Python Coding

(40h / 5 Days)

Description:

This comprehensive Python Coding course is crafted to provide trainees with a solid foundation in Python programming, while emphasising practical, real-world applications. Beginning with the basics of Python syntax and programming logic, the course introduces essential concepts such as control structures, data types, and functions. Participants will engage in hands-on labs, developing scripts that automate simple tasks, and gradually advancing to more complex data structures and object-oriented programming. By building upon each lesson, the course ensures that participants gain confidence in writing clean, efficient Python code.

As the course progresses, trainees will dive into advanced topics, such as file handling, error management, and using popular Python libraries for web development, data analysis, and machine learning. Participants will learn to create

web applications using Django or Flask, implement automation scripts to streamline workflows, and harness the power of data analytics through Pandas and NumPy. The capstone project allows participants to integrate these skills, culminating in the development of a comprehensive Python application that solves a real-world problem.

Target Audience:

1. **Beginners in Programming**: Individuals new to programming who want to start with a versatile and easy-to-learn language.
2. **Software Developers**: Developers in other languages looking to add Python to their skillset for web development or data science.
3. **IT Professionals**: System administrators, network engineers, and other IT staff who can use Python to automate tasks.
4. **Students and Educators**: College students and teachers who require Python for academic purposes or personal development.
5. **Tech Enthusiasts**: Anyone interested in technology and coding who wants to learn a new skill or start a project.

Training Expected Outcomes:

By the end of this training, participants will be able to:

1. **Confidently Code in Python**: Write efficient, maintainable Python code for various applications.
2. **Build Web and Data Applications**: Use frameworks like Django or Flask for web development and libraries like Pandas and NumPy for data analysis.
3. **Automate Tasks and Processes**: Create scripts to automate routine tasks, enhancing productivity.
4. **Deploy Python Applications**: Understand deployment processes for Python applications.
5. **Contribute to Projects**: Be capable of joining and contributing to Python projects in professional or open-source environments.

1.

Training Strategy

1. **Module-Based Learning**: Structured into 10 modules, each focusing on different aspects of Python programming, from basic syntax to advanced modules.
2. **Theoretical Lessons**: Provide a foundational understanding of programming principles in Python, data structures, object-oriented programming, and more.
3. **Project-Based Labs**: Each module includes a hands-on lab where participants develop parts of a larger project, applying the concepts learned in real-time.
4. **Assessments**: End-of-module quizzes and a final presentation of the project assess understanding and practical skills.
5. **Real-World Application**: Emphasises applying Python to solve practical problems in fields like web development, data science, and automation.

Course Modules

1. **Introduction to IoT and AWS IoT** – Understand IoT concepts and AWS IoT Core services for device connectivity.
2. **IoT Device Management** – Manage large IoT device fleets using AWS IoT Device Management.
3. **IoT Communications and Protocols** – Master MQTT, HTTP, and secure communication protocols in IoT systems.
4. **Data Handling and Processing in IoT** – Collect, process, and analyze IoT data using AWS IoT Analytics.
5. **IoT and Machine Learning** – Integrate machine learning models with IoT data to enhance decision-making.
6. **IoT Security** – Implement best practices for securing IoT devices and communication.
7. **Real-Time Data Streaming and Analysis** – Process real-time IoT data streams using AWS Kinesis.
8. **IoT Edge Computing with AWS Greengrass** – Enable local data processing on IoT devices using AWS Greengrass.
9. **IoT Integration with Other AWS Services** – Enhance IoT systems by integrating with AWS Lambda, S3, and DynamoDB.
10. **Capstone Project and IoT Solution Deployment** – Design and deploy a complete IoT solution using AWS services.

Training Program

Python Coding	
Training Objectives: • Master Python Fundamentals • Develop Web and Data Applications • Implement Machine Learning and Automation • Ensure Application Security and Management • Integrate and Apply Python Skills in Projects	
Time	***Modules***

4 Hours
 	
1.5h
 	
2.5h
	Module 1: Introduction to Python
	Objective: Familiarise trainees with Python syntax, basic programming concepts, and the Python programming environment.
	• **Python Syntax and Basics**
	  • Introduction to Python, setup, and basic syntax.
	  • Variables, data types, and operators.
	Lab: Writing simple Python scripts that perform basic calculations.
	Assessment: Quiz on Python fundamentals.

4 Hours
 	
2h
 	
2h
	Module 2: Control Structures and Functions
	
	Objective: Develop an understanding of Python control flow mechanisms, functions, and modular programming techniques.
	
	--- 	
	
	• **Control Flow**
	  ◦ Conditionals, loops, and control statements.
	• **Functions and Modules**
	  ◦ Defining functions, scope, and importing modules.
	
	Lab: Develop a Python script that uses functions and modules to organise code logically.
	
	Assessment: Hands-on project segment involving conditional logic and function definitions.

4 Hours	**Module 3: Data Structures** **Objective**: Master the use of Python's built-in data structures like lists, dictionaries, sets, and tuples for complex data manipulation. --- • **Lists, Tuples, and Dictionaries** ◦ Effective use of Python collections: lists, tuples, dictionaries, and sets. **Lab**: Building and manipulating complex data structures to store and process data. **Assessment**: Practical exercises on data structures.

4 Hours	**Module 4: Object-Oriented Programming** **Objective**: Introduce object-oriented programming concepts in Python, focusing on classes, inheritance, and polymorphism. --- • **Classes and Objects** ◦ Introduction to OOP in Python: classes, objects, inheritance, and polymorphism. **Lab**: Designing a simple class hierarchy that models a real-world scenario. **Assessment**: Developing a small application using object-oriented principles.

4 Hours	**Module 5: Exception Handling and File Operations** **Objective**: Enable robust Python programming practices using exception handling and file management for data persistence. --- 2h • **Error and Exception Handling** ◦ Managing exceptions and errors to build reliable applications. 2h • **File Handling** ◦ Reading from and writing to files. **Lab**: Implementing a file-based data logging system. **Assessment**: Error handling and file operations project.
4 Hours	**Module 6: Libraries and Frameworks for Web Development** **Objective**: Enable trainees to build web applications using Python frameworks such as Flask or Django. --- • **Using Flask/Django** ◦ Building a basic web application using Flask or Django. **Lab**: Creating a web application that captures and displays user input. **Assessment**: Quiz on web development with Python.

4 Hours
 	
2h
 	
2h
	Module 7: Security Practices in Python Development
	
	Objective: Instruct on best practices for securing Python applications, focusing on preventing common vulnerabilities and integrating AWS security tools.
	
	--- 	
	
	• **Security Fundamentals in Python**
	  ◦ Understanding common security vulnerabilities in Python applications, such as injection attacks, cross-site scripting (XSS), and cross-site request forgery (CSRF).
	  ◦ Best practices for secure coding to prevent common vulnerabilities.
	• **Using Python Packages Securely**
	  ◦ Managing dependencies securely with tools like Pip and virtual environments.
	  ◦ Checking for vulnerabilities in third-party packages and keeping dependencies up to date.
	• **Lab**: Implementing security measures in a Python web application.
	  ◦ Participants will enhance an existing Python web application by integrating security features, such as input validation, secure authentication, and the use of HTTPS.
	• **Assessment**: Security review of a Python project.
	  ◦ Participants will be given a piece of Python code to audit for security vulnerabilities, suggesting improvements and implementing fixes.

4 Hours	**Module 8: Data Analysis with Python** **Objective**: Provide trainees with skills to perform data analysis using Python libraries like Pandas and NumPy. --- • **Introduction to Pandas and NumPy** ◦ Using Pandas for data manipulation; NumPy for numerical data. **Lab**: Analysing a dataset using Pandas and visualising data with Matplotlib. **Assessment**: Data manipulation and analysis project.

4 Hours	**Module 9: Cloud's Python Applications management**
	Objective: Enable trainees to deploy and manage Python applications in the AWS cloud, using services like AWS Elastic Beanstalk, AWS Lambda, and CloudWatch for scaling, monitoring, and infrastructure management.
	--- 	
	2h
	- **Deploying Python Applications to AWS**
	  - Overview of deployment options for Python applications on AWS, including Elastic Beanstalk, AWS Lambda, and EC2.
	  - Deploying a Python application using Elastic Beanstalk and configuring the environment for optimal performance.
	2h
	- **AWS Cloud Management Best Practices**
	  - Techniques for managing and monitoring Python applications in AWS.
	  - Using AWS CloudWatch for monitoring, AWS CloudFormation for infrastructure as code, and AWS Auto Scaling to handle load changes.
	Lab: Deploying and managing a Python application using AWS Elastic Beanstalk. Participants will also set up monitoring using AWS CloudWatch and implement scaling policies.
	Assessment: A project where participants must deploy a Python application, set up a complete monitoring and alerting framework, and demonstrate scalability using AWS Auto Scaling.

4 Hours	**Module 10: Introduction to Machine Learning with Python** **Objective**: Provide a foundational understanding of how to implement basic machine learning algorithms using Python and scikit-learn. --- • **Machine Learning Basics** ◦ Overview of machine learning concepts and algorithms. **Lab**: Building a simple machine learning model using scikit-learn. **Assessment**: Developing a machine learning model to solve a classification problem.
4 Hours 1h 1h	**Module 11: Automation with Python (4 hours)** **Objective**: instruct trainees how to automate repetitive tasks and processes using Python scripts to improve efficiency. --- • **Scripting for Automation** ◦ Writing scripts to automate everyday tasks and processes. **Lab**: Automating file organization and data entry tasks. **Assessment**: Automation project presentation.

4 Hours	**Module 12: Capstone Project and Review**
	Objective: Integrate all skills learned into a comprehensive project, demonstrating proficiency in Python programming across various applications.
	--- 	
	• **Capstone Project**
	  ◦ Integrating knowledge from previous modules to create a comprehensive Python project.
	• **Review and Feedback**
	  ◦ Participants present their projects; receive feedback from instructors and peers.
	
	Assessment: Final project presentation and review.