Mastering Azure Databricks-From Foundations to Production Excellence

Course Description

This immersive 5-day training equips data engineers, analysts, and architects with expert-level skills to build, optimize, and operationalize scalable data pipelines on Azure Databricks. Participants dive deep into Spark-based ETL/ELT, Delta Lake, streaming, governance, orchestration, and testing strategies, enabling end-to-end deployment of enterprise-grade data solutions. Through hands-on labs, real-world scenarios, and best practices, you'll master cost-efficient, high-performance Databricks workflows in the Azure ecosystem.

Duration: 5 Days (40 hours)

Pre-requisites:

- Basic knowledge of Python or Scala programming
- Familiarity with SQL and data concepts
- Experience with Azure fundamentals (e.g., storage, compute)
- Understanding of ETL/ELT processes (recommended)

Learning Objectives

By the end of this course, participants will be able to:

- Architect and implement scalable data pipelines using Azure Databricks and Delta Lake

- Design ETL/ELT workflows with Spark, including structured streaming and medallion architecture
- Govern data assets with Databricks SQL and Unity Catalog
- Orchestrate jobs, implement CI/CD pipelines, and optimize for performance and cost
- Build and test data quality frameworks, integrations, and UAT-ready ETL jobs
- Apply production best practices for reliable, efficient Databricks deployments

Content Coverage

Module 1: Introduction to Azure Databricks

- Overview of Databricks platform architecture and Azure integration
- Key components: clusters, notebooks, jobs, and workspaces
- Databricks vs. open-source Spark: managed services and benefits
- Setting up Azure Databricks workspace and first cluster
- Role-based access and security fundamentals
- Use cases in data engineering, analytics, and ML

Module 2: Delta Lake Fundamentals

- Core concepts: ACID transactions, schema enforcement, and time travel
- Delta Lake architecture on Databricks

- Creating and managing Delta tables
- OPTIMIZE and Z-ORDER for performance
- Delta Lake vs. Parquet: advantages and migration strategies
- Hands-on: Building initial Delta tables from data sources

Module 3: ETL/ELT with Spark on Databricks

- Spark DataFrames and Datasets in Databricks
- Reading/writing data: formats, partitioning, and caching
- Transformations: joins, aggregations, and UDFs
- Spark SQL for ELT pipelines
- Error handling and fault tolerance in Spark jobs
- Lab: End-to-end ETL pipeline with sample datasets

Module 4: Databricks ETL Architecture & Ingestion Patterns

- Common ingestion patterns: batch, micro-batch, streaming
- Auto Loader for incremental data ingestion
- Partner integrations: Azure Data Factory, Event Hubs, Blob Storage
- Scalable ingestion architectures
- Handling schema evolution and late data
- Best practices for reliable data pipelines

Module 5: Structured Streaming Testing

- Structured Streaming model: sources, sinks, and triggers

- Watermarking, stateful operations, and foreachBatch
- Testing streaming queries: unit, integration, and chaos testing
- Monitoring streaming metrics in Databricks
- Fault recovery and exactly-once semantics
- Lab: Real-time streaming pipeline with testing

Module 6: Data Modeling & Delta Lake (Bronze → Silver → Gold)

- Medallion architecture: Bronze (raw), Silver (curated), Gold (aggregated)
- Implementing layers with Delta Live Tables
- Data quality checks at each layer
- Change Data Capture (CDC) patterns
- Versioning and auditing across layers
- Hands-on: Multi-layer pipeline build

Module 7: Databricks SQL & Unity Catalog Governance

- Databricks SQL: warehouses, queries, and dashboards
- Unity Catalog: three-level namespace and metastore
- Data lineage, sharing, and access controls
- Governance policies: tagging, auditing, and compliance
- Query federation and external tables
- Securing sensitive data with dynamic views

Module 8: Orchestration (Databricks Jobs, Workflows, Triggers)

- Databricks Jobs: task graphs, dependencies, and scheduling
- Workflows with multi-task jobs and parameters
- Triggers: file arrival, cron, and external APIs
- Delta Live Tables for declarative pipelines
- Error handling, retries, and alerts
- Lab: Multi-step workflow orchestration

Module 9: CI/CD & DevOps for Databricks

- Git integration and notebook versioning
- Databricks Asset Bundles (DABs) for IaC
- CI/CD pipelines with Azure DevOps or GitHub Actions
- Environment promotion: dev/staging/prod
- Automated testing in pipelines
- Branching strategies for collaborative development

Module 10: Performance Testing on Databricks

- Cluster sizing, autoscaling, and spot instances
- Spark UI analysis: stages, tasks, and spills
- Benchmarking queries with Delta Lake optimizations
- Photon engine for accelerated performance
- Load testing tools and frameworks
- Profiling and tuning labs

Module 11: Cost Optimization Testing

- Cost monitoring: clusters, DBUs, and usage reports
- Auto-termination, scaling policies, and job clusters
- Predicting costs with Databricks tools
- Optimization techniques: caching, broadcast joins
- Testing scenarios for cost vs. performance trade-offs
- Governance for budget controls

Module 12: Data Quality & Validation Frameworks

- Great Expectations and Delta Expectations integration
- Custom validation rules in Spark/Delta Live Tables
- Anomaly detection and data profiling
- Automated quality gates in pipelines
- Reporting and alerting on quality metrics
- Hands-on: Building a DQ framework

Module 13: End-to-End Cloud Integration Testing

- Integrating with Azure services: Synapse, Purview, Sentinel
- API testing for Databricks endpoints
- Cross-service data flows and latency testing
- Security testing: RBAC, encryption, and network isolation
- Mocking external dependencies
- Full E2E test automation

Module 14: ETL Jobs Testing & UAT Preparation

- Unit testing Spark code with PyTest/ScalaTest
- Integration and end-to-end ETL testing
- UAT checklists: data accuracy, performance SLAs
- Mock data generation and regression testing
- Deployment readiness and rollback strategies