

# Microsoft - Implement data engineering solutions using Azure Databricks

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy  
Get Your Free Copy Now

**Course Number: DP-750T00**

**Duration: 4 days**

## Overview

### Course Description

Master end-to-end data engineering with Azure Databricks and Unity Catalog. This course moves from foundational setup to production deployment, covering environment configuration and enterprise-grade governance. Learn to build robust ingestion pipelines, implement security with Unity Catalog, and deploy optimized workloads. By the end, you will have the practical skills to implement, secure, and maintain scalable lakehouse solutions that meet rigorous enterprise requirements.

### Audience Profile

The target audience is data engineers who have fundamental knowledge of data analytics concepts, a basic understanding of cloud storage, and familiarity with data organization principles. They should be comfortable working with SQL and have experience using Python, including notebooks, for data engineering tasks. Learners are expected to have a good understanding of Azure Databricks workspaces and Unity Catalog, along with familiarity with data access patterns and core data engineering and data warehouse

concepts. In addition, they should have foundational knowledge of Azure security, including Microsoft Entra ID, and be familiar with Git version control fundamentals.

## Audience

## Course Details

## Outline

- Explore Azure Databricks
  - Get started with Azure Databricks
  - Identify Azure Databricks workloads
  - Understand key concepts
  - Data governance using Unity Catalog and Microsoft Purview
  - Exercise - Explore Azure Databricks
  - Module assessment
- Understand Azure Databricks architecture
  - Understand Azure Databricks architecture
  - Understand Unity Catalog managed storage
  - Understand external storage
  - Understand default storage
  - Module assessment
- Understand Azure Databricks Integrations
  - Understand integration with Microsoft Fabric
  - Understand integration with Power BI
  - Understand integration with VS Code
  - Understand integration with Power Platform
  - Understand integration with Copilot Studio
  - Understand integration with Microsoft Purview
  - Understand integration with Microsoft Foundry
  - Module assessment
- Select and Configure Compute in Azure Databricks
  - Choose an appropriate compute type
  - Configure compute performance
  - Configure compute features

- Install libraries for compute
- Configure compute access
- Exercise - Select and Configure Compute in Azure Databricks
- Module assessment
- Create and organize objects in Unity Catalog
  - Apply naming conventions
  - Create catalog
  - Create schema
  - Create tables and views
  - Create volumes
  - Implement DDL operations
  - Implement foreign catalog
  - Configure AI/BI Genie instructions
  - Exercise - Create and Organize Objects in Unity Catalog
  - Knowledge check
- Secure Unity Catalog objects
  - Understand query lifecycle
  - Implement access control strategies
  - Understand fine-grained access control
  - Implement row filtering and column masking
  - Access Azure Key Vault secrets
  - Authenticate data access with service principals
  - Authenticate resource access with managed identities
  - Exercise - Secure Unity Catalog Objects
  - Module assessment
- Govern Unity Catalog objects
  - Create and preserve table definitions
  - Configure ABAC with tags and policies
  - Apply data retention policies
  - Set up and manage data lineage
  - Configure audit logging
  - Design secure Delta Sharing strategy

- Exercise - Govern Unity Catalog Objects
- Module assessment
- Design and implement data modeling with Azure Databricks
  - Design ingestion logic and data source configuration
  - Choose a data ingestion tool
  - Choose a data table format
  - Design and implement a data partitioning scheme
  - Choose a slowly changing dimension (SCD) type
  - Implement a slowly changing dimension (SCD) type 2
  - Design and implement a temporal (history) table to record changes over time
  - Choose granularity on a column or table based on requirements
  - Choose managed vs unmanaged tables
  - Design and implement a clustering strategy
  - Exercise - Design and Implement Data Modeling with Azure Databricks
  - Knowledge check
- Ingest data into Unity Catalog
  - Ingest data with Lakeflow Connect
  - Ingest data with notebooks
  - Ingest data with SQL methods
  - Ingest data with CDC feed
  - Ingest data with Spark Structured Streaming
  - Ingest data with Auto Loader
  - Ingest data with Lakeflow Spark Declarative Pipelines
  - Exercise - Ingest Data into Unity Catalog
  - Module assessment
- Cleanse, transform, and load data into Unity Catalog
  - Profile data
  - Choose column data types
  - Resolve duplicates and nulls
  - Transform data with filters and aggregations
  - Transform data with joins and set operators
  - Transform data with denormalization and pivots

- Load data with merge, insert, and append
- Exercise - Cleanse, Transform, and Load Data into Unity Catalog
- Module assessment
- Implement and manage data quality constraints with Azure Databricks
  - Implement validation checks
  - Implement data type checks
  - Detect and manage schema drift
  - Manage data quality with pipeline expectations
  - Exercise - Implement and Manage Data Quality Constraints with Azure Databricks
  - Module assessment
- Design and implement data pipelines with Azure Databricks
  - Design order of operations for a pipeline
  - Choose notebook vs Lakeflow Pipelines
  - Design Lakeflow job logic
  - Design error handling in pipelines and jobs
  - Create pipeline with notebook
  - Create pipeline with Lakeflow Spark Declarative Pipelines
  - Exercise - Design and Implement Data Pipelines with Azure Databricks
  - Module assessment
- Implement Lakeflow Jobs with Azure Databricks
  - Create job setup and configuration
  - Configure job triggers
  - Schedule a job
  - Configure job alerts
  - Configure automatic restarts
  - Exercise - Implement Lakeflow Jobs with Azure Databricks
  - Module assessment
- Implement development lifecycle processes in Azure Databricks
  - Apply Git version control best practices
  - Manage branching and pull requests
  - Implement testing strategy
  - Configure and package Declarative Automation Bundles

- Deploy bundle with Databricks CLI
- Exercise - Implement Development Lifecycle Processes in Azure Databricks
- Module assessment
- Monitor, troubleshoot and optimize workloads in Azure Databricks
  - Monitor and manage cluster consumption
  - Troubleshoot and repair Lakeflow Jobs
  - Troubleshoot Spark jobs and notebooks
  - Investigate caching, skewing, spilling, shuffle
  - Implement log streaming with Azure Log Analytics
  - Exercise - Monitor, Troubleshoot and Optimize Workloads in Azure Databricks
  - Module assessment