AW230 Developing on AWS

Kurzbeschreibung:

Dieser Kurs **AW230 Developing on AWS** vermittelt erfahrenen Entwicklern, wie sie programmgesteuert mit AWS-Services interagieren, um Weblösungen zu erstellen. Er führt Sie durch eine Architekturübersicht zur Ressourcenauswahl und vertieft anschließend die Nutzung der AWS Software Development Kits (AWS SDKs) und der Command Line Interface (AWS CLI) zur Entwicklung und Bereitstellung Ihrer Cloud-Anwendungen.

Sie entwickeln im Kurs eine Beispielanwendung und lernen dabei, wie Sie Berechtigungen für die Entwicklungsumgebung einrichten, Geschäftslogik zur Datenverarbeitung mit AWS-Kernservices hinzufügen, Benutzeranmeldung konfigurieren, in der AWS Cloud bereitstellen und Fehler beheben. Der Kurs **AW230 Developing on AWS** enthält Codebeispiele zur Umsetzung der behandelten Entwurfsmuster und Lösungen.

Die praktischen Übungen vertiefen zentrale Kursinhalte und helfen Ihnen, Lösungen mit dem AWS SDK für Python, .NET und Java, der AWS CLI und der AWS Management Console zu implementieren.

Dieser Kurs **AW230 Developing on AWS** umfasst Präsentationen, Demonstrationen und praktische Übungen (Hands-on Labs).

Zielgruppe:

Dieser Kurs **AW230 Developing on AWS** richtet sich an erfahrene:

- Softwareentwickler
- Solution Architects
- IT-Fachkräfte, die ihre Entwicklungskompetenz mit AWS-Services verbessern möchten

Voraussetzungen:

Um an dem Kurs **AW230 Developing on AWS** bei qSkills teilnehmen zu können, sollten Sie folgendes Training besucht haben:

- AW110 AWS Technical Essentials

Darüber hinaus sollten Sie folgende Voraussetzungen erfüllen:

- Über praktische Kenntnisse der AWS-Kernservices verfügen
- Programmiererfahrung in mindestens einer der folgenden Programmiersprachen mitbringen:
  - Python
  - .NET
  - Java

Sonstiges:

Dauer: 3 Tage

Preis: 1995 Euro plus Mwst.

Ziele:

In diesem Kurs **AW230 Developing on AWS** lernen Sie:

- Eine einfache End-to-End-Cloud-Anwendung mit AWS SDKs, AWS CLI und IDEs zu entwickeln
- AWS Identity and Access Management (IAM)-Berechtigungen zur Unterstützung einer Entwicklungsumgebung zu konfigurieren
- Mehrere Programmiermuster in Ihren Anwendungen zur Nutzung von AWS-Services anzuwenden
- Mit AWS SDKs CRUD-Operationen (Create, Read, Update, Delete) auf Amazon S3- und Amazon DynamoDB-Ressourcen durchzuführen
- AWS Lambda-Funktionen mit Service-Integrationen für Webanwendungen zu entwickeln
- Die Vorteile von Microservices-Architekturen und serverlosen Anwendungen für das Design zu verstehen
- API-Gateway-Komponenten zu entwickeln und mit anderen AWS-Services zu integrieren
- Zu erklären, wie Amazon Cognito den Benutzerzugriff auf AWS-Ressourcen steuert
- Eine Webanwendung mit Cognito zu erstellen, um Benutzerzugriffe bereitzustellen und zu steuern
- DevOps-Methodik anzuwenden, um Risiken traditioneller Softwarebereitstellungen zu reduzieren, und AWS-Services zur Umsetzung von DevOps-Praktiken zu identifizieren
- AWS Serverless Application Model (AWS SAM) zur Bereitstellung einer Anwendung zu nutzen
- Den Build Ihrer Anwendung mit Amazon X-Ray zu beobachten

Inhalte/Agenda:

- ♦ **Kursüberblick**
  - ♦ ◊ Organisatorisches
    ◊ Schulungsunterlagen
    ◊ Agenda
    ◊ Vorstellung der Teilnehmenden
- ♦ ◊
- ♦ **Entwicklung einer Webanwendung auf AWS**
  - ♦ ◊ Architektur der im Kurs zu entwickelnden Anwendung besprechen
    ◊ Notwendige AWS-Services zur Erstellung der Webanwendung identifizieren
    ◊ Speichern, Verwalten und Hosten der Webanwendung
- ♦ ◊
- ♦ **Einstieg in die Entwicklung auf AWS**
  - ♦ ◊ Zugriff auf AWS-Services über Programmierung beschreiben
    ◊ Programmierschemata und deren Vorteile in AWS SDKs und AWS CLI erläutern
    ◊ Nutzen von AWS Cloud9 erklären
- ♦ ◊
- ♦ **Einstieg in Berechtigungen**
  - ♦ ◊ Funktionen und Komponenten von IAM zur Unterstützung der Entwicklung erläutern
    ◊ Testen von IAM-Berechtigungen demonstrieren
    ◊ IDEs und SDKs für die Entwicklungsumgebung konfigurieren
    ◊ Zugriff auf AWS-Services über SDKs und Cloud9 demonstrieren
- ♦ ◊
- ♦ **Lab 1: Entwicklerumgebung konfigurieren**
  - ♦ ◊ Verbindung zur Entwicklerumgebung herstellen
    ◊ Überprüfung, ob IDE und AWS CLI installiert und konfiguriert sind
    ◊ Berechtigungen zum Ausführen von CLI-Befehlen prüfen
    ◊ IAM-Richtlinie zuweisen, um ein S3-Bucket zu löschen
- ♦ ◊
- ♦ **Einstieg in Storage**
  - ♦ ◊ Grundkonzepte von Amazon S3
    ◊ Optionen zur Datensicherung mit S3
    ◊ SDK-Abhängigkeiten definieren
    ◊ Verbindung zu Amazon S3 beschreiben
    ◊ Anfrage- und Antwortobjekte erklären
- ♦ ◊
- ♦ **Verarbeitung von Storage-Operationen**
  - ♦ ◊ Bucket- und Objektoperationen durchführen
    ◊ Umgang mit großen und mehreren Objekten erläutern
    ◊ S3-Bucket für statische Website konfigurieren
    ◊ Temporärer Objektzugriff gewähren
    ◊ S3-Operationen mit SDK demonstrieren
- ♦ ◊
- ♦ **Lab 2: Lösungen mit Amazon S3 entwickeln**
  - ♦ ◊ Programmatischer Zugriff auf S3 mit SDKs und CLI
    ◊ Bucket mit Waiters erstellen und Servicefehler prüfen
    ◊ Objekt mit Metadaten hochladen
    ◊ Objekt herunterladen, verarbeiten und erneut hochladen
    ◊ Bucket für Website-Hosting konfigurieren und mit CLI synchronisieren
    ◊ Bucket-Policies für S3-Zugriff hinzufügen
- ♦ ◊
- ♦ **Einstieg in Datenbanken**
  - ♦ ◊ Hauptkomponenten von DynamoDB beschreiben
    ◊ Verbindung zu DynamoDB herstellen
    ◊ Anforderungsobjekte erstellen
    ◊ Antwortobjekte lesen
    ◊ Häufige Fehlermeldungen beschreiben
- ♦ ◊
- ♦ **Verarbeitung von Datenbankoperationen**
  - ♦ ◊ Programme zur Interaktion mit DynamoDB entwickeln
    ◊ CRUD-Operationen auf Tabellen und Indizes durchführen
    ◊ Best Practices beim Zugriff auf DynamoDB
    ◊ Caching-Optionen zur Performance-Optimierung
    ◊

- ◊ DynamoDB-Operationen mit SDK durchführen
- ◊
- **Lab 3: Lösungen mit Amazon DynamoDB entwickeln**
  - ◊ Zugriff auf DynamoDB mit Low-Level-, Document- und High-Level-APIs
  - ◊ Daten mit Filtern, Ausdrücken und Paginierung abrufen
  - ◊ Tabelle mit JSON-Daten befüllen
  - ◊ Daten aktualisieren und Attribute hinzufügen
  - ◊ Zugriff auf DynamoDB mit PartiQL und dem Object Persistence Model
- ◊
- **Verarbeitung von Anwendungslogik**
  - ◊ Lambda-Funktion mit SDK entwickeln
  - ◊ Trigger und Berechtigungen konfigurieren
  - ◊ Testen, Bereitstellen und Überwachen von Lambda-Funktionen
- ◊
- **Lab 4: Lösungen mit AWS Lambda entwickeln**
  - ◊ Lambda-Funktion erstellen und programmatisch nutzen
  - ◊ Umgebungsvariablen und Service-Integrationen einrichten
  - ◊ Pre-Signed URLs mit SDK generieren
  - ◊ Lambda mit .zip über IDE bereitstellen und testen
  - ◊ Ausführung über Console und CLI
- ◊
- **API-Verwaltung**
  - ◊ Hauptkomponenten von API Gateway
  - ◊ Ressourcen entwickeln und integrieren
  - ◊ API-Anfragen und -Antworten konfigurieren
  - ◊ API-Ressourcen testen und bereitstellen
- ◊
- **Lab 5: Lösungen mit Amazon API Gateway entwickeln**
  - ◊ RESTful APIs erstellen und CORS konfigurieren
  - ◊ API-Methoden mit Lambda integrieren
  - ◊ Mapping Templates konfigurieren
  - ◊ Request Models definieren
  - ◊ API zu Stage deployen und testen
- ◊
- **Entwicklung moderner Anwendungen**
  - ◊ Herausforderungen traditioneller Architekturen
  - ◊ Microservice-Architekturen und Vorteile
  - ◊ Monolithen entkoppeln
  - ◊ Orchestrierung mit AWS Step Functions
- ◊
- **Benutzerzugriff verwalten**
  - ◊ Sicherheitsprotokolle und Entwicklung
  - ◊ Authentifizierung mit Amazon Cognito
  - ◊ Zugriffskontrolle und serverlose APIs
  - ◊ Best Practices für Cognito
  - ◊ JWT-Token verstehen und prüfen
- ◊
- **Abschlussprojekt – Anwendung fertigstellen**
  - ◊ User Pool und App Client erstellen
  - ◊ Benutzer hinzufügen und Login testen
  - ◊ Cognito als Authorizer für API Gateway konfigurieren
  - ◊ JWT-Token validieren
  - ◊ API-Ressourcen mit Swagger importieren
  - ◊ Frontend mit Cognito und API Gateway verbinden und testen
- ◊
- **Anwendungsbereitstellung**
  - ◊ Risiken traditioneller Bereitstellungen
  - ◊ DevOps-Methodik verstehen
  - ◊ AWS SAM Template konfigurieren
  - ◊ Bereitstellungsstrategien erläutern
  - ◊ Serverlose Anwendung mit SAM deployen
- ◊
- **Überwachung der Anwendung**
  - ◊ Unterschied zwischen Monitoring und Observability
  - ◊ Rolle von Amazon CloudWatch
  - ◊ Nutzung von CloudWatch Application Insights
  - ◊ Debugging mit X-Ray
- ◊
- **Anwendung mit X-Ray überwachen**
  - ◊ X-Ray-Integration im Code
  - ◊ Logs im Deployment-Paket aktivieren
  - ◊ SAM Template verstehen und anwenden
  - ◊ Service Maps und Traces analysieren
  - ◊

-
- **Kursabschluss**
  - Kursrückblick
  - Weitere AWS-Trainings
  - Zertifizierungen
  - Feedback
  -