

Advanced Power BI Analytics with Azure Databricks and Tabular Editor

Course Duration: 40 Hours (5 Days)

Overview

The Advanced Power BI Analytics with Azure Databricks and Tabular Editor course is designed for professionals who already understand Power BI at the PL-300 level and want to extend their analytics capabilities into an enterprise Azure data platform. The course follows the client's target-state architecture, covering data ingestion through Azure Data Factory, ADLS Bronze, Silver and Gold layers, Azure Databricks processing, Databricks SQL access, and governed information delivery through Power BI.

Participants will learn how to connect Power BI semantic models to curated Databricks data, choose suitable storage modes, design complex enterprise models, create advanced and reusable DAX calculations, and improve performance across both Power BI and Databricks. The course also includes extensive use of Tabular Editor for measure management, calculation groups, dynamic format strings, Best Practice Analyzer, scripting, metadata governance and deployment. By the end of the course, learners will be able to build, optimise, govern and deploy a production-oriented Power BI-on-Databricks analytics solution.

Audience Profile

This course is intended for Power BI professionals who have completed PL-300 training or possess equivalent hands-on experience and now need to integrate Power BI with Azure Databricks and enterprise data lake architecture.

Course Syllabus

Learning Path 1 (Day 1): Target Architecture, Databricks Fundamentals and Power BI Connectivity

- Understand the complete target-state architecture from D365 and secondary sources to Power BI information delivery
- Identify the responsibilities of data engineers, semantic model developers and report developers
- Understand Azure Databricks workspace components and their role in analytics solutions
- Work with Unity Catalog concepts: catalogs, schemas, tables, views and permissions
- Understand Delta Lake fundamentals and analytics-ready Delta tables
- Compare Databricks notebooks, clusters and Databricks SQL warehouses
- Understand serverless, pro and classic SQL warehouse concepts
- Understand the Bronze, Silver and Gold layers of the medallion architecture
- Determine when Power BI should consume Silver-layer or Gold-layer data
- Identify transformations that belong in Databricks, Power Query or DAX
- Connect Power BI Desktop to Azure Databricks using the Azure Databricks connector
- Configure the server hostname, HTTP path and authentication method
- Understand Microsoft Entra ID, OAuth and service-principal connectivity concepts
- Configure environment parameters for development, testing and production
- Compare Import and DirectQuery connectivity to Databricks SQL
- Understand the dataset-centric semantic model architecture
- Create shared semantic models and live-connected thin reports
- Understand Build permission, promotion, certification and controlled self-service reporting
- Lab 0: Validate the Power BI, Databricks and Azure training environment
- Lab 1: Connect Power BI to Databricks Gold-layer tables and publish a shared semantic model
- Lab 2: Create a thin Power BI report using a live connection to the shared semantic model

Learning Path 2 (Day 2): Enterprise Semantic Modelling and Storage-Mode

Design

- Define business processes, analytical grain and dimensional modelling requirements
- Design fact tables and dimension tables for enterprise analytics
- Understand transaction, periodic snapshot and accumulating snapshot fact tables
- Design conformed dimensions across multiple business processes
- Work with surrogate keys, natural keys and date dimensions
- Implement role-playing dimensions and inactive relationships
- Understand slowly changing dimensions and their impact on Power BI models
- Work with degenerate dimensions, bridge tables and many-to-many relationships
- Design models for parent-child hierarchies and disconnected parameter tables
- Understand one-to-many, one-to-one and limited relationships
- Control filter direction and avoid ambiguous relationship paths
- Use USERELATIONSHIP for alternate date and business relationships
- Compare Import, DirectQuery, Dual and composite storage modes
- Design table-level storage modes for large Databricks-backed models
- Use Dual dimensions with DirectQuery fact tables where appropriate
- Evaluate freshness, concurrency, latency and cost trade-offs
- Understand Direct Lake as a Microsoft Fabric comparison and possible future-state option
- Apply a Silver-versus-Gold sourcing decision framework
- Apply semantic model naming standards and business-friendly metadata
- Configure display folders, hierarchies, sort-by columns and data categories
- Hide technical columns and create dedicated measure tables
- Lab 3: Convert curated Databricks tables into an enterprise star schema
- Lab 4: Implement advanced relationships, role-playing dimensions and bridge tables
- Lab 5: Prepare a per-table storage-mode and source-layer decision matrix

Learning Path 3 (Day 3): Advanced DAX and Tabular Editor Model

Development

- Review filter context, row context and context transition in advanced DAX
- Understand the evaluation behaviour of CALCULATE and filter modifiers
- Use variables to improve DAX readability and performance
- Build reusable calculations through measure branching
- Use SUMX, FILTER, ADDCOLUMNS, SELECTCOLUMNS and SUMMARIZECOLUMNS
- Create ratio, percentage, ranking and dynamic segmentation measures
- Use virtual tables and disconnected tables for dynamic analysis
- Implement allocation patterns and correct total calculations
- Create semi-additive measures for inventory, balance and snapshot scenarios
- Implement standard, fiscal and custom time-intelligence calculations
- Create year-to-date, rolling-period, trailing-twelve-month and prior-period measures
- Handle incomplete-period comparisons and working-day calculations
- Optimise DAX by reducing unnecessary iterators and repeated expressions
- Launch Tabular Editor as an external tool from Power BI Desktop
- Navigate tables, columns, measures, hierarchies and model properties in Tabular Editor
- Create and organise measures using display folders
- Bulk-edit measure format strings, descriptions and metadata
- Create perspectives, translations and model annotations
- Create calculation groups and calculation items
- Use SELECTEDMEASURE and dynamic format strings
- Manage calculation-group precedence and avoid redundant time measures
- Use Best Practice Analyzer to evaluate semantic model quality

- Create organisation-specific Best Practice Analyzer rules
- Use C# scripting for repetitive semantic model operations
- Compare key capabilities of Tabular Editor 2 and Tabular Editor 3
- Lab 6: Build an advanced DAX measure library for the Databricks-backed semantic model
- Lab 7: Create a fiscal time-intelligence calculation group with dynamic formatting
- Lab 8: Apply bulk metadata changes, scripts and Best Practice Analyzer fixes

Learning Path 4 (Day 4): Performance, Refresh, Security and Cost Optimisation

- Understand the complete query path from a Power BI visual to Databricks SQL
- Identify report, DAX, semantic model, network, SQL warehouse and Delta-table bottlenecks
- Understand VertiPaq storage, compression, encoding and column cardinality
- Reduce model size by removing unused columns and selecting efficient data types
- Optimise high-cardinality identifiers, decimal precision and calculated columns
- Use Power BI Performance Analyzer to identify slow visuals and DAX queries
- Use DAX Studio Server Timings and query plans for detailed diagnostics
- Compare formula-engine and storage-engine workload
- Benchmark measure performance before and after optimisation
- Understand Databricks SQL warehouse sizing, concurrency and auto-stop behaviour
- Use Databricks query history and query profiling for Power BI workloads
- Understand result caching, disk caching and data-scanning considerations
- Design efficient Gold-layer tables and views for Power BI consumption
- Understand predicate pushdown and efficient DirectQuery filtering
- Recognise inefficient Power BI-generated SQL and model patterns
- Understand scheduled, on-demand and incremental refresh
- Configure RangeStart and RangeEnd parameters for incremental refresh
- Understand partitions, refresh policies and large semantic model considerations
- Compare incremental refresh, hybrid tables and near-real-time patterns
- Manage selected partitions and refresh policies through Tabular Editor and XMLA
- Understand Unity Catalog permissions and least-privilege access
- Configure Power BI workspace roles, Build permission and controlled sharing
- Implement row-level security and object-level security
- Understand service principals, credentials and Azure Key Vault considerations
- Prepare security, refresh and cost responsibility matrices
- Lab 9: Diagnose and optimise a slow semantic model and DAX measure
- Lab 10: Compare Import and DirectQuery behaviour against Databricks SQL
- Lab 11: Configure incremental refresh and implement a security role

Learning Path 5 (Day 5): Governance, SDLC, Deployment and End-to-End Workshop

- Design separate Development, Test and Production Power BI workspaces
- Parameterise Databricks workspace, SQL warehouse, catalog and schema values
- Understand Power BI deployment pipelines and deployment rules
- Define testing, approval, rollback and post-deployment validation responsibilities
- Save Power BI solutions in Power BI Project (PBIP) format
- Understand report and semantic model project folders
- Understand Tabular Model Definition Language (TMDL) files
- Store semantic model and report definitions in Git
- Use Azure DevOps repositories, branches, commits and pull requests
- Review model changes and reduce conflicting edits
- Plan source-control conventions for collaborative semantic model development
- Use Tabular Editor Best Practice Analyzer in automated quality checks
- Validate naming, metadata and model-quality standards during CI processes

- Understand command-line validation and release quality gates
- Connect Tabular Editor to published semantic models through the XMLA endpoint
- Understand XMLA read-only and read-write operations
- Compare metadata-only deployment, model comparison and full deployment
- Safely deploy semantic model changes and refresh selected partitions
- Use promoted and certified semantic models for governed reuse
- Use lineage and impact analysis before model changes
- Define semantic model owners, KPI owners and refresh responsibilities
- Align model metadata with a business glossary and data-governance standards
- Generate technical and business documentation from TMDL and Tabular Editor metadata
- Use AI-assisted documentation with human review and confidentiality controls
- Workshop: Build an end-to-end Databricks-to-Power BI analytics solution
- Workshop: Implement advanced DAX, a calculation group, metadata and model-quality rules
- Workshop: Validate performance, security, refresh and Databricks SQL behaviour
- Workshop: Prepare PBIP, Git, deployment and governance deliverables
- Workshop: Present the solution architecture and production-handover recommendations

Note: The course is designed as an advanced, hands-on programme. Foundational PL-300 topics are reviewed only where required for integration, modelling, performance or governance scenarios.