type: none">○ Motivation & Agenda○ Python Basics○ Setting up Python Environment○ Getting familiarity with basic code constructs	<ul style="list-style-type: none">○ Python Data Structures○ Immutable Data Structures○ Mutable Data Structures○ Expressions, Operations & Decision Structures	<ul style="list-style-type: none">○ Iterative Constructs○ Functions○ Files
24/10/2021 – S4 (9AM)	30/10/2021 – S5 (9AM)	31/10/2021 – S6 (2PM)
<ul style="list-style-type: none">○ SciPy Ecosystem○ NumPy○ Pandas Basics	<ul style="list-style-type: none">○ Data Exploration with Pandas○ Visualization with Matplotlib	<ul style="list-style-type: none">○ Visualization with Seaborn○ Brief Introduction to scikit-learn

Program & Programming Language



Computer Program

- Set of instructions that perform a specific task executed by computer
- Required by computer to function
- Written by programmer using programming languages
 - Like C, C++, Java, Python etc.
- Executed with compiler and interpreter

Python as a Programming Language

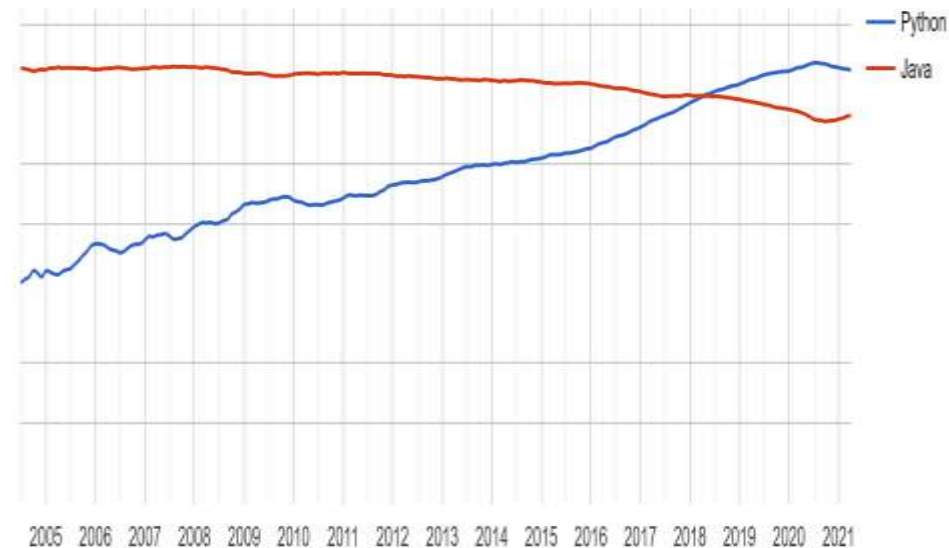


Why Python ?

Worldwide, Oct 2021 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	29.66 %	-2.1 %
2		Java	17.18 %	+0.8 %
3		JavaScript	8.81 %	+0.4 %
4		C#	7.3 %	+1.1 %
5	↑	C/C++	6.48 %	+0.7 %
6	↓	PHP	5.92 %	+0.1 %
7		R	4.09 %	+0.2 %
8		Objective-C	2.24 %	-1.2 %
9	↑	TypeScript	1.91 %	+0.1 %

PYPL Popularity of Programming Language



Worldwide, Python is the most popular language ...

Source : <http://pypl.github.io/PYPL.html>

Python as a Programming Language



Python

- Designed by Guido van Rossum around 1990
- Not just a scripting language
- Easy to learn, read, use
- Extensible (add new modules)
- Highly readable
- Latest Version 3.9
- Most fond of language for Data Scientists

Touchy Feel Properties

- Open Source
 - copyrighted but use not restricted
 - owned by independent non-profit, PSF
- Mature (29 years old)
- Supportive user community
 - plenty of good books, too
 - Active user community
- Simple design, easy to learn
 - reads like “pseudo-code”
 - Suitable as first language
 - Suitable as last language :-)
(Hopefully)

Python Applications



Use Python for...

Web Development: Django , Pyramid , Bottle , Tornado , Flask , web2py

GUI Development: tkinter , PyGObject , PyQt , PySide , Kivy , wxPython

Scientific and Numeric: SciPy , Pandas , IPython

Software Development: Buildbot , Trac , Roundup

System Administration: Ansible , Salt , OpenStack

Python Ecosystem



Components of Python World :

- Core Python
- Distributions
- Frameworks / IDEs
- Third party Libraries

Core Python

- Programming Language itself
- Some standard modules are available
- Other packages needs to be explicitly installed

Python Distribution

- Python + packages
- Majority of packages, libraries are already available
- Package management is simplified
 - Anaconda from Continuum Analytics
 - IPython and its IPyKit variant

Python Ecosystem



Frameworks / IDEs

- Use frameworks to create code and develop applications
- Provides a defined structure to the developers so that they can focus on the core logic of the application rather than on other elements
- Python web framework
 - ✓ Django
 - ✓ Web2py
 - ✓ Flask
- Python IDEs
 - ✓ IDLE
 - ✓ PyCharm
 - ✓ Spyder
 - ✓ Jupyter Notebooks

Third party Libraries

- Makes life of developers very simple
- Just need to know the right library to carry out a task
 - NumPy
 - Scipy
 - Pandas
 - Matplotlib
 - Seaborn
 - Bokeh
 - ScikitLearn
 - And List goes on ...

Python Installation



Three Ways :

- Install Python directly
 - Install the Python language with installer
 - Need to install other packages explicitly using pip install
 - <https://www.python.org/downloads/>
- Use Python distribution
 - The open-source Anaconda Distribution is the easiest way to perform Python coding
 - Works on Linux, Windows, and Mac OS X
 - <https://docs.anaconda.com/anaconda/install/windows/>
- Use Cloud based services
 - The simplest of all but needs internet connectivity to use
 - Microsoft Azure Notebooks
 - Google Collab

Integrated Development Environments (IDE) for Python



Common IDE's:



Our Favourite (For DSE):



- **.py** is a regular python file. It's plain text and contains just your code.
- **.ipynb** is a python notebook and it contains the notebook code, the execution results and other internal settings in a specific format.

Input / Output with Python



- `print()` can be used to output a message
- `input()` can be used to enter an input to the python program.
- `#` can be used to provide comments.
- `"""`(triple quotes) can be used to write documentation.

Demo:

- Let's see how to launch Jupyter Notebook
- See the basics of Notebook
- Practice some I/O statements and comments.

Basic Code Constructs



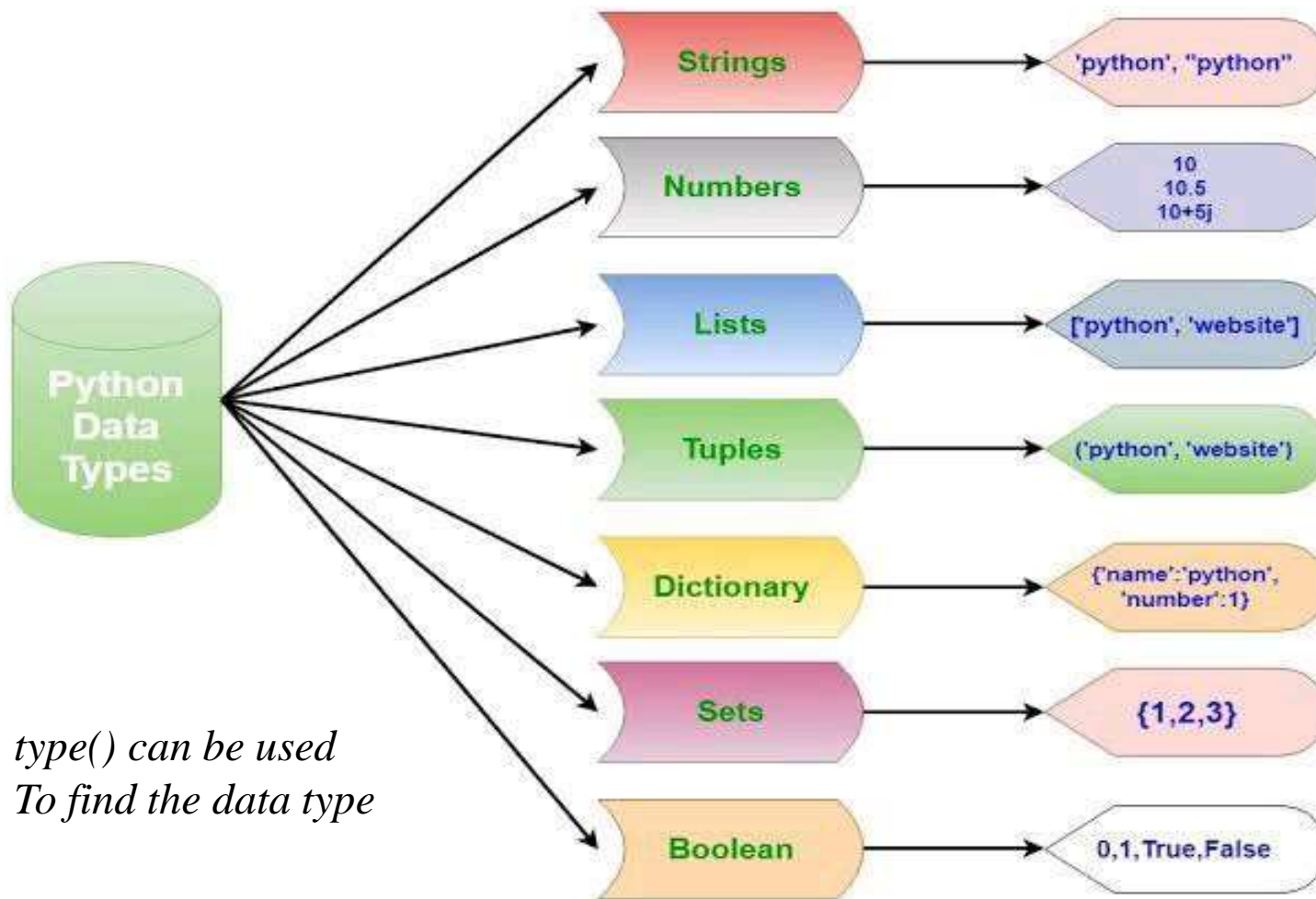
Imports:

- Import in Python is similar to `#include` in C/C++. Python modules can get access to code from another module by importing the file/function using `import`.
- Ex: `import math`
- `print(math.pi)`

Variable

- A Python variable is a reserved memory location to store values. In other words, variables are containers for storing data values.
- *Python has no command for declaring a variable.*
- A variable is created the moment you first assign a value to it.
- Ex: `a = 100`

Data Types in Python



Immutable



*type() can be used
To find the data type*

Data Types in Python

Name	Type	Description
Integers	int	Whole numbers, such as: 3 300 200
Floating point	float	Numbers with a decimal point: 2.3 4.6 100.0
Strings	str	Ordered sequence of characters: "hello" 'Sammy' "2000" "楽しい"
Lists	list	Ordered sequence of objects: [10,"hello",200.3]
Dictionaries	dict	Unordered Key:Value pairs: {"mykey": "value", "name": "Frankie"}
Tuples	tup	Ordered immutable sequence of objects: (10,"hello",200.3)
Sets	set	Unordered collection of unique objects: {"a","b"}
Booleans	bool	Logical value indicating True or False

type() can be used to find the data type



Post your queries in the Discussion Forum!!

Feedback

😊 👍 : 5

😏 🙅 : 3

😞 👎 : 1

Thank You for your
time & attention !

Contact : parthasarathypd@wilp.bits-pilani.ac.in