PYTHON
Python Beginners: Level 1

Nature of the Course: Theory + Practical
Total Hours per Day: 2 Hours
Course Duration: 4 Weeks

Course Summary

The DTC - Python 1 course is targeted for beginners who want to learn how to think and write meaningful pieces of codes or read codes written by someone else in Python. This course teaches how to map literary description of a problem (requirement) to an application/library coded in Python. This is a core basic level course that is essential for anyone who has no prior programming experience but wishes to be a professional Android engineer in future.

Completion Criteria

After fulfilling all of the following criteria, the student will be deemed to have finished the Module:

- Has attended 90% of all classes held.
- Has received an average grade of 80% on all assignments
- Has received an average of 60% in assessments.
- The tutor believes the student has grasped all of the concepts and is ready to go on to the next module.

Required Textbooks

- Eric Matthes, "Python Crash Course: A Hands-On, Project-Based Introduction to Programming", No Starch Press.
- John Zelle, "Python Programming: An Introduction to Computer Science", Franklin, Beedle & Associates Inc.

Prerequisites

- Fundamental understanding of programming, bits/bytes, procedures, classes, and computer architecture. It's absolutely acceptable if you

only have a theoretical understanding of programming, but you should be certain about what programming is and what you intend to gain from this session.

- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in Python (self-study and practice).
- If you are only interested in theory and have no interest/patience in spending at least 10 hours every week throughout the duration of the course, then this course might not be for you.
- If you have absolutely no idea about programming or do not see yourself doing programming in the next six -odd months, then this class may not be for you.

Course Details
Week I
Overview Of Python Language

- Introduction
- Hardware and Software Requirements
- Installation of Python
- Using an Interpreter

Core Data Structures

- Strings, Variables
- Tuples
- List
- Dictionary
- Operation on Data Structures
- Slicing

Constants, Variables And Data Types

- Primitive / Non-Primitive Variables

Week II
Decision And Branching

- If, Else, Switch, Break, Continue

Looping
- For, While, Do – While

Functions
- Building Modules
- Functions
- Function Types
- Lambdas
- Map / Filter
- Comprehension
- For, While, Do – While

Week III
Exception Handling
- Introduction
- Handling Exceptions
- Raising Exceptions
- Catching Exceptions
- Chaining Exceptions

Object – Oriented Programming (Oop)
- Introduction to Class / Objects
- Writing a Class
- Inheritance
- Polymorphism
- Encapsulation
- Operator Overloading
- Working with Database

Labs
Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem-solving techniques to the students.

Python Intermediate: Level 2

Nature of the Course: Theory + Practical
Total Hours per Day: 2 Hours
Course Duration: 4 Weeks

Course Summary

The DTC – Python – Level 2 course is designed for students who have some prior hands-on programming experience with the Python programming language at a beginning level. This course is ideal for people who have previously programmed in another programming language (e.g., Java, Obj-C, PHP, C, C++, etc.) and wish to learn Python. This course is designed for high school and university students who want to do Python coursework, including those who have worked in the media industry since graduation or are working as a professional freelance PHP developer.

Completion Criteria

After fulfilling all of the following criteria, the student will be deemed to have finished the Module:

- Has attended 90% of all classes held.
- Has received an average grade of 80% on all assignments
- Has received an average of 60% in assessments.
- The tutor believes the student has grasped all of the concepts and is ready to go on to the next module.

Required Textbooks

- Eric Matthes, "Python Crash Course: A Hands-On, Project-Based Introduction to Programming", No Starch Press.
- John Zelle, "Python Programming: An Introduction to Computer Science", Franklin, Beedle & Associates Inc.

Prerequisites

- Successfully complete the entrance test with a score of at least 40% (for trainees directly applying to this level).
- Successfully complete the DWIT Training - Python – Level 1 course (not applicable to trainees directly applying to this level).
- Successfully complete the interview.

- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in Python (self-study and practice).

Course Details
Week I
Web Application Basics
- How the Web Works
- Overview of Django
- Django Philosophies
- Project

Getting Started
- Introduction to Projects and Apps
- Creating a Simple Web Application
- Introduction to Django Admin
- Using Django Admin
- Using Applications

Week II
Views And Templates
- Introduction to Views and Templates
- How Views and Templates Work
- Introduction to Bootstrap
- Bootstrap for UI Design
- Introduction to Django Forms
- Using Django Forms

Week III
Models
- Introduction to Models
- How Models Work
- Using Databases
- Making Queries
- Database Relationships
- Sessions

Week IV
Restful Api
- Introduction to REST API
- Applications of REST API
- Creating a REST API for an Application
- Using REST API in UI

Extras
- Conclusion and Discussions on the Topic

Labs
Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem-solving techniques to the students.

Python Advanced: Level 3

Nature of the Course: Theory + Practical
Total Hours per Day: 2 Hours
Course Duration: 4 Weeks

Course Summary
This course expands on the DTC – Python – Level 2 foundation and offers advanced subjects to equip learners for a career as an Android software engineer.

Completion Criteria
After fulfilling all of the following criteria, the student will be deemed to have finished the Module:
- Has attended 90% of all classes held.
- Has received an average grade of 80% on all assignments
- Has received an average of 60% in assessments.
- The tutor believes the student has grasped all of the concepts and is ready to go on to the next module.

Required Textbooks
- Luciano Ramalho, "Fluent Python: Clear, Concise, and Effective Programming", O'Reilly.
- Mark Lutz, "Programming Python: Powerful Object – Oriented Programming", O'Reilly.

Prerequisites
- Successfully complete the entrance test with a score of at least 40% (for trainees directly applying to this level).
- Successfully complete the DWIT Training - Python – Level 2 course (not applicable to trainees directly applying to this level).
- Successfully complete the interview.
- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in Python (self-study and practice).

Course Details

Week I

Introduction To Python For Data Science

- Python Basics
- Data Types

Python Lists

- Storing Different Data Types under a Single Name
- Create, Subset and Manipulate
- Lists

Functions And Packages

- Importing Python Packages
- Calling Functions
- NumPy
- Writing Numerical Python
- Creating Visualizations
- Calculations with Large Datasets

Week II

Matplotlib

- Creating Visualizations
- Building Complex & Customized Plots

Control Flow & Pandas

- Writing Conditional Constructs
- Pandas Dataframe
- Data Structure for Data Science

Python For Data Science (Advanced)

- Getting Started with Data Science
- Data Science: Generating Value from Data
- The Data Science Process

Background In Python And Unix

- Key Data Structures

Jupyter Notebooks And Numpy
- Jupyter Notebooks
- NumPy Advanced Practice
- Satellite Image Application in NumPy

Pandas
- Working with Pandas

Week III
Data Visualization
- Introduction to Data Visualization
- Case Studies
- Matplotlib and Other Libraries

Introduction To Machine Learning
- Regression Classification
- Clustering
- Analysis

Working With Text And Databases
- Working with Databases
- Natural Language Processing
- Working with Text

Final Project

Labs
Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem-solving techniques to the students.

Learning Outcomes
- Learn how to set up Python and develop a simple application.
- Declare and perform operations on simple data types, including strings, numbers, and dates.
- Declare and perform operations on data structures, including lists, ranges, tuples, dictionaries, and sets.
- Write conditional statements and loops.

- Define and use functions, classes, and modules
- Learn how to design object‑oriented programs with Python classes.
- Learn how to use class inheritance in Python for reusability.
- Learn how to use exception handling in Python applications for error handling.