Exam AZ-202: Microsoft Azure Developer Certification Transition – Skills Measured

Develop for cloud storage (15-20%)

Develop solutions that use file storage

- implement quotas for file shares in storage account
- move items in file shares between containers asynchronously
- set file storage container properties in metadata

Develop solutions that use a relational database

- create, read, update, and delete database tables by using code
- implement dynamic data masking

Create Platform as a Service (PaaS) Solutions (30-35%)

Create an app service Logic App

- create a custom connector for Logic Apps, a custom template for a Logic App
- create a Logic App
- package an Azure App Service Logic App

Create app or service that runs on Service Fabric

- develop a stateful Reliable Service and a stateless Reliable Service
- develop an actor-based Reliable Service
- write code to consume Reliable Collections in your service

Schedule bulk operations

- define the batch output and conditions by using Batch Service API
- write code to run a batch job
- run a batch job by using Azure CLI, Azure Portal, and other tools

Design and develop applications that run in containers

- configure diagnostic settings on resources
- create a container image by using a Docker file

- create an Azure Container Service (ACS/AKS) cluster by using the Azure CLI and Azure Portal
- publish an image to the Azure Container Registry
- implement an application that runs on an Azure Container Instance
- implement container instances by using Azure Container Service (ACS/AKS), Azure Service Fabric, and other tools
- manage container settings by using code

Secure cloud solutions (5-10%)

Implement access control

- implement Claims-Based Access Control (CBAC) and Role-Based Access Control (RBAC) authorization

Develop for an Azure cloud model (25-30%)

Develop for asynchronous processing

- implement parallelism, multithreading, processing, durable functions, Azure logic apps, interfaces with storage, interfaces to data access, and appropriate asynchronous compute models

Develop for autoscaling

- implement autoscaling rules and patterns (schedule, operational/system metrics, code that addresses singleton application instances, and code that addresses transient state

Implement distributed transactions

- identify tools to implement distributed transactions (e.g., ADO.NET, elastic transactions, multi-database transactions)
- manage transaction scope
- manage transactions across multiple databases and servers

Implement cloud integration solutions (15-20%)

Configure a message-based integration architecture

- configure an app or service to send emails, Event Grid, and the Azure Relay Service
- create and configure a Notification Hub, an Event Hub, and a Service Bus
- configure queries across multiple products
- configure an app or service with Microsoft Graph

Develop an application message model

- create a message schema and a message exchange
- create an event model
- create topics and subscriptions