DATABRICKS

Data Engineering · Complete Training

Mastering Databricks Data Engineering — From Zero to Production

Delta Lake • Delta Live Tables • Unity Catalog • Lakeflow Jobs • PySpark
Auto Loader • Medallion Architecture • Databricks SQL • AWS • Azure

databrickstraining.in +91-8500002025 | info@databrickstraining.in

★★★★★ 5.0

Databricks | Delta Lake | PySpark | Delta Live Tables | Unity Catalog | Lakeflow Jobs

■ Duration	■ Level	■ Labs	■ Projects	■ Cert Target
120 Hours	Beginner → Expert	35+ Hands-On	3 Capstone	Databricks Certified

- ■ Complete end-to-end Databricks Data Engineering curriculum — beginner to production-ready expert
- ■ Covers Databricks Lakehouse, Delta Lake, DLT, Unity Catalog, Lakeflow Jobs, Databricks SQL
- ■ Hands-on AWS (EMR, S3, Glue) & Azure (ADLS Gen2, ADF, Synapse) integration modules
- ■ Three graded capstone projects + Databricks Certified Data Engineer Associate preparation
- ■ Resume building, salary negotiation, and 60+ interview Q&A included

Databricks Platform & Engineering Workspace

M01 | 3 Hours | Beginner

What is Databricks & the Lakehouse Architecture — unifying Data Warehouses & Data Lakes
Databricks vs Traditional Hadoop & Cloud Warehouses (Redshift, Synapse, BigQuery)
Databricks Components Overview
• Engineering Workspace — Notebooks, Clusters, DBFS
• Databricks SQL — BI & ad-hoc queries
• Machine Learning — MLflow, Feature Store
• Delta Live Tables — Declarative pipelines
• Unity Catalog — Enterprise governance
Databricks on AWS vs Azure vs GCP — Key Differences
Databricks Architecture — Control Plane vs Data Plane
Workspace Setup — Create & Configure Databricks Workspace
Databricks Notebook — Cell types (Python, SQL, Markdown, Scala, R)
Databricks File System (DBFS) — Browse & manage files
Databricks CLI Setup & Configuration
Databricks REST API — Automate workspace operations
Databricks Repos — Git integration for version-controlled notebooks
Lab: Create Databricks Workspace, Run First Notebook, Connect to DBFS

Databricks Cluster Architecture & Configuration

M02 | 3 Hours | Beginner-Intermediate

Cluster Types
• All-Purpose Clusters — Interactive development
• Job Clusters — Single-use pipeline execution
• SQL Warehouses — BI query endpoints
Cluster Configuration — Driver & Worker nodes
• Node types — Memory-optimised, Compute-optimised, GPU
• Databricks Runtime (DBR) — Standard, ML, Photon
• Auto-scaling — Min/Max workers
• Auto-termination — Cost control
Cluster Policies — Govern cluster creation & settings
Instance Pools — Pre-warmed VMs for fast start
Spot/Preemptible instances — Cost optimisation (AWS & Azure)
Photon Engine — Vectorised C++ execution engine
• When to use Photon — SQL-heavy, Delta operations
• Photon vs Standard DBR benchmarks
DBU (Databricks Unit) Pricing — Understand cloud costs
Cluster Libraries — Attach PyPI, Maven, DBFS libraries
Init Scripts — Configure clusters at startup
Lab: Create Cluster Policies, Configure Instance Pools, Benchmark Photon vs Standard

Delta Lake — The Foundation

M03 | ■ 5 Hours | ● Intermediate

What is Delta Lake & Why It Replaced Parquet
• ACID Transactions on data lakes
• Scalable metadata handling
• Unification of batch & streaming
Delta Lake Architecture — Transaction Log (_delta_log)
• How Delta guarantees ACID — JSON commit files
• Checkpoint files — Snapshot state
Creating Delta Tables
• CREATE TABLE ... USING DELTA
• CONVERT TO DELTA — Migrate Parquet to Delta
• Managed vs External Delta Tables
Delta DML Operations
• INSERT, UPDATE, DELETE — Row-level mutations
• MERGE (upsert) — CDC & SCD Type 1/2 patterns
Delta Time Travel
• VERSION AS OF / TIMESTAMP AS OF
• RESTORE TABLE to previous version
• DESCRIBE HISTORY — Audit all changes
Delta Schema Management
• Schema enforcement — Reject bad data
• Schema evolution — Auto-update on new columns
Delta Performance Optimisation
• OPTIMIZE — Compact small files
• Z-ORDER BY — Co-locate related data
• Liquid Clustering — Auto-optimise layout (Databricks 13.3+)
• VACUUM — Remove stale files, retention config
Delta Table Statistics — Collect for query planning
Change Data Feed (CDF) — Track row-level changes
Delta Lake vs Apache Iceberg vs Apache Hudi
Labs: Delta CRUD + MERGE, Time Travel Recovery, OPTIMIZE + Z-ORDER Benchmark
Quiz: Delta Lake — 20 Questions

Relational Entities on Databricks

M04 | ■ 3 Hours | ● Intermediate

Databricks Metastore — Legacy Hive Metastore vs Unity Catalog
Three-Level Namespace — Catalog → Schema (Database) → Table
Creating Catalogs, Schemas, Tables in Databricks SQL
Managed Tables vs External Tables
• Managed — Databricks owns lifecycle & location
• External — Bring your own S3/ADLS location
Views — Standard, Materialized, Dynamic
Temporary Views & Global Temp Views
Database Objects in Databricks
• Functions — User-defined & built-in

• Stored Procedures — SQL scripting
• Constraints — Primary Key, Foreign Key, NOT NULL (informational)
Table Properties — TBLPROPERTIES for custom metadata
Data Types — All supported types in Databricks SQL
DESCRIBE TABLE / DESCRIBE EXTENDED / DESCRIBE HISTORY
SHOW TABLES, SHOW DATABASES, SHOW COLUMNS
DROP TABLE — Managed vs External behaviour
Lab: Build Three-Level Namespace Hierarchy, Create Managed & External Delta Tables

ETL with Spark SQL

M05 | ■ 5 Hours | ● Intermediate

Spark SQL Architecture — Catalyst Optimizer & Tungsten Engine
Databricks Notebook SQL cells vs %sql magic
DDL — CREATE, ALTER, DROP tables & schemas
DML — INSERT, INSERT OVERWRITE, INSERT INTO SELECT
Advanced SELECT
• CTEs — WITH clause for readable queries
• Subqueries — Scalar, Correlated
• PIVOT & UNPIVOT operations
• QUALIFY — Filter window function results
Joins in Spark SQL — Inner, Left, Right, Full, Cross, Semi, Anti
Aggregations — GROUP BY, HAVING, ROLLUP, CUBE, GROUPING SETS
Window Functions
• ROW_NUMBER, RANK, DENSE_RANK, NTILE
• LEAD, LAG — Offset row access
• FIRST_VALUE, LAST_VALUE
• Running Totals, Moving Averages
Semi-Structured Data — Working with JSON & Arrays
• from_json(), to_json(), schema_of_json()
• explode(), explode_outer(), posexplode()
• get_json_object(), json_tuple()
• FLATTEN, ARRAY_AGG, FORALL, EXISTS, FILTER
String Functions — split, concat, trim, regexp_replace, like
Date & Timestamp Functions — date_trunc, datediff, date_add, to_date
Higher-Order Functions — TRANSFORM, FILTER, REDUCE on arrays
COPY INTO — Load files from cloud storage to Delta
INSERT OVERWRITE PARTITION — Atomic partition replacement
Labs: Complex SQL ETL Pipeline, JSON Parsing & Flattening, Window Function Analytics
Quiz: Spark SQL & ETL — 20 Questions

Just Enough Python for Spark — PySpark

M06 | ■ 6 Hours | ● Intermediate

Python Fundamentals for Data Engineers
• Variables, data types, f-strings
• Lists, dicts, sets, tuples
• List comprehensions & dict comprehensions
• Functions — def, lambda, *args, **kwargs
• Error handling — try/except/finally
• File I/O — open, read, write
SparkSession — Entry point in Databricks
PySpark DataFrame API — Core Operations
• Reading: CSV, JSON, Parquet, Delta, JDBC
• Writing: overwrite, append, partitionBy
• select, filter, where, withColumn, drop, rename
• distinct, dropDuplicates, sort, orderBy, limit

• groupBy, agg, pivot, rollup, cube
• join (inner/left/right/full/semi/anti)
• union, unionByName, intersect, except
Schema Management
• StructType & StructField — Define schema explicitly
• InferSchema — When to use vs explicit
Working with Complex Types in PySpark
• explode(), flatten(), struct(), array(), map()
• Nested JSON — get_json_object, from_json
Null Handling — dropna(), fillna(), coalesce(), isNull()
PySpark Built-in Functions — pyspark.sql.functions (F)
UDFs — User Defined Functions
• Python UDFs — Flexible, slower
• Pandas UDFs (Vectorized) — High performance
• UDF registration for SQL use
DataFrame vs SQL — switch between both freely
Pandas on Spark (Koalas) — pandas API on Spark
PySpark Best Practices — Avoid collect(), use F. functions
Labs: PySpark ETL Pipeline, Pandas UDF for ML feature engineering, Complex type handling
Quiz: PySpark — 20 Questions

Incremental Data Processing — Auto Loader & Structured Streaming

M07 | 5 Hours | Intermediate-Advanced

Why Incremental Processing — Full reload vs incremental
Auto Loader — Databricks Incremental Ingestion
• cloudFiles format — stream new files only
• Supported sources: S3, ADLS Gen2, GCS, DBFS
• Schema inference & schema evolution in Auto Loader
• Rescue column — capture bad records
• File notification mode (SQS/Event Grid) vs directory listing
• Checkpointing — Fault-tolerant incremental state
• Auto Loader with Unity Catalog external locations
Structured Streaming — Spark's Real-Time Engine
• Micro-Batch vs Continuous processing
• Input sources: Kafka, Event Hubs, Auto Loader, Delta
• Output sinks: Delta Lake, Kafka, Console, Memory
• Output modes: Append, Update, Complete
• Trigger modes: ProcessingTime, Once, AvailableNow, Continuous
Watermarking — Late-Arriving Data Handling
• withWatermark() — Define event-time threshold
• Late data drop vs include strategies
Stateful Streaming — Running Aggregations
• Window operations: Tumbling, Sliding, Session
• mapGroupsWithState & flatMapGroupsWithState
Stream-Static Joins & Stream-Stream Joins
Streaming Deduplication — dropDuplicates with watermark
Checkpointing & Fault Tolerance
• Checkpoint location config
• Recover from failures
Exactly-Once Semantics — Delta Lake guarantees
foreachBatch — Write to multiple sinks
Labs: Auto Loader ADLS → Delta, Kafka → Structured Streaming → Delta, Windowed Aggregations
Quiz: Streaming & Auto Loader — 20 Questions

Medallion Architecture in the Data Lakehouse

M08 | 4 Hours | Intermediate-Advanced

What is Medallion Architecture — Bronze, Silver, Gold
• Bronze Layer — Raw landing zone (exact copy, schema-on-read)
• Silver Layer — Cleansed, conformed, enriched data
• Gold Layer — Business-ready aggregates for analytics
Why Medallion Architecture — Progressive quality improvement
Designing the Bronze Layer
• Auto Loader ingestion → Bronze Delta tables
• Capture all raw data including bad records
• Partitioning strategy for Bronze (by date/source)
Designing the Silver Layer
• Deduplication — dropDuplicates + watermark

• Type casting & NULL handling
• SCD Type 1 — Overwrite pattern with MERGE
• SCD Type 2 — Full history with valid_from/valid_to
• Data quality validation — Great Expectations on Databricks

Designing the Gold Layer

• Aggregated fact tables (fct_ prefix)
• Dimension tables (dim_ prefix)
• Star schema vs Flat tables for BI tools
• Databricks SQL views on Gold for reporting

Multi-Hop Architecture — Chaining Bronze → Silver → Gold

Medallion with Delta Live Tables — Full automation

Medallion with Databricks Workflows — Orchestrate layers

Cost Governance — Lifecycle policies per layer

Medallion on AWS — S3 bucket structure + EMR/Glue

Medallion on Azure — ADLS Gen2 containers + ADF

Labs: Build Full Bronze→Silver→Gold Pipeline, SCD Type 2 Silver layer, Gold reporting tables

Delta Live Tables (DLT) — Declarative Pipelines

M09 | 6 Hours | Advanced

What is Delta Live Tables & Why Use It
• Declarative syntax — Define WHAT, not HOW
• Automatic dependency management & ordering
• Built-in data quality enforcement
• Auto-scaling & auto-recovery
DLT vs Traditional Spark Pipelines — Key Differences
DLT Table Types
• Live Tables — Materialized, refreshed by pipeline
• Streaming Live Tables — Incremental processing
• @dlt.table decorator (Python)
• CREATE LIVE TABLE / CREATE STREAMING LIVE TABLE (SQL)
DLT Expectations — Data Quality Gates
• @dlt.expect — Warn on violation
• @dlt.expect_or_drop — Drop bad records
• @dlt.expect_or_fail — Halt pipeline on violation
• Metrics — Track quality over time in Event Log
DLT Pipeline Configuration
• Triggered vs Continuous pipeline modes
• Development vs Production mode
• Target schema — Where tables land
• Cluster configuration per pipeline
DLT with Auto Loader — Incremental ingestion in DLT
DLT Change Data Capture (CDC)
• APPLY CHANGES INTO — Handle inserts, updates, deletes
• SCD Type 1 & Type 2 with APPLY CHANGES
• apply_changes() in Python DLT
DLT Parameterisation — Pipeline parameters & widgets
DLT Event Log — Monitor pipeline execution
DLT Graph UI — Visual lineage of pipeline steps
DLT with Unity Catalog — Governed DLT tables
DLT Monitoring — Expectations metrics, row counts, latency
DLT Best Practices — When to use DLT vs Jobs + PySpark
Labs: Full DLT Medallion Pipeline, DLT Expectations Quality Gates, DLT CDC with APPLY CHANGES
Quiz: Delta Live Tables — 20 Questions

Lakeflow Jobs — Task Orchestration with Databricks Workflows

M10 | 5 Hours | Advanced

What is Databricks Workflows (Lakeflow Jobs)
• Native orchestration — No external tool needed
• Databricks Jobs vs Apache Airflow — When to use each
Creating Databricks Jobs — UI, REST API, CLI, Terraform
Job Cluster vs All-Purpose Cluster for jobs
Task Types in Databricks Workflows
• Notebook Task — Run a notebook

• Python Script Task — Run .py file
• Delta Live Tables Task — Trigger DLT pipeline
• SQL Task — Run Databricks SQL query or dashboard refresh
• dbt Task — Native dbt integration
• Spark JAR Task — Run compiled Scala/Java
• Spark Python Task — Submit PySpark script
• Pipeline Task — Reference a DLT pipeline
• Run Job Task — Trigger another job

Multi-Task Workflows — DAG of tasks
• Task dependencies — depends_on
• Parallel tasks — Run independent tasks simultaneously
• Conditional branching — if/else task paths

Job Scheduling
• Cron-based scheduling
• File arrival trigger (S3/ADLS event)
• Manual trigger — on-demand
• Continuous trigger — keep running

Job Parameters — Pass runtime values to notebooks/scripts

Task Values — Share values between tasks (get_task_value/set_task_value)

Job Retry Policies — Auto-retry on failure

Job Repair — Rerun only failed tasks

Job Compute — Serverless compute for jobs

Job Monitoring — Run history, duration, logs

Job Alerts — Email & webhook on failure/success/SLA

Databricks Asset Bundles (DAB) — Deploy jobs as code
• bundle.yml — Define jobs, pipelines, permissions
• databricks bundle deploy — CI/CD for Databricks
• Multiple targets — dev, staging, production

Labs: Multi-Task Workflow with DLT + Notebook + SQL, Job CI/CD with DAB, Conditional Task Branching

Quiz: Databricks Workflows — 15 Questions

Unity Catalog — Enterprise Data Governance

M11 | ■ 6 Hours | ● Advanced

What is Unity Catalog & Why It Matters
• Centralised governance across all Databricks workspaces
• Replaces per-workspace Hive Metastore
Unity Catalog Architecture
• Metastore — Top-level governance object
• Catalogs — Logical containers for schemas
• Schemas — Container for tables, views, functions
• Three-level namespace: catalog.schema.table
Unity Catalog vs Legacy Hive Metastore — Migration guide
Setting Up Unity Catalog
• Create Metastore & assign to workspace
• External Locations — Register cloud storage paths
• Storage Credentials — Service Principals & Managed Identity
Unity Catalog Object Hierarchy & Permissions
• Securable objects: Metastore, Catalog, Schema, Table, View, Function
• Principals: Users, Groups, Service Principals
• Privileges: SELECT, MODIFY, CREATE, USAGE, ALL PRIVILEGES
• GRANT / REVOKE syntax
• Inherit vs explicit permissions
Column-Level Security — Fine-Grained Access Control
• Column Masks — Dynamic masking based on user/group
Row-Level Security — Row Filters
• Row filter functions — Filter rows per user context
• current_user(), is_member() — Context functions
Delta Sharing — Open Protocol for Data Sharing
• Share data without copying
• Recipients — External consumers
• Share tables, schemas, partitions
• Delta Sharing with Power BI, Python, Tableau
Data Lineage — End-to-End Tracing
• Automatic lineage capture — column & table level
• Lineage graph in Unity Catalog UI
Data Discovery — Search & Browse Data Assets
• Tags — Apply business metadata
• Comments — Document tables & columns
• AI-generated descriptions (Databricks AI/BI)
Audit Logs — Who accessed what & when
• Audit log delivery to S3/ADLS
• Query audit logs with Databricks SQL
System Tables — Monitor platform usage
• system.access.audit — Security events
• system.billing.usage — DBU consumption
• system.compute.clusters — Cluster history
Labs: Unity Catalog Full Setup, Column Masking + Row Filters for PII, Delta Sharing, Lineage Graph
Quiz: Unity Catalog — 20 Questions

Databricks SQL — BI & Analytics

M12

■ 4 Hours

● Intermediate

What is Databricks SQL & When to Use It

SQL Warehouses — Serverless vs Pro vs Classic

• Serverless SQL Warehouse — Zero management, instant scale

• Pro — Custom VPC & advanced features

• Classic — Legacy option

SQL Warehouse Sizing — T-Shirt sizes (XS to 4X-Large)

Auto-Stop — Idle cost control

Query Editor — Write & run SQL in browser

Query History — Performance analysis

Query Profiles — Understand execution plans

Databricks SQL Queries

• Save & share queries

• Query parameters — Dynamic filters

• Scheduled query refresh

Databricks SQL Dashboards

• AI/BI Dashboards — New native dashboards

• Legacy Dashboards — Migrate plan

• Widgets: Bar, Line, Pie, Counter, Table, Map

• Dashboard parameters — Interactive filters

• Subscribe — Email snapshot delivery

Databricks SQL Alerts

• Alert on query result threshold

• Destinations: Email, Slack, webhook

Databricks SQL with Unity Catalog — Table access & permissions

Connecting BI Tools to Databricks SQL

• Power BI — DirectQuery + Import modes

• Tableau — Partner Connect

• Looker, Metabase, Redash

Productionizing Dashboards & Queries

• Scheduled refresh, Alert on stale data

• Embed dashboards in applications

• Dashboard version history

Labs: Build AI/BI Dashboard, SQL Query with Parameters, Scheduled Refresh + Alerts

Managing Permissions in the Lakehouse

M13

■ 3 Hours ● Advanced

Permission Model in Databricks
• Account-level vs Workspace-level permissions
• Unity Catalog data permissions vs Workspace object permissions
Workspace Object Permissions
• Notebooks, Folders, Clusters, Jobs, SQL Warehouses
• Permission levels: CAN READ, CAN RUN, CAN EDIT, CAN MANAGE
Unity Catalog Data Permissions (Data Plane)
• GRANT / REVOKE on catalog, schema, table, column
• Inherited permissions — Parent → Child
Service Principals — Machine-to-Machine authentication
• Create service principal in Databricks account
• Assign to groups & grant data permissions
• Token-based authentication for CI/CD
Groups — Manage permissions at scale
• Account groups vs Workspace-local groups
• Nested groups
Personal Access Tokens (PATs) — Secure API access
OAuth — Machine-to-Machine & User-to-Machine flows
Entitlements — Control workspace capabilities
• Cluster creation, SQL access, Workspace access
IP Access Lists — Restrict access by IP range
Private Link — Secure data plane connectivity
Compliance — SOC2, HIPAA, PCI-DSS on Databricks
Lab: Build Complete RBAC Setup — Analyst, Engineer, Admin roles with least privilege

Spark Performance Optimization on Databricks

M14

■ 5 Hours ● Advanced

Spark Execution Model on Databricks — Jobs, Stages, Tasks
Spark UI — Read & Interpret on Databricks
• DAG visualization, Stage details, Task metrics
• Identify stragglers, spill, skew
Partitioning Deep Dive
• Default parallelism — spark.default.parallelism
• repartition() vs coalesce() — When to use each
• Partition by date — Best practice for Delta tables
Shuffle Optimization
• Sort Merge Join vs Broadcast Join
• spark.sql.autoBroadcastJoinThreshold — Tune broadcast
• AQE Auto-Broadcast Join
Adaptive Query Execution (AQE) — Databricks Default On
• Auto Partition Coalescing — Reduce small shuffle partitions
• Skew Join Optimization — Detect & fix automatically
• Auto Broadcast Join — Switch plan mid-query
Data Skew — Detection & Manual Fixes

• Salting technique — Distribute skewed keys
• AQE skew hint — Force skew handling

Caching & Persistence on Databricks

• Delta Cache — Automatic local SSD caching
• DataFrame.cache() vs .persist()
• CACHE TABLE / UNCACHE TABLE in SQL

File & Compaction Optimization

• Small file problem — auto-compact & optimizeWrite
• OPTIMIZE — Compact Delta files
• Z-ORDER BY — Cluster data for skip efficiency
• Liquid Clustering — No static partition needed

Predicate Pushdown & Column Pruning

Memory Management — driver vs executor, off-heap

Kryo Serialization — Faster than Java serialization

Labs: Spark UI Bottleneck Analysis, Fix Data Skew with Salting, AQE Before & After

Quiz: Spark Performance — 15 Questions

AWS Integration for Databricks Data Engineers

M15 | 4 Hours | Intermediate-Advanced

Databricks on AWS Architecture
• AWS Databricks control plane vs data plane (customer VPC)
• Databricks Workspace on AWS — VPC, subnets, security groups
Amazon S3 — Primary Cloud Storage for Databricks on AWS
• Reading S3 from Databricks — abfss vs s3a paths
• Writing Delta tables to S3
• S3 bucket policies & IAM roles for cluster access
• Unity Catalog External Locations on S3
AWS IAM — Authentication from Databricks
• Instance Profile — Attach IAM role to cluster
• IAM Role chaining — Cross-account access
• AWS Secrets Manager with Databricks secret scopes
AWS Glue Data Catalog — Use as Hive Metastore
• Glue Catalog integration with Databricks
Amazon Kinesis — Real-Time Streaming
• Read Kinesis Data Streams from Structured Streaming
• kinesis-asl connector in Databricks
AWS EMR vs Databricks — Migration guide
• Migrate EMR PySpark jobs to Databricks
AWS Lambda + Databricks — Event-driven triggers
• Lambda triggers Databricks Job via REST API
AWS Step Functions + Databricks — Orchestration
Amazon Redshift from Databricks
• JDBC connection to Redshift
• Spark-Redshift connector
Databricks + AWS Glue ETL — Hybrid pipelines
Labs: S3 + Databricks Delta Pipeline, Kinesis → Structured Streaming, AWS Secrets in Databricks

Azure Integration for Databricks Data Engineers

M16 | 4 Hours | Intermediate-Advanced

Databricks on Azure Architecture
• Azure Databricks — Native managed service
• VNET injection — Databricks in customer VNet
• Private Link — Secure data plane
Azure Data Lake Storage Gen2 (ADLS Gen2) — Primary Storage
• Reading ADLS Gen2 from Databricks — abfss:// paths
• Mounting ADLS Gen2 in Databricks (legacy) vs Direct access
• OAuth 2.0 authentication — App Registration
• Unity Catalog External Locations on ADLS Gen2
Azure Key Vault — Secret management for Databricks
• AKV-backed secret scope in Databricks
• Access secrets in notebooks: dbutils.secrets.get()
Azure Data Factory + Databricks
• ADF Notebook Activity — Run Databricks notebook from ADF

• ADF Pipeline → Databricks → ADLS Gold layer
• Pass parameters from ADF to Databricks widgets

Azure Event Hubs — Kafka-compatible Streaming
• Read Event Hubs from Databricks Structured Streaming
• Event Hubs connection string configuration

Azure Synapse Analytics + Databricks
• Synapse Link — Read Delta from Synapse Serverless
• Write Databricks Gold → Synapse Dedicated Pool

Microsoft Fabric + Databricks — Latest integration
• OneLake as unified storage layer
• Fabric Mirroring for Delta tables

Azure Active Directory — Authentication
• Service Principal auth in Databricks
• Managed Identity (MSI) for secure access

Labs: ADLS Gen2 → Databricks Delta Pipeline, ADF → Databricks → Synapse, AKV Secret Scope

Databricks Asset Bundles & CI/CD

M17

■ 4 Hours

● Advanced

What is Databricks Asset Bundles (DAB)

• Infrastructure as Code for Databricks

• Define Jobs, DLT Pipelines, SQL Queries, Permissions in YAML

DAB bundle.yml Structure

• resources: jobs, pipelines, experiments, models

• targets: dev, staging, prod environments

• workspace: host, auth config per target

• permissions: who can run & view

DAB Commands

• databricks bundle init — Create new project

• databricks bundle validate — Check YAML

• databricks bundle deploy — Deploy to workspace

• databricks bundle run — Trigger deployed job

• databricks bundle destroy — Tear down resources

Databricks Repos — Git Integration

• Link workspace folder to GitHub/GitLab/Azure DevOps

• Pull, push, branch, merge from Databricks UI

• Repos API — Automate git operations

CI/CD Pipelines for Databricks

• GitHub Actions + DAB — Full workflow

• Azure DevOps Pipelines + DAB

• GitLab CI + DAB

Testing Databricks Code

• pytest for PySpark unit tests

• Databricks pytest plugin — Run tests on cluster

• nutter — Notebook testing framework

Multi-Environment Strategy

• Dev workspace → Staging workspace → Prod workspace

• Feature flags — Control feature rollout

• Blue/Green deployment for DLT pipelines

Labs: CI/CD with GitHub Actions + DAB, Unit Tests with pytest, Multi-Env Deploy Strategy

Capstone Projects — End-to-End Databricks Pipelines

M18

■ 12 Hours

● Advanced

PROJECT 1 — Retail Analytics Medallion Platform (Graded)

Submit: GitHub repo + DLT pipeline code + Unity Catalog setup + Databricks SQL dashboard

• Source: Azure SQL DB (orders) + S3 raw event files

• Bronze: Auto Loader → ADLS Delta Bronze (raw orders, events)

• Silver: DLT pipeline — Deduplicate, validate, SCD Type 2 customers

• Gold: DLT aggregated sales by region, product, time period

• Quality: DLT Expectations on every layer

• Governance: Unity Catalog — Catalog per env, PII column masking

• Orchestration: Databricks Workflow multi-task (Auto Loader → DLT → SQL refresh)

• Reporting: Databricks SQL AI/BI Dashboard + Power BI

• CI/CD: GitHub Actions + DAB deploy to staging & prod

PROJECT 2 — Real-Time Clickstream Streaming Pipeline (Graded)

Submit: Streaming notebook + watermark proof + Delta table output + dashboard

• Source: Python Kafka producer → AWS Kinesis / Azure Event Hubs
• Processing: Databricks Structured Streaming with 5-min tumbling windows
• Late Data: Watermark — Accept events up to 2 hours late
• Exactly-Once: Delta Lake + checkpointing
• Storage: Delta Lake partitioned by date & event_type
• Output: Databricks SQL Real-Time Dashboard + Alerts
• Monitoring: Streaming metrics + Slack alerts on lag

PROJECT 3 — CDC & SCD Type 2 Pipeline (Graded)

Submit: DLT CDC code + Delta Merge logic + pytest test results

• Source: MySQL → Debezium CDC → Kafka → ADLS Bronze
• Ingestion: Auto Loader → Bronze Delta
• CDC: DLT APPLY CHANGES INTO — Capture insert/update/delete
• SCD Type 2: Full customer history with valid_from / valid_to
• Tests: pytest unit tests for merge logic
• Governance: Unity Catalog row-level security on customer table

Databricks Certification & Interview Preparation

M19

■ 5 Hours

● Advanced

Databricks Certified Data Engineer Associate — Exam Overview

• Exam format: 45 questions, 90 minutes, passing score 70%

• Exam cost: $200 USD

Exam Domain Weightage

• Databricks Lakehouse Platform (24%)

• ELT with Apache Spark (29%)

• Incremental Data Processing (22%)

• Production Pipelines (16%)

• Data Governance (9%)

Databricks Certified Data Engineer Professional — Overview

• Advanced streaming, monitoring, testing

• Security & governance deep dive

Top 60 Databricks Interview Questions & Answers

• Delta Lake architecture & ACID questions

• DLT vs Jobs vs Structured Streaming — Decision guide

• Unity Catalog permission scenarios

• Auto Loader vs COPY INTO — When to use each

• Spark performance optimization scenarios

• Medallion architecture design questions

• Real-world CDC & SCD Type 2 implementation

• Databricks SQL vs Synapse vs Redshift

Resume Building for Databricks Data Engineer Roles

• Quantify impact: '5TB/day DLT pipeline, reduced latency from 4h to 12min'

• Highlight DLT + Unity Catalog + DAB experience

• Include GitHub project links

LinkedIn Optimization

• Keywords: Databricks, Delta Lake, Unity Catalog, DLT, PySpark, Lakeflow

Salary Guide

• India: ■12-20 LPA (Entry) | ■22-35 LPA (Mid) | ■35-55 LPA (Senior)

• US: $120K-180K | UK: £80K-130K | EU: €85K-130K

BONUS TOPICS — Advanced Databricks Mastery

10 expert-level topics to differentiate yourself in the job market

1. Databricks ML & MLflow for Data Engineers

- MLflow Tracking — Log metrics, params, artifacts from notebooks
- MLflow Model Registry — Stage: Staging → Production
- Feature Store — Define, serve & share ML features
- Model Serving — Real-time inference endpoint from Databricks
- How data engineers build ML feature pipelines

2. dbt on Databricks

- dbt Core with Databricks adapter (dbt-databricks)
- dbt models on Delta Lake tables — incremental materialisation
- dbt + DLT — Hybrid approach (ingestion DLT, transform dbt)
- Orchestrate dbt with Databricks Workflows (dbt Task type)
- dbt docs + Unity Catalog lineage — unified governance

3. Databricks Lakehouse Monitoring

- Lakehouse Monitoring — Automated data & model quality monitoring
- Monitor Delta table statistics over time (drift, volume, quality)
- Inference monitoring — Track ML model accuracy post-deployment
- Custom monitors — Define metrics per table
- Integration with Databricks SQL dashboards for monitoring

4. Apache Iceberg & Open Table Formats

- Delta Lake vs Apache Iceberg vs Apache Hudi — Full comparison
- UniForm — Write Delta, read as Iceberg or Hudi
- Interoperability — Read Iceberg from Databricks
- Delta Lake open-source — External engines reading Delta
- OneLake (Microsoft Fabric) + Delta Sharing

5. Databricks Serverless Compute

- Serverless SQL Warehouses — Zero management, instant start
- Serverless Jobs — No cluster config for jobs
- Serverless DLT — Auto-managed pipeline compute
- Cost model — Per-second billing, no idle cost
- When to use serverless vs classic clusters

6. Data Mesh on Databricks

- Data Mesh principles — Domain-oriented, self-serve, federated
- Unity Catalog for Data Mesh — Catalog per domain
- Data Products — Define, publish & discover domain data
- Delta Sharing — Cross-domain data sharing without copy
- Data contracts — Schema & quality agreements between domains

7. Databricks AI/BI & GENIE

- AI/BI Dashboards — Natural language + visual analytics

- GENIE — Ask data questions in plain English
- AI-generated SQL — Explain, optimize, document queries
- Databricks Assistant — Notebook AI copilot
- Vector Search — Semantic search for AI applications

---

8. Advanced Security & Compliance

- Network isolation — VNet injection, Private Link, IP allowlists
- Customer-Managed Keys (CMK) — Encrypt Databricks storage
- Compliance — SOC2 Type II, HIPAA, PCI-DSS, ISO 27001
- SCIM — Auto-provision users from Azure AD / Okta
- Audit log analysis — Detect anomalous access patterns

---

9. Databricks for Streaming at Scale

- Kafka → Databricks — Full production setup
- Multiple concurrent streams — Resource isolation
- RocksDB state backend — Large stateful aggregations
- Async checkpointing — Reduce checkpoint overhead
- Delta Live Tables Continuous mode — Sub-second latency

---

10. Platform Engineering on Databricks

- Terraform provider for Databricks — Full IaC
- Databricks Terraform modules — Reusable workspace setup
- Cost governance — Cluster policies, budgets, tagging
- FinOps on Databricks — DBU usage analysis by team/project
- Self-service developer platform — Approved cluster templates

INTERVIEW PREPARATION & CAREER GUIDE

Top questions · Resume tips · Salary guide · Certifications roadmap

Top Technical Interview Questions

'Explain the difference between Delta Live Tables and Databricks Workflows'

'How do you implement SCD Type 2 in Delta Lake using MERGE?'

'When would you use Auto Loader vs COPY INTO?'

'What is the difference between OPTIMIZE + Z-ORDER and Liquid Clustering?'

'How does Unity Catalog row-level security work?'

'Design a real-time clickstream pipeline on Databricks'

'How do you handle schema evolution in Delta Lake?'

'Explain how DLT expectations work and what happens on violation'

'How would you debug a slow Structured Streaming job?'

'What is Change Data Feed and how does it enable incremental processing?'

Resume Bullet Upgrades

■ Built Databricks pipelines

■ Engineered DLT Medallion pipeline processing 8TB/day (Bronze→Silver→Gold) with 99.98% data quality via DLT expectations

■ Used Delta Lake

■ Implemented SCD Type 2 customer history pipeline using Delta MERGE + CDF, reducing reporting latency from 8h to 18min

■ Set up Unity Catalog

■ Designed Unity Catalog governance framework across 12 workspaces — column masking for 6 PII fields, row-level security for 3 data domains

Certifications Roadmap

Month 1-2: Databricks Certified Data Engineer Associate (top priority)

Month 3-4: Databricks Certified Data Engineer Professional

Month 5-6: AWS Certified Data Engineer Associate (if AWS stack)

Month 5-6: DP-203 Azure Data Engineer (if Azure stack)

Month 7+: Databricks Certified ML Professional (career expansion)

Salary Expectations

India — Entry (0-2 yrs): ■12-18 LPA | Mid (2-5 yrs): ■20-35 LPA | Senior (5+): ■35-55 LPA

US: $115K-175K (Mid-level) | $175K-220K+ (Staff/Principal)

UK: £75K-130K | EU: €80K-130K

Canada: CAD $105K-155K

Remote (India-based): 1.5-2x Indian market for global companies

SKILLS PORTFOLIO & COURSE OUTCOMES

MUST-HAVE SKILLS	DIFFERENTIATING SKILLS	TARGET JOB TITLES
■ PySpark — DataFrame API, UDFs ■ Spark SQL — Window, CTEs, JSON ■ Delta Lake — CRUD, MERGE, Time Travel ■ Auto Loader — Incremental ingestion ■ DLT — Declarative pipeline authoring ■ Unity Catalog — Governance ■ Databricks SQL — Dashboards, alerts ■ Databricks Workflows — Orchestration	■ Delta Live Tables APPLY CHANGES (CDC) ■ Liquid Clustering — Auto-optimise ■ Databricks Asset Bundles (DAB) ■ Lakehouse Monitoring ■ Delta Sharing — Cross-org sharing ■ dbt on Databricks ■ AWS/Azure cloud storage integration ■ Photon engine optimisation	★ Databricks Data Engineer ★ Senior Data Engineer ★ Analytics Engineer ★ Lakehouse Architect ★ Data Platform Engineer ★ Cloud Data Engineer ★ MLOps Engineer (advanced path) ★ Data Architect (7+ years)

PROJECT PORTFOLIO CHECKLIST

- ■ Full Medallion Pipeline on Databricks (Bronze → Silver → Gold with DLT)
- ■ Real-time Structured Streaming pipeline (Kafka/Event Hubs → Delta Lake)
- ■ SCD Type 2 CDC pipeline using Delta MERGE or DLT APPLY CHANGES
- ■ Unity Catalog setup with PII masking, row-level security & data lineage
- ■ Databricks Workflow with multi-task orchestration (DLT + Job + SQL refresh)
- ■ CI/CD pipeline using GitHub Actions + Databricks Asset Bundles
- ■ Databricks SQL AI/BI Dashboard for business stakeholders
- ■ AWS or Azure integration (S3/ADLS + cloud secrets + Unity Catalog external location)

COURSE SUMMARY & ENROLLMENT

COURSE STATS	WHAT YOU WILL MASTER	ENROLLMENT
Duration: 120 Hours 19 Core Modules 10 Bonus Topics 35+ Hands-On Labs 3 Graded Capstone Projects 60 Interview Q&A Lifetime Materials Access Job Placement Support	✓ Build production DLT Medallion pipelines ✓ Master Unity Catalog governance ✓ Implement real-time streaming ✓ Deploy CI/CD with DAB ✓ Integrate AWS & Azure ✓ Pass Databricks certification ✓ Build impressive portfolio ✓ Land top data engineering roles	databrickstraining.in info@databrickstraining.in +91-8500002025 ★★★★★ 5.0 Rating Trusted by 5000+ Data Engineers Batch Starts Monthly