Azure Databricks Course and Training

Curriculum 7 Days

Day - 1

Getting Started with Azure

- **Describe Cloud Concepts for Data**
  - Identify the benefits and considerations of using cloud services
  - Identify the benefits of cloud computing, such as High Availability, Scalability, Elasticity,
  - Agility, and Disaster Recovery
  - Identify the differences between Capital Expenditure (CapEx) and Operational
  - Expenditure (OpEx)
  - Describe the consumption-based model
  - Describe the differences between categories of cloud services
  - Describe the shared responsibility model
  - Describe Infrastructure-as-a-Service (IaaS),
  - Describe Platform-as-a-Service (PaaS)
  - Describe server less computing
  - Describe Software-as-a-Service (SaaS)
  - Identify a service type based on a use case
  - Describe the differences between types of cloud computing
  - Define cloud computing
  - Describe Public cloud
  - Describe Private cloud
  - Describe Hybrid cloud
  - Compare and contrast the three types of cloud computing.

- **Describe core solutions and management tools on Azure**
  - Describe the benefits and usage of Internet of Things (IoT) Hub, IoT Central
  - Describe the benefits and usage of Azure Synapse Analytics, HDInsight, and Azure Data bricks
- **Describe the benefits and usage of server less computing solutions that include Azure**
  - Functions and Logic Apps for data
  - Describe the benefits and usage of Azure DevOps, GitHub, GitHub Actions, and Azure
  - DevTest Labs
- **Develop for Azure storage**
  - Understanding Data Lake, Delta Lake for data processing and storage
  - Develop solutions that use storage tables
  - Design and implement policies for tables
  - Query table storage by using code
  - Implement partitioning schemes

Day - 2	Day - 3

Azure Data Platform – RDBMS (SQL Server and PostGre), NoSQL – Cosmo DB
• **Develop solutions that use Cosmos DB storage**
  ○ Create, read, update, and delete data by using appropriate APIs
  ○ Implement partitioning schemes
  ○ Set the appropriate consistency level for operations
• **Develop solutions that use a relational database**
  ○ provision and configure relational databases
  ○ configure elastic pools for Azure SQL Database
  ○ create, read, update, and delete data tables by using code
  ○ provision and configure Azure SQL Database serverless instances
  ○ provision and configure Azure SQL and Azure PostgreSQL Hyperscale instances
• **Develop solutions that use blob storage**
  ○ move items in Blob storage between storage accounts or containers
  ○ set and retrieve properties and metadata
  ○ implement blob leasing
  ○ implement data archiving and retention
  ○ implement Geo Zone Redundant Storage
	Data Lake – Azure Storage Account – Blobs, Containers, File Share etc
	• **Implement access control for data**
	  ○ mplement CBAC (Claims-Based Access Control) authorization
	  ○ Implement RBAC (Role-Based Access Control) authorization
	  ○ Create shared access signatures
	• **Implement secure data solutions**
	  ○ Encrypt and decrypt data at rest and in transit
	  ○ Create, read, update, and delete keys, secrets, and certificates by using the Key Vault API

Day - 4

Azure Data Factory (Pipeline, ETL Process)

- **Design and Develop Data Processing**

- **Ingest and transform data**
  - Transform data by using Apache Spark
  - Transform data by using Transact-SQL
  - Transform data by using Data Factory
  - Transform data by using Azure Synapse Pipelines
  - Transform data by using Stream Analytics
  - Cleanse data
  - Split data
  - Shred JSON
  - Encode and decode data
  - Configure error handling for the transformation
  - Normalize and deformalize values
  - Transform data by using Scala
  - Perform data exploratory analysis

Day - 5

Azure Data bricks (Clusters, PySPark, Jobs) – ETL Process and Analysis

- **Design and develop a batch processing solution**
  - Design and develop a batch processing solution
  - develop batch processing solutions by using Data Factory, Data Lake, Spark, Azure
  - Synapse Pipelines, PolyBase, and Azure Databricks
  - create data pipelines
  - design and implement incremental data loads
  - design and develop slowly changing dimensions
  - handle security and compliance requirements
  - scale resources
  - configure the batch size
  - design and create tests for data pipelines
  - integrate Jupyter/Python notebooks into a data pipeline
  - handle duplicate data
  - handle missing data
  - handle late-arriving data
  - upsert data
  - regress to a previous state
  - design and configure exception handling
  - configure batch retention
  - design a batch processing solution

- **Debug Spark jobs by using the Spark UI develop a stream processing solution by using Stream Analytics**
  - Azure Databricks
  - Azure Event Hubs
  - process data by using Spark structured streaming
  - monitor for performance and functional regressions
  - design and create windowed aggregates
  - handle schema drift
  - process time series data
  - process across partitions
  - process within one partition
  - configure checkpoints/watermarking during processing
  - scale resources
  - design and create tests for data pipelines
  - optimize pipelines for analytical or transactional purposes
  - handle interruptions
  - design and configure exception handling
  - upsert data
  - replay archived stream data
  - design a stream processing solution

Day - 6

Azure Data Bricks and ADF Integration, Monitoring, troubleshooting, Maintenance

- **Manage batches and pipelines**
  - Trigger batches
  - Handle failed batch loads
  - Validate batch loads
  - Manage data pipelines in Data Factory/Synapse Pipelines
  - Schedule data pipelines in Data Factory/Synapse Pipelines
  - Implement version control for pipeline artifacts
  - Manage Spark jobs in a pipeline
- **Monitor and Optimize Data Storage and Data Processing Monitor data storage and data processing**
  - Mplement logging used by Azure Monitor
  - Configure monitoring services
  - Measure performance of data movement
  - Monitor and update statistics about data across a system
  - Monitor data pipeline performance
  - Measure query performance
  - Monitor cluster performance
  - Understand custom logging options

Day - 7

Azure Automation with ARM Templates

- **Schedule and monitor pipeline tests**
  - mplement logging used by Azure Monitor
  - Configure monitoring services
  - Measure performance of data movement
  - Monitor and update statistics about data across a system
  - Monitor data pipeline performance
  - Measure query performance
  - Monitor cluster performance
  - Understand custom logging options
- **Optimize and troubleshoot data storage and data processing**
  - Compact small files
  - Rewrite user-defined functions (UDFs)
  - Handle skew in data
  - Handle data spill
  - Tune shuffle partitions
  - Find shuffling in a pipeline
  - Optimize resource management
  - Tune queries by using indexers
  - Tune queries by using cache
  - Optimize pipelines for analytical or transactional purposes
  - Optimize pipeline for descriptive versus analytical workloads
  - Troubleshoot a failed spark job
  - Troubleshoot a failed pipeline run