DP-203: Data Engineering on Microsoft Azure

STRUCTURE

Data Engineering on Microsoft Azure (DP-203)

About Croma Campus:

Croma Campus Training & Development Private Limited is an education platform since 2010 providing rigorous industry-relevant programs designed and delivered in collaboration with world-class faculty and industry.

- Hands-On Live Projects
- Simulation Test Papers
- Industry Cases Studies
- 61,640+ Satisfied Learners
- 140+ Training Courses
- 100% Certification Passing Rate
- Live Instructor Classroom / Online Training
- 100% Placement Assistance

Croma Campus Training Program Deliverables:

- **Session Recordings** - Original Class Room Voice & Video Recording
- **Training Material** - Soft Copy Handbooks
- **Assignments** | Multiple Hands-on Exercises
- **Test Papers** - We provide **Practice Test** as part of our course to help you prepare for the actual certification exam.
- **Live Case Studies**
- **Live Projects** - Hands-on exercises and Project work. You will work on real time industry-oriented projects and assignments for each module to practice.
- **Key focus on Hands-on exercises and Project work**. You will work on real time industry-oriented projects.
- Faculty with more than **10+ Years of Experience** in the Industry.
- **Technical Resume Designing & Job Assistance:** With more than 100+ Clients across the Globe and we help learners to get a good job in their respective field. We also help learners with resume preparation.
- **Interview Q&A**
- **About Croma Campus Training Certificate:** Croma Campus will provide you with an industry-recognized (Certified by **ISO 9001:2015** & **E-Cell IIT Jodhpur**) course completion certificate, which has lifelong validity.
- **How I unlock my Croma Campus Certificate:** Attend Complete Batch & Submit at least One Completed Project.

Training Description (Data Engineering on Microsoft Azure):

Azure Data Engineers help stakeholders understand the data through exploration, and they build and maintain secure and compliant data processing pipelines by using different tools and techniques. These professionals use various Azure data services and languages to store and produce cleansed and enhanced datasets for analysis.

Azure Data Engineers also help ensure that data pipelines and data stores are high-performing, efficient, organized, and reliable, given a set of business requirements and constraints. They deal with unanticipated issues swiftly, and they minimize data loss. They also design, implement, monitor, and optimize data platforms to meet the data pipelines needs.

A candidate for this exam must have strong knowledge of data processing languages such as SQL, Python, or Scala, and they need to understand parallel processing and data architecture patterns.

Here are some strong reasons why should you consider this certification course.

- Validate your Azure Fundamental skills like storage, networking, compute, security, and other Cloud operations on Microsoft Azure.
- Validate your basic skills and learn how Azure can help you in designing robust cloud solutions
- Top-paying info-tech certification in the world.
- It provides you with global recognition for your knowledge, skills, and experience.
- The organization looks for those who know Oracle Cloud, AWS, Azure, etc.

Necessary Details about Certification You Must Know

- **Certification Name:** Data Engineering on Microsoft Azure
- **Exam Code:** DP-203
- **Exam Duration:** 150 Minutes
- **Exam Format:** Multiple Choice and Multi-Response Questions
- **Number of Questions:** 40-60
- **Passing Score:** 700 (Out of 1000)
- **Exam Cost:** USD 165.00
- **Validity:** 2 Years

Certification Exam Structure:

- Design and Implement Data Storage (40-45%)
- Design and Develop Data Processing (25-30%)
- Design and Implement Data Security (10-15%)
- Monitor and Optimize Data Storage and Data Processing (10-15%)

Training Curriculum:

Module 1: Design and Implement Data Storage

- **Design a Data Storage Structure**
  - Design an Azure Data Lake solution
  - Recommend file types for storage
  - Recommend file types for analytical queries
  - Design for efficient querying
  - Design for data pruning
  - Design a folder structure that represents the levels of data transformation
  - Design a distribution strategy
- **Design a Partition Strategy**
  - Design a partition strategy for files
  - Design a partition strategy for analytical workloads
  - Design a partition strategy for efficiency/performance
  - Design a partition strategy for Azure Synapse Analytics
  - Identify when partitioning is needed in Azure Data Lake Storage Gen2
- **Design the Serving Layer**
  - Design star schemas
  - Design slowly changing dimensions
  - Design a dimensional hierarchy
  - Design a solution for temporal data
  - Design for incremental loading
  - Design analytical stores
  - Design meta stores in Azure Synapse Analytics and Azure Data bricks
- **Implement Physical Data Storage Structures**
  - Implement compression
  - Implement partitioning
  - Implement sharing
  - Implement different table geometries with Azure Synapse Analytics pools
  - Implement data redundancy
  - Implement distributions
  - Implement data archiving
- **Implement Logical Data Structures**
  - Build a temporal data solution
  - Build a slowly changing dimension
  - Build a logical folder structure
  - Build external tables
  - Implement file and folder structures for efficient querying and data pruning
- **Implement the Serving Layer**
  - Deliver data in a relational star schema
  - Deliver data in Parquet files
  - Maintain metadata
  - Implement a dimensional hierarchy

Module 2: 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
- **Design and Develop a Batch Processing Solution**
  - Develop batch processing solutions by using Data Factory, Data Lake, Spark, Azure
  - Synapse Pipelines, Polybasic, and Azure Data bricks
  - 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
  - Upset 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
- **Design and Develop a Stream Processing Solution**
  - Develop a stream processing solution by using Stream Analytics, Azure Data bricks, and
  - 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
- Upset data
- Replay archived stream data
- Design a stream processing solution

- **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 artefacts
  - Manage Spark jobs in a pipeline

Module 3: Design and Implement Data Security

- **Design Security for Data Policies and Standards**
  - Design data encryption for data at rest and in transit
  - Design a data auditing strategy
  - Design a data masking strategy
  - Design for data privacy
  - Design a data retention policy
  - Design to purge data based on business requirements
  - Design Azure role-based access control (Azure RBAC) and POSIX-like Access Control List
  - (ACL) for Data Lake Storage Gen2
  - Design row-level and column-level security
- **Implement Data Security**
  - Implement data masking
  - Encrypt data at rest and in motion
  - Implement row-level and column-level security
  - Implement Azure RBAC
  - Implement POSIX-like ACLs for Data Lake Storage Gen2
  - Implement a data retention policy
  - Implement a data auditing strategy
  - Manage identities, keys, and secrets across different data platform technologies
  - Implement secure endpoints (private and public)
  - Implement resource tokens in Azure Data bricks
  - Load a Data Frame with sensitive information
  - Write encrypted data to tables or Parquet files
  - Manage sensitive information

Module 4: Monitor and Optimize Data Storage and Data Processing

- **Monitor Data Storage and Data Processing**
  - Implement 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
  - Schedule and monitor pipeline tests
  - Interpret Azure Monitor metrics and logs
  - Interpret a Spark directed acyclic graph (DAG)
- **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