Unwired Learning

The Complete Python Developer Roadmap 🔗 (Course Curriculum)

Any doubt in mind?
Whatsapp us at 7014963730

(Check Page 26-27 For Certificate Related Information)

💻 **Enroll Now**

https://courses.unwiredlearning.com/s/store/courses/description/python-roadmap

Module - 1

Python Programming Course Introduction

- Introduction
- Welcome - Let's Get Started!

Setup - Installing Python

- Python Programming in 2022 - Is Python A Bubble?
- Installing Python on Windows
- Python 3.10 Version Update
- Different Methods To Execute Python Codes
- Writing Our First Python Program
- Variables and Python Memory Management

Data Types in Python

- Data Types in Python
- Sequences in Python
- Sets, Dictionary
- Literals and Identifiers
- Reserve Words - You Can't Use Them!

Operators in Python

- Operators: Arithmetic, Assignment
- Operators: Unary Minus, Relational, Logical, Boolean
- Understanding Escape Characters

Input and Output

- Introduction to Input and Output Statements
- Output Statements
- Input Statements
- Python Built-in Functions and Built-in Module

IDE (Integrated Development Environment)

- Editor 1 - Working With ATOM
- Editor 2 - Working With VS Code
- Run .py File Through System Terminal

Python Control Statements

- Conditionals: If, If ... Else and Indentation
- Conditionals: If ... Else Statements With Operators
- Conditionals: if... elif ... else Statements + Nested if Statements

Loops In Python

- While Loop
- For Loops - Part 1
- For Loops - Part 2
- Break and Continue Statements

Strings and Characters

- Comments and Doc Strings
- Diving Deep With "Strings"

Lists, Tuples and Dictionaries

- Diving Deep - List
- Diving Deep - Tuples
- Diving Deep - Dictionaries
- Indexing, Slicing, Negative Indexing

Functions

- What are Functions?
- Parameters, Arguments, Return
- Formal and Actual Arguments (arg, *arg, **karg)
- Local and Global Variables

Object Oriented Programming (OOP)

- Introduction to Object Oriented Programming (OOP)

- Classes and Objects in Python (OOP)
- Understanding init() Method and 'self ' Parameter
- Solving Task: OOPS
- Defining Multiple Constructors in Python?
- Encapsulation
- Public And Private Methods
- Inheritance
- Getter and Setter
- Creating And Importing Module
- Creating User Defined Module
- Multiple Inheritance
- Understanding super() Function [Part 1]
- Understanding super() Function [Part 2]
- Python Naming Convention (Classes, Variables, Functions, Methods)
- Composition
- Aggregation
- Abstract Classes
- Discussing Over import And from
- Operator Overloading [Part 1]
- Operator Overloading [Part 2]

Object Oriented Programming - Theory

- What Is Object-Oriented Programming (OOP)?
- Classes and Objects in Python (OOP)
- Encapsulation in Python (OOP)
- Inheritance in Python (OOP)

Exercise - Python Practice

- 10 Quiz Questions - Python Programming Basics
- 10 Quiz Questions - Object-Oriented Programming

Errors And Exceptions Handling

- Errors - Types of Errors!
- Exceptions
- Exceptions Handling Introduction
- Exceptions Handling

- Try, Except, Else And Finally
- Raising Exception
- Creating User-Defined Exception

Special Variables

- if __name__ == "__main__"

Python I/O - File Handling

- Creating Text File And Write Content
- Appending Files | Solving Example - File Handling
- How To Read Content From A File

Exercise - Python Practice

- 30 Exercise (Coding Question Round 0)

Python Package Management System

- Python Package Management System

Project - Face Detection

- What is OpenCV?
- OpenCV Face Detection With Python
- Detecting 'Elon Musk' And 'Mark Zuckerberg' Faces
- OpenCV Face Detection - Project Code
- Detecting Faces Of All Images In The Folder

Exercise - Assignment

- Other Detection Using OpenCV

Self Dependent Developer

- How to become a Self-Dependent Developer

Project - Password Generator

- Random Module

- Random Password Generator
- Readable Password Generator

Exercise - Python Practice

- 10 Exercise (Coding Question Round 1)
- 10 Exercise (Coding Question Round 2)

Python 3 Cheat Sheet

- Python 3 Cheat Sheet
- PEP 8 - Style Guide for Python Code

Module - 1 / Part 2

Python Programming - Advanced Concepts

- Introduction

Recursion

- What is Recursion?
- Control of a Function
- Tracing Tree
- Call Stack
- Tree Recursion
- Example: Factorial of a Number

Map, Filter and Reduce

- Lambda Functions
- Map
- Filter
- Reduce

Map, Filter and Reduce

- List Comprehensions

Regular Expressions

- RegEx 1 - Why We Use Regular Expressions
- RegEx 2 - Different Methods With Regular Expressions
- RegEx 3 - Writing Patterns
- RegEx 4 - Creating Pattern For Email Validation

Other Important Topics

- Decorators
- Logging
- Date and Time

What's New With Python 3.8?

- Walrus Operator
- Positional Only Parameters
- F-strings For Debugging

What's New With Python 3.9?

- Union Operators
- Type Hinting
- New String Class Methods
- Native Time Support (Zoneinfo)

Exercise - Python Practice

- 45 Theory Questions - (Basics + Intermediate)
- 15 Theory Questions - Advanced
- 10 Exercise (Coding Question Round 3)
- 10 Quiz Questions - Python Programming Advanced Concepts

Module Completion

- Well Done!

1 - Course Completion Certificate [Module 1] 🎉

- Complete Python Programming - Basics to Advanced Concepts

Module - 2

Project - Building Portfolio Desktop Application With Tkinter (GUI)
● Module Introduction
● What is API?
● What are API and Request Methods?
● Working with CoinMarketCap API
● Fetching API Data
● Extracting API and Coin Data
● Creating Coin List & Understanding Portfolio
● Creating Portfolio Dictionary
● Calculating Profit and Loss
● GUI Introduction - Graphical User Interface
● Understanding 3 Steps Tkinter Working
● Tkinter Main Window and Labels
● Application Formatting - Adding Header
● Application Formatting - Adding More Data
● Application Formatting - Improving User Interface
● Color Indication For Profit and Loss
● Adding Update Button
● Creating Executable App (.exe file)

Project - Integrating Database With GUI Application
● Databases - SQL vs SQLite3
● Introduction to SQLite3 and its Workings
● Creating Table | Working with DB Browser
● Insert Values - Databases
● Update Values - Databases
● Fetching Data From Database
● Delete Data - Database
● Creating Functions For Operation - Database
● Connecting With Database
● Fetching Data From Database
● Modifying Current Application

- Adding "Add Coin" Button
- Adding "Update Coin" - "Delete Coin" Button
- Solving Multiple Layer Issue
- Adding Notification System - Popup Box
- Adding Navigation System - Menu Bar

Module - 3

Project - Building Twitter Bot With Python and Tweepy

- Section Introduction
- Creating Twitter Developer Account
- Creating App and Generation Access Tokens
- Installing Tweepy
- Update Status Through Bot
- Introduction To Auto-Reply
- Working With Timeline and Mentions
- Iterate Tweets And Add Condition
- Storing Replied Tweet ID's
- Storing-Accessing Last Seen ID
- Replying Tweet
- Formatting Function and Testing Bot
- Auto Retweet and Auto Like
- Setting-up PythonAnyWhere
- Deployment Done Along Testing
- Auto Retweet To Particular Hashtag
- Implementing Error Handling And Testing Bot

Exercise - Assignment

- Python Image Optimization and Transformations

Module - 4

Project - Building Web Scraping Bot With Beautiful Soup

- Course Introduction
- Web Scraping Project Demo
- How Do We Scrape Data?
- Web Scraping - Overview
- Project Setup + Installing Libraries
- Working With BeautifulSoup
- Filtering Required Data
- Extracting Current Price
- Extracting Table Content - I
- Extracting Table Content - II
- Extracting All Stock Result
- Working with Static File
- Storing Stock Content in CSV File
- Sending Text Mail Through SMTPLIB
- Sending Text Mail Through Email Module
- Sending Attachment Through Email Module
- Integrating Mail System with Web Scraper
- File Name According To Today's Date

Exercise - Assignment

- Web Scraping Encyclopedia Article

Module - 5

Project - Data Analysis With Pandas

- What is Pandas?
- Starting With Pandas And iPython
- Working with Jupyter Notebooks
- Important Jupyter Notebook Commands
- Working with CSV, Excel, TXT and JSON Files
- Working with API Response
- Indexing and Slicing Dataframe Tables [Part 1]
- Indexing and Slicing Dataframe Tables [Part 2]
- Deleting Columns and Rows
- Adding and Updating new Columns and Rows

Exercise - Pandas Questions

- Common Pandas Questions - Theory

Module - 6

Project - Automating Instagram Post Designing

- Course Introduction
- OpenWeatherMap API
- Working With API Data
- Working With Pillow Library
- Adding Content: Date and Time
- Adding Content: Multiple City Position
- Adding Content: Multiple City Data
- Adding Content: Multiple Country Data
- Saving Post as PNG and PDF

Exercise - Assignment

- Exercise: Automate Designing For Instagram Stories

Module - 7 / Part 1

Project - Backend Development with Django (Web Application With Python)

- Project Introduction
- Welcome - Let's Get Started!
- Understanding Django
- Setting Up Text Editor - VSCode
- Setting Up Virtual Environment
- Django 2.2 Version Update
- Startapp - Taskmate
- Runserver For First Time
- Django Flow and Django Structure
- Urls and Views
- Templates
- Bootstrap
- Base Template and Jinja 2
- Static Folder and Improving Website Look
- Admin Panel
- Models
- Database Migrations
- Fetch Data From Database
- Displaying Data On Templates
- Adding Condition On Data
- Adding Form and Accepting Input
- Adding Messages and Alert Option
- Close Option On Alert
- Understanding CSS and Bootstrap
- Deleting Queryset Item
- Editing Queryset Item - 1
- Editing Queryset Item - 2
- Editing Queryset Item - 3
- Adding Option - Mark Task As Completed
- Adding Option - Mark Task As Pending
- Pagination Overview
- Implementing Pagination - 1

- Implementing Pagination - 2
- Implementing Pagination - 3
- Fixing URLs and Links
- Fixing Task Page Design - 1
- Fixing Task Page Design - 2
- Designing Home Page - 1
- Designing Home Page - 2

Module - 7 / Part 2

Project - Integrating Accounts & Authentication on Django Application

- Introduction - Django Authentication System
- Creating User Section
- Registration Form
- Register Templates
- Register Views [Part I]
- Register Views [Part II]
- Adding Email Field In Forms
- Improving Form Design With Crispy Forms
- Adding Grid Layout On Registration Page
- Login - URLs, Views & Template
- Login - Functionality Test
- Login - Redirect URL
- Logout
- Quick Design Change
- Adding Header Restriction [Part I]
- Adding Header Restriction [Part II]
- Adding Page Restrictions
- Relationship Between Task & User
- Adding Foreign Key
- Updating Task Input Fields
- Task Of Logged-in User Only
- Security Problem & Solution [Part I]
- Security Problem & Solution [Part II]

Module - 7 / Part 3

Project - Deploying Django Web Application on Cloud Application Platform (Heroku)

- 4 Important Pillars to Deploy
- Registering on Heroku and GitHub
- Creating GitHub Repository
- Working with requirements.txt and .gitignore
- Understanding Django Environ
- Working with Environment Variables
- Solving .env Errors [Part 1]
- Solving .env Errors [Part 2]
- Push project from Local System to GitHub
- Working with Django Heroku
- Working with StaticRoot
- Handling WSGI with gunicorn
- Empty folder Staticfiles
- Configuring Secret Keys
- Setting up Database and adding users

Exercise - Django Practice Questions

- 10 Quiz Questions - Django Framework
- 20 Theory Questions - Django Framework Interview

2 - Course Completion Certificate [Module 7] 🎉

- Django Development Bootcamp: Build & Deploy Web Application With Python & Django

Module - 8

Project - Django REST Framework - Build Powerful API Using Python

- Course Introduction
- Basic API Concepts - Theory
- Understanding API
- Understanding REST API
- API With Django
- Basic Django Setup
- Installation
- Models and Migrations
- Creating JSON Response - All Elements
- Creating JSON Response - Individual Elements
- Code Source - Github

Django REST Framework

- Django REST Framework
- Understanding DRF
- DRF Project Source Code - Github
- Views and Serializers
- Serializers - GET Request
- Serializers - POST, PUT, DELETE Request
- Status Codes
- APIView Class
- Validation
- Serializer Fields and Core Arguments
- Model Serializer
- Custom Serializer Fields
- Updating Models
- Django Relationships
- Nested Serializers
- Serializer Relations
- HyperLinked Model Serializer
- Serializer Relations
- GenericAPIView and Mixins
- URL Structure
- Concrete View Classes

- Overwrite Queryset
- Viewsets and Routers
- ModelViewSets
- Postman and Project Update
- PostmanUser Model
- Temporary Login and Logout
- Permissions
- Introduction to Permissions
- Custom Permissions
- Custom Calculation
- Authentication
- Introduction to Authentications
- Basic Authentication
- Basic Authentication
- Token Authentication
- Token Authentication - Part 1
- Token Authentication - Part 2
- Token Authentication - Part 3 (Login)
- Token Authentication - Part 4 (Registration)
- Token Authentication - Part 5 (Registration)
- Token Authentication - Part 6 (Logout)
- Manual Testing Entire Project ondemand_video
- Manual Testing Entire Project - Part 1
- Manual Testing Entire Project - Part 2
- JWT Authentication
- JWT Authentication - Access Token and Refresh Token
- JWT Authentication - Login
- JWT Authentication - Registration
- Throttling
- Throttling Introduction
- Throttle Rate (Anon and User)
- Throttle Rate (Custom and Scope)
- Filtering, Searching, Ordering
- Filtering Introduction
- Filter, Search, Ordering
- Project Update
- Pagination
- Pagination Part 1 - PageNumber
- Pagination Part 2 - LimitOffset

- Pagination Part 2 - Cursor
- Browsable API Update
- Automated API Testing
- API Testing - Registration
- API Testing - Login and Logout
- API Testing - StreamPlatform
- API Testing - WatchList
- API Testing - ReviewTestCase
- API Testing - UserTestCase
- About Test Driven Development
- Project Completed
- Project Source Code
- DRF Project Source Code - Github
- (Optional) - Code Clean Up!
- (Optional) - Working With Structure
- DRF Project Source Code - Github
- Thank You For Being Here!

3 - Course Completion Certificate [Module 8] 🎉

- Django REST Framework - Build Powerful API Using Python

Module - 9

Learning Version Control and Code management with Git and Github

- Git Introduction And Version Control System
- What is Git?
- Downloading And Installing Git
- Local Configuration Git
- How To Start Our Project?
- Creating Our First Repository
- Basic Git Commands
- Git Commands - Post
- Git Workflow
- Comparing Changes
- Solving Git Error Using StackOverflow
- Reverting Changes
- Branching
- Merging Different Branches
- Deleting Branches
- List of Git Commands for Working with Branches - Post
- Ignoring Unwanted Files and Folders
- Stashing In Git
- Understanding HEAD
- Assisting On Open Source Project
- Pull Request
- A Quick Recap

Exercise - Git and GitHub Practice Questions

- 10 Quiz Questions - Git and GitHub
- 10 Theory Questions - Git and GitHub Interview Theory
- Git and GitHub Assignment

4 - Course Completion Certificate [Module 9] 🎉

Git & GitHub for Beginners: Mastering the Modern Workflow

Module - 10

Course Introduction

- Course Introduction
- Curriculum Walkthrough

Big O Notation

- Section Introduction
- Complexity Analysis
- Why do we need Big O Notation?
- Big O(n) Complexity
- Big O(1) Complexity
- Counting Operations
- Simplifying Big O - Part 1
- Big O(n^2) Complexity
- Simplifying Big O - Part 2
- Big O(n!) Complexity
- Space Complexity
- Space Complexity - II
- Section Summary

Essential Concepts - I

- Memory
- Logarithm

Data Structure - Introduction

- Introduction to Data Structures

Data Structures - Array

- Array Introduction
- Array - Common Operations I
- Array - Common Operations II
- Static vs Dynamic Array - Common Operations III

Data Structures - Linked List

- Linked List
- Linked List Complexities
- Doubly Linked List
- Circular Linked List

Data Structures - Stack and Queue

- Stack and Queue

Data Structures - Hash Tables

- Hash Tables

Data Structures - Trees

- Tree - Part 1
- Tree - Part 2
- Binary Tree
- Types of Binary Tree
- Binary Search Tree
- AVL - Red Back Tree

Data Structures - Heaps

- Heaps
- Heap Sort and Priority Queue

Data Structures - Trie

- Trie - I
- Trie - II
- Why are Tries Important?

Data Structures - Graph

- Graph

Essential Concepts - II

- What is Recursion?
- Recursion: Control of a Function
- Recursion: Tracing Tree
- Recursion: Understanding Call Stack
- Recursion: Tree Recursion
- Recursion Example - Factorial of a Number
- Practice Questions

Algorithm: Searching

- Linear Search
- Binary Search
- Binary Search Complexity
- Binary Search Implementation
- Binary Search Implementation - Recursion

Algorithm: Sorting Elementary

- Sorting Algorithm Introduction
- Bubble Sort
- Bubble Sort Visualization
- Bubble Sort Implementation
- Bubble Sort Complexity
- Selection Sort
- Selection Sort Visualization
- Selection Sort - Implementation
- Selection Sort - Complexity
- Insertion Sort
- Insertion Sort Implementation
- Insertion Sort Complexity
- Performance Analysis

Algorithm: Sorting Advanced

- Divide and Conquer Algorithms
- Quick Sort
- Quick Sort Complexity
- Quick Sort Implementation
- Merge Sort

- Merge Sort Complexity
- Merge Sort Implementation

Algorithms: Tree Traversal

- Tree Traversal
- Depth First Search - Preorder Inorder Postorder
- Binary Tree Implementation
- Depth First Search - Implementation
- Depth First Search - Complexity
- Breadth First Search - Level Order
- Breadth First Search - Implementation
- Breadth First Search - Complexity

Algorithms: Graph Traversal

- Graph Traversal
- Graph Implementation
- Breadth First Search - Implementation
- Depth First Search - Implementation
- Graph Traversal Complexity

Implementations and Interview Questions (IQ)

- Data Structure Implementation
- Problem Solving Approach

IQ: Two Sum

- Two Sum
- Solution: Two Sum

IQ: Min Stack

- Min Stack
- Min Stack Implementation
- Solution: Min Stack

IQ: Max Stack

- Max Stack

IQ: Design a Linked List

- Design a Linked List - I
- Design a Linked List - II
- Design a Linked List - III
- Design a Linked List - IV
- Solution: Design a Linked List

IQ: Reverse Linked List

- Reverse Linked List - I
- Reverse Linked List - II
- Solution: Reverse Linked List

IQ: Construct Binary Tree

- Traversal (Preorder-Inorder-Postorder)
- Construct BT: From Preorder and Inorder Traversal - I
- Construct BT: From Preorder and Inorder Traversal - II
- Solution: Construct BT

IQ: Invert Binary Tree

- Invert Binary Tree - I
- Invert Binary Tree - II
- Solution: Invert Binary Tree

IQ: Construct Binary Search Tree

- Construct BST: From Preorder Traversal
- Construct BST: From Preorder Traversal - II
- Solution: Construct BST

IQ: Detect Capital

- Detect Capital
- Solution: Detect Capital

IQ: Reverse String

- Reverse String
- Solution: Reverse String

IQ: Longest Palindromic Substring

- Longest Palindromic Substring - I
- Longest Palindromic Substring - II
- Solution: Longest Palindromic Substring

Thank You For Being Here!

- Thank You For Being Here!

5 - Course Completion Certificate [Module 10] 🎉

- Data Structures and Algorithms Bootcamp Using Python

Module - 11

Course Introduction
- Course Introduction
- Access Resources

Profile Building
- Section Introduction
- Github Profile README
- Github Project README
- LinkedIn
- Blog
- Twitter
- Freelancing

Resume Building
- Resume Basics
- Resume Content
- Resume: Building First Version
- Resume Examples
- Resume Templates
- Resume Checklist

Searching Internship
- Internship Search - 1
- Internship Search - 2
- Internship Search - 3
- Internship Search - 4

Random Questions
- Section Introduction
- Should You Join Unpaid Internship?
- Building Experience As A Beginner?

Thank You For Being Here!
- Thank You For Being Here!

---------------------- **Note** ----------------------

1. Complete course is divided into 7 different stages according to topics.

2. Future these 7 stages are divided into 11 different modules according to concepts and projects. Here is a quick guide about stages and modules.

Stage 1 - Complete Python Programming - Basics to Advanced Concepts
- Module 1 🏆

Stage 2 - Python Practice In Action - Build Projects From Scratch
- Module 2
- Module 3
- Module 4
- Module 5
- Module 6

Stage 3 - Build and Deploy Web Application with Django Framework
- Module 7 🏆

Stage 4 - Django REST Framework - Build Powerful API Using Python
- Module 8 🏆

Stage 5 - Git & GitHub for Beginners: Mastering the Modern Workflow
- Module 9 🏆

Stage 6 - Data Structures and Algorithms Bootcamp Using Python
- Module 10 🏆

Stage 7 - Internship Search & Resume Building: Preparation For Freshers
- Module 11

Here '*trophy emoji* ' represents a certificate.

3. Throughout the course you get **5 certificates**,

- **First,** after completing [Module 1]
  Certificate 1:
  Complete Python Programming - Basics to Advanced Concepts

- **Second,** after completing [Module 7]
  Certificate 2:
  Django Development Bootcamp: Build & Deploy Web Application With Python

- **Third,** after completing [Module 8]
  Certificate 3:
  Django REST Framework - Build Powerful API Using Python

- **Fourth,** after completing [Module 9]
  Certificate 4:
  Git & GitHub for Beginners: Mastering the Modern Workflow

- **Fifth,** after completing [Module 10]
  Certificate 5:
  Data Structures and Algorithms Bootcamp Using Python

4. For more information about modules, topics and course flow, please check the curriculum part.

5. You will get access to certificates instantly after completing the module lectures as well as quiz included in them.