

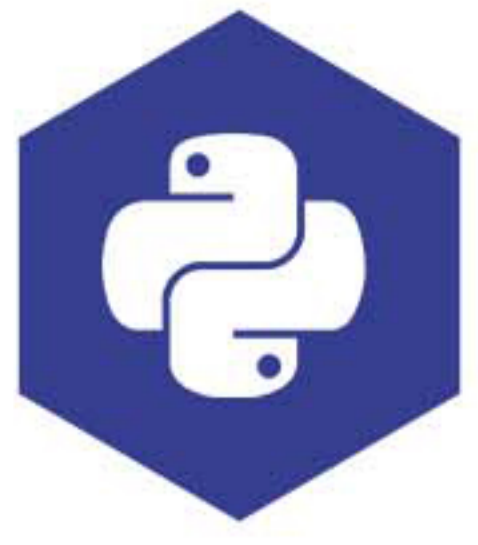


PYTHON Full Stack

(Python Specialization)

Become a Full Stack Developer with expertise on Python,MySQL,Flask, HTML, CSS, JavaScript along with AWS Deployment. Build strong web development fundamentals and problem solving skills.





Python Programming



Python Introduction and Setting Up the Environment

- Introduction to programming
- R or Python?
- Why Python for Data Science?
- Different job roles with Python
- Different Python IDEs
- Downloading and setting up the Python environment

Hands-On - Installing Python - IDLE



Python Basic Syntax and Data Types

- Python input and output operations.
- Comments
- Variables, rules for naming variables
- Basic Data Types in Python
- Typecasting in python

Hands-On - Using comments, variables, data types, and typecasting in python program



Operators in Python

- Arithmetic operators
- Assignment operators
- Comparison operators
- Logical operators
- Identity operators
- Membership Operators
- Bitwise Operators

Hands-On - Working with different types of Operators

Strings

- Creating strings
- String formatting
- Indexing
- Slicing
- String methods

Hands-On - Performing string operations

Tuples

- Syntax to create tuples
- Tuple properties
- Indexing on tuples
- Slicing on tuples
- Tuple methods

Hands-On - Working with Tuples

Lists

- Creating lists
- Properties of lists
- List indexing
- List slicing
- List of lists
- List Methods
- Adding, Updating & removing elements from lists

Hands-On - Slicing, Indexing, and using methods on lists

Sets

- The syntax for creating sets
- Updating sets
- Set operations and methods
- Difference between sets, lists, and tuples

Hands-On - Performing set operations in a program



Dictionaryes

- The syntax for creating Dictionaryes
- Storing data in dictionaryes
- Dictionaryes keys and values
- Accessing the elements of dictionaryes
- Dictionary methods

Hands-On - Creating dictionaryes and using dictionaryes methods



Python Conditional Statements

- Setting logic with conditional statements
- If statements
- If -else statements
- If-elif-else statements

Hands-On - Setting logic in programs using conditional statements



Loops in Python

- Iterating with python loops
- while loop
- for loop
- range
- break
- continue
- pass
- enumerate
- zip
- assert

Hands-On - Iterating with loops in python

Getting Started with **HackerRank** use cases and working on them

- ✓ Solving Level by Level Challenges
- ✓ Assignments to acquire Bronze and Silver Level badges



List and Dictionaries comprehension

- ✓ Why List comprehension
- ✓ The syntax for list comprehension
- ✓ The syntax for dict comprehension

Hands-On - Using List and Dictionary comprehension



Functions

- ✓ What are Functions
- ✓ Modularity and code reusability
- ✓ Creating functions
- ✓ Calling functions
- ✓ Passing Arguments
- ✓ Positional Arguments
- ✓ Keyword Arguments
- ✓ Variable-length arguments (*args)
- ✓ Variable Keyword length arguments (**kargs)
- ✓ Return keyword in python
- ✓ Passing function as an argument
- ✓ Passing function in return
- ✓ Global and local variables
- ✓ Recursion

Hands-On - Creating our own functions, passing arguments, and performing operations



Anonymous Function

- Lambda
- Lambda with filter
- Lambda with map
- Lambda with reduce

Hands-On - Working with lambda, filter, map, and reduce in python



Generators

- Creating and using generators

Hands-On - Creating and using generators



Modules

- Creating modules
- Importing functions from a different module
- Importing Variables from different modules
- Python built-in modules

Hands-On - Creating and importing Modules



Exceptions and Error handling

- Syntax errors
- Logical errors
- Handling errors using try, except and finally

Hands-On - Handling Errors with try and except



Working on Object Oriented Programming



Classes and Objects (OOPS)

- Creating classes & Objects
- Attributes and methods
- Understanding __init__ constructor method
- Class and instance attributes
- Different types of methods
- Instance methods
- Class methods
- Static methods
- Inheritance
- Creating child and parent class
- Overriding parent methods
- The super() function
- Understanding Types of inheritance
- Single inheritance
- Multiple Inheritance
- Multilevel Inheritance
- Polymorphism
- Operator overloading

Hands-On - Creating classes, objects. Creating methods and attributes. Working with different methods. Using inheritance and polymorphism.



Packages

- Creating packages
- Importing modules from the package
- Different ways of importing modules and packages

Introduction to Git and Github

Linkedin Profile creation and Daily Posts

Generate and expire the OTP



Date and Time

- date module
- time module
- datetime module
- time delta
- formatting date and time
- strftime()
- strptime()

Hands-On - working with date and time



Regex

- Understanding the use of regex
- re.search()
- re.compile()
- re.find()
- re.split()
- re.sub()
- Meta characters and their use

Hands-On - Using regular expression to search patterns



Files

- Opening file
- Opening different file types
- Read,write,close files
- Opening files in different modes

Hands-On - Reading, Writing, Appending, opening, and closing files.

Use Cases : Download Instagram public Data

Download Youtube Videos with its details

Sending Automated Emails

Learner will achieve five star badge in HackerRank

Able to Crack any competitive coding question

Learner HackerRank rank will be less than 1 Lakh.



MYSQL DATABASE



Introduction to Database

- What is a Database?
- Why use a Database?
- What is DBMS?
- Types of DBMS
- Introduction to SQL
- Database vs DBMS vs SQL vs MySQL
- MySQL Installation



SQL Sublanguages

- Introduction to SQL Sublanguages
- Creating tables and describing its structure
- DML:INSERT part-1
- DML:INSERT part-2
- Null and Not Null constraints,altering default behaviour
- DQL:Select
- DCL:Commit,Rollback,Savepoint
- DML:Working with Update and Delete commands



MySQL Operators

- Where Clause
- Introduction to MySQL Operators-1
- Introduction to MySQL Operators-2
- MySQL Warnings vs Errors
- Comparison Operators part-1
- Logical AND and other comparison operators

Comparison Operators

- BETWEEN AND and NOT BETWEEN AND,OR operator in Logical Operators
- IN and NOT IN Operator
- IS NULL and IS NOT NULL



DDL:Alter and Rename

- ALTER - ADD
- ALTER - MODIFY
- ALTER - DROP
- ALTER - RENAME and ALTER-RENAME COLUMN
- ALTER - CHANGE
- RENAME



String Functions

- Use of Alias
- CONCAT
- CONCAT WS
- LOWER and UPPER
- CHARACTER LENGTH and SUBSTRING
- REVERSE and REPLACE
- Other String functions



Constraints

- Introduction to Constraints
- NOT NULL and DEFAULT Constraints
- Primary key Constraint
- Unique Constraint
- Working with Auto_increment
- CHECK Constraints
- Relationships in DBMS(One to One,One to many etc.)
- Introduction to Foreign key constraint
- Working with foreign keys
- REFERENTIAL ACTIONS and Alter Constraints
- Alter -add with other constraints



Refining Selections and working with MySQL workbench

- DISTINCT
- ORDER BY-1
- ORder BY -2
- LIMIT
- LIKE
- MySQL workbench Installation
- Working with MySQLworkbench



Working with Aggregate functions and SQL FILES

- Introduction to aggregate functions
- Count
- Group by
- Min and Max
- Subqueries
- Group by with Min and Max
- SUM And AVG
- Group by with SUM and AVG
- Exporting MySQL Databases
- Importing MySQL Databases



More on Data types

- Char Vs Varchar
- Integer data types(SMALLINT,TINYINT,..)
- FLOAT VS DECIMAL
- Working with Date and TIME
- CURDATE,CURTIME AND CURRENT_TIMESTAMP
- Date functions
- Time Functions
- Formatting dates
- Operations on dates
- Datetime vs Timestamp
- Automatic Initialization and Updating for TIMESTAMP and DATETIME



MySQL JOINS

- Why JOINS?
- Introduction to JOINS
- CROSS JOIN
- INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN



Project:Instagram Database Clone

- Cloning Instagram's DB: Users Schema
- Cloning Instagram's DB:Followers Schema
- Cloning Instagram's DB:Photos Schema
- Cloning Instagram's DB:LikesSchema
- Cloning Instagram's DB:Hashtags Schema
- Instagram Clone Challenges



Python for Web Development - Flask



Flask Introduction and installation

- ✓ Introduction to flask and its architecture
- ✓ Installing flask package and introduction to its components
- ✓ Introduction to Virtual Environment and its importance in project development
- ✓ Creating Virtual Environment and activating,deactivating it
- ✓ Introduction to routing in Flask and Building sample flask application



Building routes with flask

- ✓ What is a dynamic route?
- ✓ Building dynamic routes with flask
- ✓ Redirection in Flask
- ✓ Dynamic url building with url_for function
- ✓ URL converters in Flask
- ✓ int and string url converters
- ✓ Float,uuid and path converters
- ✓ request and response in Flask

**Hands on:Flask package Installation and creating virtual environment,
Building dynamic Routes with flask**



Getting into Front-end WebDevelopment - HTML



Introduction

- ✓ History and Importance of HTML
- ✓ Basic HTML syntax
- ✓ The current state of HTML
- ✓ HTML resources
- ✓ Choosing a code editor



Basic Page Structure

- Exploring an HTML document
- DOCTYPE declarations
- Document head
- Document body
- Understanding content models



Formatting Page Content

- Formatting content with HTML
- Headings
- Paragraphs
- Emphasising text
- Displaying special characters
- Whitespaces
- Images



Creating Lists

- Unordered lists
- Ordered lists
- Definition lists



Structuring Content

- Value of structure
- Document outlines
- Nav element
- Article element
- Section element
- Aside element
- Div element
- Other semantic elements



Creating Links

- Anchor element
- Page internal linking
- Linking to external pages
- Linking to downloadable
- Linking to page regions



Controlling styling

- HTML and CSS
- Creating inline styles
- Style element
- Controlling typography
- Adding colour
- Externalising styles



Basic Scripting

- HTML and JavaScript
- The script element
- Writing a function
- Using the DOM
- Listening for an event
- Responding to events
- Externalizing JavaScript



CSS3:



Getting started

- ✓ HTML overview
- ✓ Default browser styles
- ✓ Browser support and inconsistencies
- ✓ Inline, internal and external CSS
- ✓ Naming conventions



CSS Core

- ✓ Syntax, terminology, and naming conventions
- ✓ Type, class, and id selectors
- ✓ Combinator selectors
- ✓ Selectors: Best practices



Flask Request Handling

- ✓ Introduction to flask request object
- ✓ Requests handling in Flask
- ✓ Introduction to Flask files structure
- ✓ Implementation of Flask files structure



Jinja 2 Template Engine

- ✓ Flask Intro to Jinja2
 - Rendering HTML pages with render_template function
- ✓ Integrating css with flask
 - More about url_for and its usage in Jinja 2

**Hands on: Requesting handling with flask and integrating frontend with flask
i.e HTML and CSS with Jinja2**



Dynamic Web Pages with flask-Jinja2

- ✓ More on render_template and its syntax
- ✓ Jinja2 outputting values to template
- ✓ Introduction to Jinja Inheritance
- ✓ Defining Inheritance blocks with jinja2
- ✓ Inheriting parent contents to child template with jinja 'extends'
- ✓ Introduction to Jinja 2 statements
- ✓ If statement in Jinja2
- ✓ Nested if,if else statements in jinja2
- ✓ For loop in Jinja2

Hands on: Rendering Dynamic web pages and performing conditional actions with jinja2



CSS



Typography

- ✓ Web-safe fonts and the font-family property
- ✓ Web fonts and Google fonts
- ✓ The font-size property
- ✓ Practicing with web fonts and font-size
- ✓ The font-style and font-weight properties
- ✓ The color, line height, and text properties
- ✓ Adjusting the font-weight property



Layouts

- Block vs. inline display
- The box model
- Margin and page layouts
- Practising layouts
- Practising with padding and spacing
- Floats
- Practising with floats
- The box model fix
- Practising with columns
- Box model review
- Float and display review
- Horizontal navs with the display property
- Horizontal navs with the float property
- Practising with the nav element
- Positioning
- Practising with fixed navigation
- Practising with positioning elements
- Float, display, and position
- Layers and the z-index property



Login system with flask, Server side sessions

- Introduction to Server side sessions
- Server side sessions with flask_session
- More on flask_session and session lifecycle control
- Creating and popping user sessions
- mysql database integration with flask using mysql_connector
- More on mysql_connector package



CSS

- Horizontal navs with the display property
- Horizontal navs with the float property
- Practicing with the nav element
- Positioning
- Practicing with fixed navigation
- Practicing with positioning elements
- Float, display, and position
- Layers and the z-index property



Files Handling with flask

- Introduction to binary data
- Storing user files in MySQL database(i.e pdfs,excel,docx, etc)
- Viewing and Downloading files with flask's send_file on browser

Hands on: Storing binary data into MySQL database i.e files data and conversion of binary data to bytes using BytesIO, sending files to browser using flask's send_file on browser



CSS



Advanced layout

- Design with a grid
- Intro to CSS Grid
- CSS Grid example
- Intro to flexbox
- CSS Flexbox
- Responsive page building



Projects

Custom DevBlog Application

- Users sign up and login and log out, forgot password
- Blogs-create, update, delete

Creating an internal template using HTML & CSS

- Customized Template with addition of Media Queries, Layouts.

Instagram Login page

- Creating a similar Instagram Clone Login page



JavaScript



Introduction

- What is JavaScript?
- How does JavaScript relate to Java?
- History of JavaScript
- Navigating the JavaScript landscape



Basics

- Tools for JavaScript development
- Introducing the browser console
- Add inline JavaScript to an HTML document
- Add JavaScript in an external file



Writing JavaScript

- Basic syntax & comments
- Declarations
- Arithmetic operators and math
- Working with strings and numbers
- Variable scope
- Variable hoisting
- Data structures and types
- Literals



Custom DevBlog Application

- Conditional statements and logic
- Advanced conditions and logic
- if..else
- Switch



JavaScript



Control flow

- Conditional statements and logic
- Advanced conditions and logic
- if..else
- Switch



Arrays

- Arrays
- Typed arrays



Loops and iteration

- for
- while
- do..while
- break/continue
- for..in
- for..of



JavaScript



Functions

- Defining functions
- Calling functions
- Function scope
- Closures
- Arguments & parameters
- Arrow functions



Flask



Fully Functional Ecommerce Application



Admin Dashboard

- sign up and login and log out, forgot password

Users sign up, login and logout, forgot password



Essential JavaScript Built-in methods

- Number Methods
- String Methods
- Array Methods
- Math Methods



Writing JavaScript-Advanced

- Text formatting - String literals
- String object
- Template literals
- Regular expressions



Fully Functional Ecommerce Application



Admin Dashboard

Add and update,delete items Items dashboard



Fully Functional Ecommerce Application



Admin Dashboard

Items Dashboard

User dashboard



JavaScript and the DOM

- Intro to DOM
- Target elements in the DOM with querySelector methods
- Access and change elements
- Access and change classes
- Access and change attributes
- Add DOM elements
- Apply inline CSS to an element
- DOM events



Fully Functional Ecommerce Application

- Search bar
- Shopping cart



JavaScript



Es6 Concepts

- map
- filter
- const and let
- promises



Fully Futnctional Ecommerce Application



Payment Gateway

- Stripe Payment Gateway Integration with flask

Creating Orders section-1



JAVASCRIPT

- Asynchronous actions
- API calls
- API integration using fetch and axios



Fully Functional Ecommerce Application

- Adding more Orders section



Deployment in Cloud

- ✓ AWS Elastic Beanstalk
- ✓ Creating Environment,sample flask application hosting
- ✓ AWS Codepipeline and mysql rds configuration with flask
- ✓ Project deployment



Calculator using JS

- ✓ We create a real time calculator to enter values, perform multiple operations



Random Password Generator

- ✓ OTP Generator combination of alphabets,numbers and symbols

Capstone Projects

Users will be able to deploy every project in
Amazon Web Services



Fully Functional Ecommerce Application

- ✓ Admin dashboard-
 - Sign up,Sign in
 - Forget password
 - Add and update,delete items
 - Items dashboard
- ✓ User dashboard
 - Sign up,Sign in
 - Forget Password
 - Search bar
 - Shopping cart
- ✓ Payment Gateway
 - Stripe Payment Gateway Integration with flask
- ✓ Generating Invoice for an Order
- ✓ Orders Section



Custom DevBlog Application

- Users sign up
- Sign in and Sign out
- forgot password
- Blogs-create,update,delete
- Search bar
- Blogs sharing on social media



Discussion Board Application

- User Sign up,Sign in and sign out
- Forget Password
- Discussion - Create,Update and Delete
- Adding Comments
- Adding Replies
- Share Discussions



Personal Records Manager

- User Sign up,Sign in and sign out
- Forget Password
- Notes-Add,View,Update and delete
- Download notes in Excel
- Files-upload,delete,view and download



Outcomes

Learner will able to build end to end project including deployment in cloud

Learner can apply to 7 different job roles

- Python Developer
- Backend Developer
- Database Developer
- Frontend Developer
- UI Developer
- Javascript Developer
- Full Stack Developer