

Developing Microservices with Python

Course Summary

Description

Python Microservices begins by introducing you to serverless microservice structures. You will then learn how to create your first serverless data API and test your microservice. Moving on, you'll delve into data management and work with serverless patterns. Finally, the course introduces you to the importance of securing microservices.

Objectives

By the end of the course, you will have gained the skills you need to combine microservices with serverless computing, making their deployment much easier thanks to the cloud provider managing the servers and capacity planning.

Topics

- Serverless Microservices Architectures and Patterns
- Creating Your First Serverless Data API
- Deploying Your Serverless Stack
- Testing Your Serverless Microservice
- Securing Your Microservice

Prerequisites

Basic computer skills, internet access, basic analytic or programming skills.

Duration

Three Days

Developing Microservices with Python

Course Outline

- I. *Serverless Microservices Architectures and Patterns***
 - A. Understanding different architecture types and patterns
 - B. Virtual machines, containers, and serverless computing
 - C. Overview of microservice integration patterns
 - D. Communication styles and decomposition microservice patterns
 - E. Serverless computing in AWS
 - F. Setting up your serverless environment
- II. *Creating Your First Serverless Data API***
 - A. Overview of security in AWS
 - B. Securing your serverless microservices
 - C. Building a serverless microservice data API
 - D. Setting up Lambda security in the AWS Management Console
 - E. Creating and writing to a NoSQL database called DynamoDB using AWS
 - F. Creating and writing to a NoSQL database called DynamoDB using Python
 - G. Creating a Lambda to query DynamoDB
 - H. Setting up the API Gateway and integrating it with a Lambda proxy
 - I. Connecting API Gateway, Lambda, and DynamoDB
 - J. Cleaning-up
- III. *Deploying Your Serverless Stack***
 - A. An overview of serverless stack build and deploy options
 - B. Creating a profile, an S3 bucket, IAM policies, and IAM roles resources
 - C. Building and deploying with API Gateway, Lambda, and DynamoDB
- IV. *Testing Your Serverless Microservice***
 - A. Unit testing your Python Lambda code
 - B. Running and debugging your AWS Lambda code locally
 - C. Integration testing using real test data
 - D. The AWS Serverless Application Model CLI
 - E. Loading and end-to-end testing at scale
 - F. Strategies to reduce the API's latency
 - G. Cleaning up
- V. *Securing Your Microservice***
 - A. Overview of the security in AWS
 - B. Overview of AWS Identity and Access Management (IAM)
 - C. Securing your serverless microservice