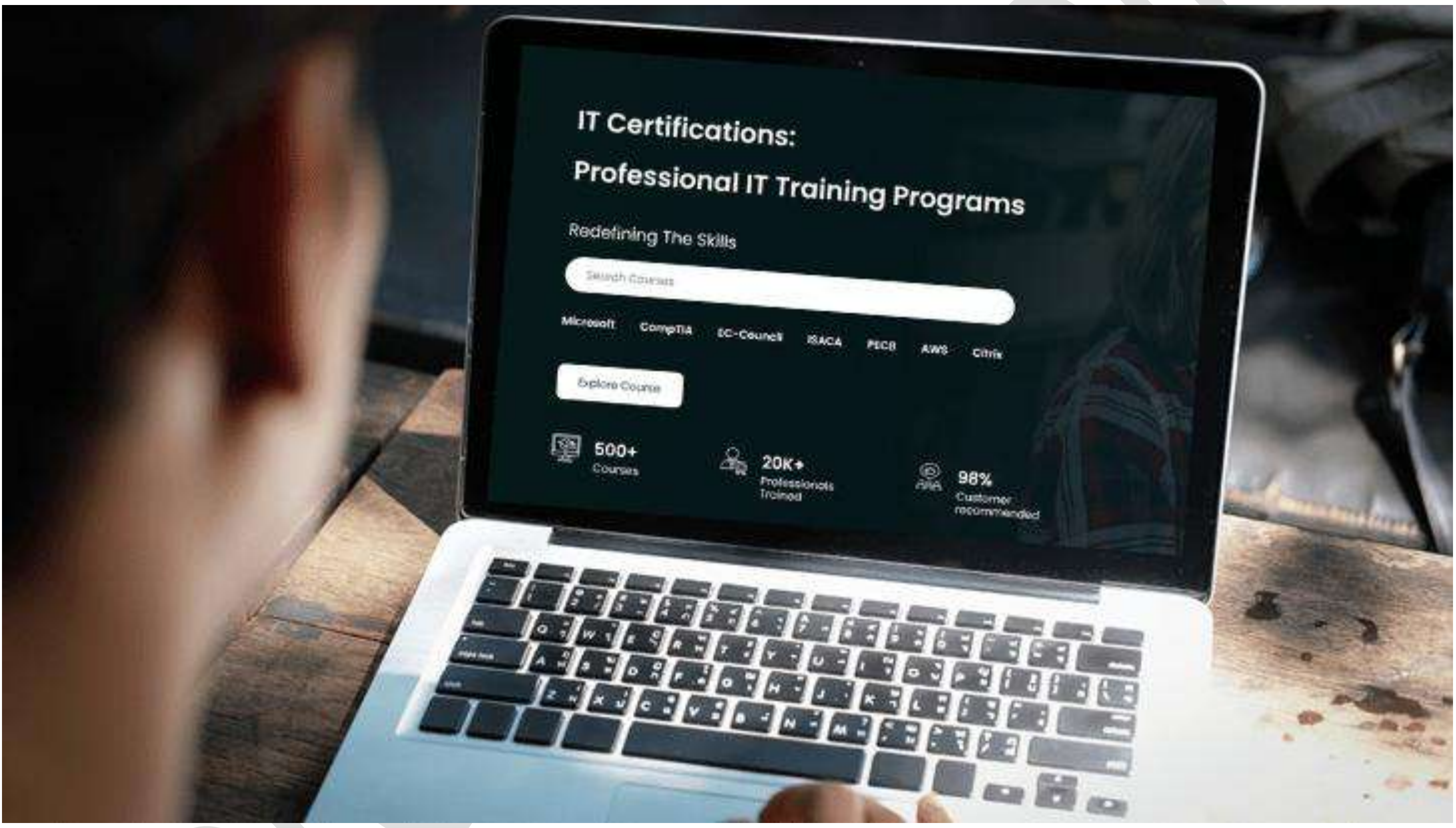
MICROTEK LEARNING | Redefining The Skills



AZ-204T00: DEVELOPING SOLUTIONS FOR MICROSOFT AZURE TRAINING

Duration: 5 Days

Course Description

AZ-204T00: Developing Solutions for Microsoft Azure (Developer) Training is the perfect certification for developers. This course teaches the core functions of how to build end-to-end solutions in Azure. It also helps in gaining knowledge on creating Azure functions, implementing Azure compute solutions, and developing solutions with Azure storage.

AZ-204 training course also specializes in key aspects of cloud development which include topics like user authentication, developing event-based and message-based solutions, and optimizing Azure services.

Microtek Learning's training program is best for organizations and companies as it guides professionals to connect and consume Azure, as well as third-party services. This course is designed with details for covering all the important elements from secure cloud solutions to API management. Doing so makes this training very resourceful for individuals preparing for the Microsoft Certified Azure Developer Associate Certification Exam.

This training is designed based on the objectives of the course variant AZ-204T00-A.

Who should attend this course?

- Any professional who has basic experience in cloud concepts and Azure services.
- Professionals who are aiming to gain a deep understanding of developing solutions.
- Given below are professionals who can use Developing Solutions for Microsoft Azure to upskill their current positions:
    - Application Engineers
    - Technical Team Leads
    - DevOps Engineers
    - Cloud Solution Architects
    - Software Developers

What you will learn

- Creating Azure App Service Web Apps
- Implementing Azure functions
- Developing solutions that use blob storage
- Developing solutions that use Cosmos DB storage
- Implementing IaaS solutions
- Implementing user authentication and authorization
- Implementing secure cloud solutions
- Implementing API Management
- Developing App Service Logic Apps
- Developing event-based solutions
- Developing message-based solutions
- Monitoring and optimizing Azure solutions
- Integrating caching and content delivery within solutions

Prerequisites

- Professionals who want to pass the Azure Developer Associate Certification Exam and understand cloud concepts and the Azure portal.

- Candidates thinking to pursue this certification should have at least one year of experience in developing solutions. This experience should be in all phases of software development.
- If you are someone new to the domain, it is recommended to complete the AZ-900: Azure Fundamentals Course.

Curriculum

Module 1: Implement Azure App Service web apps

- Explore Azure App Service
  - Describe Azure App Service key components and value.
  - Explain how Azure App Service manages authentication and authorization.
  - Identify methods to control inbound and outbound traffic to your web app.
  - Deploy an app to App Service using Azure CLI commands.
- Configure web app settings
  - Create application settings that are bound to deployment slots.
  - Explain the options for installing SSL/TLS certificates for your app.
  - Enable diagnostic logging for your app to aid in monitoring and debugging.
  - Create virtual app to directory mappings.
- Scale apps in Azure App Service
  - Identify scenarios for which autoscaling is an appropriate solution.
  - Create autoscaling rules for a web app.
  - Monitor the effects of autoscaling.
- Explore Azure App Service deployment slots
  - Describe the benefits of using deployment slots.
  - Understand how slot swapping operates in App Service.
  - Perform manual swaps and enable auto swap.
  - Route traffic manually and automatically.

Module 2: Implement Azure Functions

- Explore Azure Functions
  - Explain functional differences between Azure Functions, Azure Logic Apps, and WebJobs
  - Describe Azure Functions hosting plan options
  - Describe how Azure Functions scale to meet business needs
- Develop Azure Functions
  - Explain the key components of a function and how they are structured
  - Create triggers and bindings to control when a function runs and where the output is directed
  - Connect a function to services in Azure
  - Create a function by using Visual Studio Code and the Azure Functions Core Tools

Module 3: Develop solutions that use Blob storage

- Explore Azure Blob storage
  - Understanding Azure Blob Storage: Features, Types of Storage Accounts, and Access Tiers
  - Understanding Azure Blob Storage: Storage Accounts, Containers, and Blobs
  - Understanding Azure Storage Security and Encryption Features
- Manage the Azure Blob storage lifecycle
  - Describe how each of the access tiers is optimized.
  - Create and implement a lifecycle policy.
  - Rehydrate blob data stored in an archive tier.

- Work with Azure Blob storage
  - Create an application to create and manipulate data by using the Azure Storage client library for Blob storage.
  - Manage container properties and metadata by using .NET and REST.

Module 4: Develop solutions that use Azure Cosmos DB

- Explore Azure Cosmos DB
  - Identify the key benefits provided by Azure Cosmos DB
  - Describe the elements in an Azure Cosmos DB account and how they are organized
  - Explain the different consistency levels and choose the correct one for your project
  - Explore the APIs supported in Azure Cosmos DB and choose the appropriate API for your solution
  - Describe how request units impact costs
  - Create Azure Cosmos DB resources by using the Azure portal.
- Work with Azure Cosmos DB
  - Identify classes and methods used to create resources
  - Create resources by using the Azure Cosmos DB .NET v3 SDK
  - Write stored procedures, triggers, and user-defined functions by using JavaScript

Module 5: Implement containerized solutions

- Manage container images in Azure Container Registry
  - Explain the features and benefits Azure Container Registry offers.
  - Describe how to use ACR Tasks to automate builds and deployments.
  - Explain the elements in a Dockerfile.
  - Build and run an image in the ACR by using Azure CLI.
- Run container images in Azure Container Instances
  - Describe the benefits of Azure Container Instances and how resources are grouped.
  - Deploy a container instance in Azure by using the Azure CLI.
  - Start and stop containers using policies.
  - Set environment variables in your container instances.
  - Mount file shares in your container instances.
- Implement Azure Container Apps
  - Describe the benefits of Azure Container Instances and how resources are grouped
  - Deploy a container instance in Azure by using the Azure CLI
  - Start and stop containers using policies
  - Set environment variables in your container instances
  - Mount file shares in your container instances

Module 6: Implement user authentication and authorization

- Explore the Microsoft identity platform
  - Identify the components of the Microsoft identity platform.
  - Describe the three types of service principals and how they relate to application objects.
  - Explain how permissions and user consent operate, and how conditional access impacts your application.
- Implement authentication by using the Microsoft Authentication Library
  - Explain the benefits of using Microsoft Authentication Library and the application types and scenarios it supports.
  - Instantiate both public and confidential client apps from code.
  - Register an app with the Microsoft identity platform.
  - Create an app that retrieves a token by using the MSAL.NET library.

- Implement shared access signatures
  - Identify the three types of shared access signatures.
  - Explain when to implement shared access signatures.
  - Create a stored access policy.
- Explore Microsoft Graph
  - Explain the benefits of using Microsoft Graph.
  - Perform operations on Microsoft Graph by using REST and SDKs.
  - Apply best practices to help your applications get the most out of Microsoft Graph.

Module 7: Implement secure Azure solutions

- Implement Azure Key Vault
  - Describe the benefits of using Azure Key Vault
  - Explain how to authenticate to Azure Key Vault
  - Set and retrieve a secret from Azure Key Vault by using the Azure CLI
- Implement managed identities
  - Explain the differences between the two types of managed identities
  - Describe the flows for user- and system-assigned managed identities
  - Configure managed identities
  - Acquire access tokens by using REST and code
- Implement Azure App Configuration
  - Explain the benefits of using Azure App Configuration
  - Describe how Azure App Configuration stores information
  - Implement feature management
  - Securely access your app configuration information

Module 8: Implement API Management

- Explore API Management
  - Describe the components (and their functions) of the API Management service.
  - Explain how API gateways can help manage calls to your APIs.
  - Secure access to APIs by using subscriptions and certificates.
  - Create a backend API.

Module 9: Develop event-based solutions

- Explore Azure Event Grid
  - Describe how Event Grid operates and how it connects to services and event handlers.
  - Explain how Event Grid delivers events and how it handles errors.
  - Implement authentication and authorization.
  - Route custom events to web endpoint by using Azure CLI.
- Explore Azure Event Hubs
  - Describe the benefits of using Event Hubs and how it captures streaming data.
  - Explain how to process events.
  - Perform common operations with the Event Hubs client library.

Module 10: Develop message-based solutions

- Discover Azure message queues
  - Choose the appropriate queue mechanism for your solution.
  - Explain how the messaging entities that form the core capabilities of Service Bus operate.
  - Send and receive message from a Service Bus queue by using .NET.
  - Identify the key components of Azure Queue Storage
  - Create queues and manage messages in Azure Queue Storage by using .NET.

Module 11: Troubleshoot solutions by using Application Insights

- Monitor app performance
  - Explain how Azure Monitor operates as the center of monitoring in Azure.
  - Describe how Application Insights works and how it collects events and metrics.
  - Instrument an app for monitoring, perform availability tests, and use Application Map to help you monitor performance and troubleshoot issues.

Module 12: Implement caching for solutions

- Develop for Azure Cache for Redis
  - Explain the key scenarios Azure Cache for Redis covers and its service tiers.
  - Identify the key parameters for creating an Azure Cache for Redis instance and interact with the cache.
  - Connect an app to Azure Cache for Redis by using .NET Core.
- Develop for storage on CDNs
  - Explain how the Azure Content Delivery Network works and how it can improve the user experience.
  - Control caching behavior and purge content.
  - Perform actions on Azure CDN by using the Azure CDN Library for .NET.

LAB Outline

- Build a web application on the Azure platform as a service offerings
- Implement task processing logic by using Azure Functions
- Retrieve Azure Storage resources and metadata by using the Azure Storage SDK for .NET
- Construct a polyglot data solution
- Deploy compute workloads by using images and containers
- Authenticate by using OpenID Connect, MSAL, and .NET SDKs
- Access resource secrets more securely across services
- Create a multi-tier solution by using Azure services
- Publish and subscribe to Event Grid events
- Asynchronously process messages by using Azure Service Bus Queues
- Monitor services that are deployed to Azure
- Enhance a web application by using the Azure Content Delivery Network
- Create and configure an Azure web app (Optional)
- Implement containerized solutions (Optional)

For any query Contact Us – Microtek Learning