

DATA ENGINEERING

INTERMEDIATE

2 DAYS

Power BI and Databricks Integration

Connect Power BI to Databricks SQL warehouses, model lakehouse Delta data, and deliver optimised DirectQuery or Import reports with Unity Catalog-aligned governance.

Why Power BI + Databricks?

BI on the lakehouse

Query governed Delta tables through Databricks SQL warehouses instead of brittle extract pipelines.

Clear SQL Server migration path

Understand how gateway, schema, security, and refresh patterns change on Databricks.

Right connectivity mode

Choose DirectQuery, Import, or composite models based on freshness, scale, and cost.

Governed self-service

Align Power BI RLS with Unity Catalog policies for consistent access control.

What you'll be able to do

- ✓ Connect Power BI to Azure Databricks using Partner Connect and manual SQL warehouse endpoints
- ✓ Understand the difference between DirectQuery and Import mode when connecting to Databricks and choose appropriately
- ✓ Build data models in Power BI optimised for lakehouse-sourced Delta tables
- ✓ Understand the key architectural and connectivity differences between Power BI on SQL Server versus Databricks
- ✓ Apply performance optimisation techniques for Power BI reports on Databricks data
- ✓ Develop production-ready dashboards and reports against Databricks data
- ✓ Implement row-level security in Power BI aligned with Unity Catalog governance

Who this is for

- Data Engineers
- Data Analysts
- Analytics Engineers
- BI Professionals
- Cloud Engineers
- Data Platform Teams

Curriculum

01 Architecture Overview: Power BI on the Databricks Lakehouse

- How Power BI and Databricks connect for lakehouse analytics
- Why SQL warehouses are preferred over all-purpose clusters for BI workloads
- Photon acceleration, result caching, and BI-oriented query planning
- Unity Catalog three-level namespace: catalog.schema.table discovery in Power BI
- Where Gold-layer Delta tables fit in a Power BI semantic model

02 Connectivity Setup: Partner Connect, SQL Warehouse, and Gateway

- Databricks Partner Connect: generate a Power BI connection package quickly
- Manual connection using server hostname, SQL warehouse HTTP path, and authentication
- Personal access token versus Microsoft Entra ID authentication patterns
- ODBC/JDBC and on-premises data gateway scenarios for hybrid environments
- **Hands-on:** Connect Power BI Desktop to a Databricks SQL warehouse and browse lakehouse tables

03 DirectQuery, Import, and Composite Models

- DirectQuery: live queries to Databricks at visual interaction time
- Import mode: load into the Power BI model, scheduled refresh, and size considerations
- Composite models: combine Databricks DirectQuery facts with imported dimensions
- Decision criteria: data freshness, dataset size, concurrency, and warehouse cost
- Common anti-patterns that overdrive SQL warehouse compute from Power BI

04 SQL Server vs Databricks for Power BI Developers

- Connectivity: on-premises gateway for SQL Server versus SQL warehouse endpoints for Databricks
- Data format: relational star schema versus Delta Lake tables in a lakehouse catalog
- Query handling: SQL Server engine versus Photon-accelerated Databricks SQL
- Security model: SQL logins/Windows AD versus Unity Catalog with Entra ID
- Refresh patterns: gateway scheduled refresh versus DirectQuery or warehouse-backed freshness

05 Data Modelling for Gold-Layer Delta Tables

- Modelling principles for lakehouse-sourced Gold tables and pre-aggregated datasets
- Wide tables versus dimensional models when sourcing from Databricks
- Relationships, cardinality, and cross-filter direction on Delta-backed tables
- When to model in Power BI versus pushing logic into Databricks Gold tables
- **Hands-on:** Build an initial Power BI data model against Databricks Gold Delta tables

06 DAX and Query Folding on Databricks Sources

- DAX fundamentals for Databricks-connected models: measures, calculated columns, context
- Common patterns: time intelligence, running totals, and period-over-period comparisons
- Query folding with DirectQuery: which Power Query steps fold to Databricks SQL
- Transformations that break folding and how they affect warehouse load
- Designing measures that stay efficient under DirectQuery constraints

07 Performance Optimisation for Databricks-Backed Reports

- Power BI Performance Analyser: slow visuals, heavy DAX, and DirectQuery bottlenecks
- Aggregation tables: pre-build summaries in Databricks Gold and reference them in Power BI
- SQL warehouse sizing, concurrency, and auto-stop settings for cost and latency
- Reducing DirectQuery latency with result caching, lean models, and SQL tuning
- Practical checklist for production-ready report performance on Databricks

08 Governance, RLS, and Publishing to Power BI Service

- Static RLS with roles and dynamic RLS using `USERNAME()` or `USERPRINCIPALNAME()`
- Aligning Power BI RLS with Unity Catalog access policies for consistent governance
- Publishing to Power BI Service: workspaces, datasets, and Databricks connection settings
- Scheduled refresh versus DirectQuery in the service: operational trade-offs
- Dashboard design principles for analytical usability on lakehouse data

09 Capstone: Production Dashboard from a Databricks Gold Table

- Connect to a governed Gold Delta table through a SQL warehouse
- Model relationships and measures for a production-style semantic model
- Apply at least one performance optimisation (aggregations, lean visuals, or warehouse tuning)
- Configure RLS aligned to a simple governance scenario
- **Hands-on:** Publish an end-to-end analytics dashboard sourced from Databricks Gold data