

Databricks & Fabric Data Engineer Training

100% Practical & Project-Based

This Hands-On training program is divided into **FOUR Comprehensive Modules**:

Module 1: SQL Server (MSSQL) & T-SQL

Module 2: Databricks Data Engineering – Spark, Python & PySpark

Module 3: Fabric Data Engineering – Data Factory, One Lake, KQL, PySpark

Module 4: Real-Time Projects

Duration: 12 Weeks

Prerequisites: None. The course starts from basics, progresses to **advanced** Databricks concepts!

Trainer: Mr. Sai Phanindra

Trainer Profile: [linkedin.com/in/saiphanindra/](https://www.linkedin.com/in/saiphanindra/)

Learning Outcomes

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

- Design enterprise-grade Lakehouse architectures
- Build scalable ETL pipelines using Databricks
- Implement Bronze, Silver, and Gold data layers
- Perform advanced data transformations using PySpark
- Secure data using Unity Catalog
- Deliver business-ready datasets for analytics
- Build executive dashboards in Power BI
- Implement OneLake with Data Factory
- Implement Fabric Mirroring and DWH
- Implement Fabric PySpark ETL & ELT
- Gain hands-on experience through production-style Data Engineering projects

Detailed Course Content

Module 1: MSSQL & TSQL

Ch 1: SQL Database Job Roles

- ✓ Database Intro
- ✓ OLTP, DWH, OLAP
- ✓ DBMS Basics
- ✓ Data Engineer Job Roles

Ch 2: Database Intro & Installations

- ✓ SQL Server Installations

- ✓ Instance & Collations
- ✓ SSMS Tool Installation
- ✓ Connections, Authentications

Ch 3: SQL Basics V1 (Commands)

- ✓ SQL Basics (DDL, DML, etc..)
- ✓ Creating Databases, Tables
- ✓ Data Inserts (GUI, SQL)
- ✓ Basic SELECT Queries

Ch 4: SQL Basics V2 (Commands, Operators)

- ✓ DDL: Create, Alter, Drop
- ✓ DML: Insert, Update, Delete
- ✓ DQL: Select, Fetch
- ✓ Add, Truncate Statements
- ✓ SQL Operators

Ch 5: Data Types & Variables

- ✓ Integer Data Types
- ✓ Character, MAX Data Types
- ✓ Decimal & Boolean Data Types
- ✓ Date and Time Data Types
- ✓ SQL_Variant Type

Ch 6: Data Imports

- ✓ Data Imports with Excel
- ✓ Data Imports with CSV
- ✓ Data Types Detection
- ✓ OLE-DB Connections
- ✓ ORDER BY, TOP, OFFSET

Ch 7: Schemas & Batches

- ✓ Schemas & Table Grouping
- ✓ Real-world Banking Database
- ✓ 2 Part, 3 Part & 4 Part Naming
- ✓ Batch Concept & “Go” Command

Ch 8: Constraints, Keys & RDBMS

- ✓ Null, Not Null Constraints
- ✓ Unique Key & Check
- ✓ Primary Key Constraint
- ✓ Foreign Keys, Default
- ✓ DB Diagrams & ER Models

Ch 9: Normal Forms & RDBMS

- ✓ Normal Forms: 1 NF, 2 NF
- ✓ 3 NF, BCNF and 4 NF

- ✓ 1:1, 1:M, M:1 Cardinality
- ✓ Cascading Keys, Self Referencing Keys

Ch 10: Joins & Queries

- ✓ Joins: Table Comparisons
- ✓ Inner Joins & Matching Data
- ✓ Outer Joins: LEFT, RIGHT
- ✓ Full Outer Joins & Aliases
- ✓ Self Joins & Aliases

Ch 11: Sub Queries

- ✓ Basic Sub Queries
- ✓ Aggregations
- ✓ Combining Queries
- ✓ Correlated Sub Queries
- ✓ UNION, UNION ALL

Ch 12: Group By Queries

- ✓ Group By, Distinct
- ✓ GROUP BY, HAVING
- ✓ Cube() and Rollup()
- ✓ Sub Totals & Grand Totals
- ✓ Grouping() & Usage
- ✓ ISNULL, COALESCE

Ch 13: Joins with Group By, Sub Queries

- ✓ 3 Table, 4 Table Joins
- ✓ Join Queries & WHERE
- ✓ Join & Group By, Sub Queries
- ✓ IIF(), CASE Statement
- ✓ EXISTS, NOT EXISTS
- ✓ Query Execution Order

Ch 14: Views & RLS

- ✓ Views: Realtime Usage
- ✓ DML, SELECT with Views
- ✓ WITH CHECK OPTION
- ✓ Row Level Security (RLS)
- ✓ Important System Views

Ch 15: Functions & Queries

- ✓ User Defined Functions
- ✓ Scalar, Table Value Functions
- ✓ Variables & Parameters
- ✓ Date & Time Functions
- ✓ String Functions
- ✓ Aggregated Functions

- ✓ Data Conversion Functions

Ch 16: Advanced SQL & Window Functions

- ✓ Window Functions (Rank)
- ✓ Row_Number, DenseRank
- ✓ Partition By & Order By
- ✓ Lag & Lead Functions
- ✓ Pivot, UnPivot
- ✓ Running Totals
- ✓ Moving Average

Ch 17: Stored Procedures - 1

- ✓ Stored Procedures: Realtime Use
- ✓ Parameters Concept with SPs
- ✓ Procedures with SELECT
- ✓ System Stored Procedures
- ✓ Stored Procedures, Tuning

Ch 18: Stored Procedures - 2

- ✓ Merge Statement (Upsert)
- ✓ Merge with OLTP & DWH
- ✓ Matched and Not Matched
- ✓ Merge & SP Recompilations

Ch 19: Triggers & Automations

- ✓ Need for Triggers in Real-world
- ✓ DDL & DML Triggers
- ✓ For / After Triggers
- ✓ Instead Of Triggers
- ✓ Disabling DMLs Triggers

Ch 20: Transactions & ACID

- ✓ Auto Commit Transaction
- ✓ Explicit Transactions
- ✓ COMMIT, ROLLBACK
- ✓ Checkpoint & Query Blocking
- ✓ READPAST, LOCKHINT
- ✓ TRY...CATCH & Error Handling

Ch 21: SQL Query Performance & Indexing

- ✓ Clustered vs Nonclustered Indexes
- ✓ Index Seek vs Scan
- ✓ Execution Plans
- ✓ SARGable Queries
- ✓ Tuning Queries
- ✓ Avoiding SELECT *
- ✓ Indexing JOIN/WHERE columns

- ✓ Statistics Basics
- ✓ Query Optimization Examples

Ch 22: CTEs & Tuning

- ✓ Common Table Expression
- ✓ CTEs for Data Retrieval
- ✓ CTEs for DML Operations
- ✓ Data Cleansing Techniques
- ✓ Duplicate Detection/Removal

Ch 23: Temp Tables & Query Techniques

- ✓ Local & Global Temp Tables
- ✓ SELECT..INTO Statement
- ✓ Cursor Basics – When to Use & Avoid
- ✓ NULLIF
- ✓ Basic Execution-Plan Reading

Ch 24: Bonus / Advanced Module: SQL Server Internals

- ✓ Database Engine Components
- ✓ Parser, Compiler & Optimizer
- ✓ Parsing and Compilation
- ✓ Memory Manager & IO Managers

Real-Time Project on HealthCare Domain

Project Requirement:

Solve 20+ Real-World Healthcare Business Requirements.

Project Workflow:

Database Design → SQL Development → Data Analysis → Performance Optimization

Project Operational Flow:

Healthcare DB → Patients → Doctors → Appointments → Treatments → Billing → Insurance → SQL Analysis → Stored Procedures → Performance Tuning → Business Reports

Business Requirements:

Monthly Hospital Revenue, Top Doctors By Patients Treated, Repeat Patients, Department Performance, Average Treatment Cost, Unpaid Bills, Insurance Claim Analysis, Patient Trends, Month-Over-Month Revenue And Top Procedures...

Module 2: Databricks (Spark, PySpark, Big Data, Genie AI)

Phase 1: Databricks Fundamentals, Architecture & Unity Catalog

Ch 1: Databricks Introduction

- ✓ Cloud ETL, DWH
- ✓ Cloud Computing
- ✓ Databricks Concepts
- ✓ Databricks Account

- ✓ Big Data in Cloud

Ch 2: Databricks Architecture

- ✓ Databricks Runtime (DBR)
- ✓ RDD & DAG
- ✓ Databricks Lakehouse Architecture
- ✓ Spark Compute Concepts
- ✓ Workers & Drivers
- ✓ Databricks Framework
- ✓ Databricks APIs

Ch 3: Unity Catalog

- ✓ Unity Catalog Concepts
- ✓ Metastore
- ✓ Catalogs & Schemas
- ✓ Managed & External Tables
- ✓ Storage Credentials
- ✓ External Locations
- ✓ Volumes
- ✓ GRANT / REVOKE
- ✓ Data Permissions
- ✓ Lineage, Data Governance
- ✓ Databricks Workspace UI
- ✓ Spark Table Creations

Phase 2: Spark SQL, Python ETL & PySpark

Ch 4: Spark SQL: Basics

- ✓ Spark SQL Notebooks
- ✓ Creating Catalog
- ✓ Creating Schemas
- ✓ Creating Tables
- ✓ Spark Data Types
- ✓ PySpark API: SQL Queries
- ✓ Dropping Objects
- ✓ Notebooks: Exports, Clone

Ch 5: Spark SQL: Joins & Table Types

- ✓ Delta Tables
- ✓ Managed Tables
- ✓ Data Partitioning
- ✓ Union, Views in Spark
- ✓ Spark SQL Joins
- ✓ Aggregations
- ✓ Data Analytics

Ch 6: Spark SQL: Functions

- ✓ Math, Sort Functions

- ✓ String, DateTime Functions
- ✓ Conditional Statements
- ✓ SQL Expressions with `expr()`
- ✓ Volume for our Data Assets
- ✓ File Formats, Schema Inference
- ✓ Spark SQL Aggregations

Ch 7: Spark SQL: Time Travel

- ✓ Time Travel Concepts
- ✓ Spark DB: Logical Architecture
- ✓ Spark DB: Physical Store
- ✓ Data File & Log File Store
- ✓ Time Travel
- ✓ DESCRIBE, EXTENDED
- ✓ HISTORY
- ✓ Version Numbers

Ch 8: Python: Introduction, Print

- ✓ Python Introduction
- ✓ Python Versions
- ✓ Python Implementations
- ✓ Python in Spark (PySpark)
- ✓ Python `Print()`
- ✓ Single, Multiline Statements

Ch 9: Python: Variables

- ✓ Python Variables
- ✓ Variable Declarations
- ✓ Variable Values
- ✓ Multi Variable Values
- ✓ Common Variable Values
- ✓ Realtime use of Variables

Ch 10: Python: Operators

- ✓ Need for Operators
- ✓ Arithmetic Operators
- ✓ Assignment Operators
- ✓ Comparison Operators
- ✓ Operator Precedence
- ✓ Operands in Python

Ch 11: Python: Control Statements

- ✓ Python Control Structures
- ✓ If ... Else Statement
- ✓ Short Hand If
- ✓ ELIF & ELSE IF Statements
- ✓ OR, AND Concepts

- ✓ Python Loops

Ch 12: Python: Data Types

- ✓ Python Data Types
- ✓ Integer / Int Data Types
- ✓ Float, String Data Types
- ✓ List Data Type
- ✓ Dictionary Data Type
- ✓ Tuple Data Type

Ch 13: Python: Modules & DataFrames

- ✓ Python Modules
- ✓ Pandas, NumPy
- ✓ DataFrame Concepts
- ✓ Handling Nulls
- ✓ Data Cleansing Concepts
- ✓ Python Functions

Phase 3: Medallion Architecture, Delta Lake, Auto Loader, Lakeflow & Airflow

Ch 14: PySpark DataFrames & Transformations

- ✓ Creating PySpark DataFrames
- ✓ Reading Data Sources
- ✓ select(), filter(), withColumn()
- ✓ Handling NULL Values
- ✓ Data Type Casting
- ✓ Joins, Aggregations
- ✓ Window Functions
- ✓ Writing DataFrames

Ch 15: Medallion Architecture - 1

- ✓ Medallion Architecture
- ✓ Aggregated Data Loads
- ✓ Bronze, Silver and Gold
- ✓ Temp Views
- ✓ Spark Tables (Parquet)
- ✓ Work with File Sources

Ch 16: Medallion Architecture - 2

- ✓ Medallion Architecture
- ✓ Azure SQL DB Connections
- ✓ Joining Source Tables
- ✓ Data Frames, Temp Views
- ✓ Aggregated Data Loads
- ✓ Gold Data Consumption
- ✓ Raw Sources → Bronze → Silver → Gold → BI/Analytics

Ch 17: Delta Lake

- ✓ Databricks Delta Lake
- ✓ Schema Evolution
- ✓ Azure SQL DB Connections
- ✓ DataFrames, Temp Views
- ✓ Delta Table API
- ✓ MERGE / Upsert Operations
- ✓ Version History & Data Retention
- ✓ Delta Transaction Log

Ch 18: PySpark: Widgets

- ✓ PySpark Parameters
- ✓ Text Widgets
- ✓ User Parameters
- ✓ Manual Executions
- ✓ Automations
- ✓ UI & JSON For Widgets

Ch 19: Lake Flow Jobs

- ✓ Workflows & CRON
- ✓ Job Compute, Running Tasks
- ✓ Python Script Tasks
- ✓ Parameters into Notebook Tasks
- ✓ Parameters into Python Script Tasks
- ✓ Concurrent Executions, Dependencies
- ✓ Branching Control

Ch 20: PySpark: Auto Loader

- ✓ Cloud Files Architecture
- ✓ Checkpoint Configurations
- ✓ Schema Location
- ✓ Checkpoint Location
- ✓ Data Stream
- ✓ Schema Evaluation Modes
- ✓ Incremental File Ingestion
- ✓ Rescued Data Column

Ch 21: LakeFlow Declarative Pipelines

- ✓ SDP: Spark Declarative Pipelines
- ✓ Delta LIVE Tables
- ✓ Streaming Data Loads
- ✓ Bronze, Silver, Gold Data
- ✓ Materialized Views
- ✓ Pipeline Clusters
- ✓ Data Quality Checks

Ch 22: Databricks Optimizations

- ✓ Lazy Evaluation

- ✓ Explain Plan
- ✓ Caching, Data Shuffling
- ✓ Broadcast Joins
- ✓ Partitions, Data Skew
- ✓ Liquid Clustering
- ✓ VACUUM
- ✓ OPTIMIZE, Z Ordering

Ch 23: Databricks Security, AI

- ✓ ACL: Access Control List
- ✓ Workspace Access Control
- ✓ Catalog Security
- ✓ Schema, Volume Security
- ✓ Table Security
- ✓ Column Security

Ch 24: Databricks with Power BI

- ✓ Workspace Settings
- ✓ Access Keys (Tokens)
- ✓ Server Host
- ✓ HTTP Path
- ✓ Spark Connectors
- ✓ Databricks Connectors

Ch 25: Genie AI

- ✓ Genie AI Concepts
- ✓ Unity Catalog
- ✓ AI Components in Databricks
- ✓ Debugging Controls
- ✓ Using AI for Notebook Design

Ch 26: GitHub & Asset Bundle

- ✓ GitHub Concepts
- ✓ GIT: Main, Branches
- ✓ Asset Bundle
- ✓ Integrating GIT with Asset Bundle
- ✓ Process Integration

Ch 27: Apache Airflow

- ✓ Apache Airflow Concepts
- ✓ Airflow Operators & DAG
- ✓ Python Scripts For Airflow
- ✓ DatabricksRunNowOperator
- ✓ DatabricksSubmitRunOperator
- ✓ Airflow Jobs

Module 3: Fabric Data Engineering

Part 1: Fabric Concepts, DWH & Fabric Data Factory

Ch 1: Fabric Introduction

- ✓ Need for Fabric, Big Data
- ✓ Fabric Data Engineering Model
- ✓ Fabric Components (Items)
- ✓ Microsoft Fabric: Advantages
- ✓ Cloud Warehouse & AI
- ✓ AI with CoPilot
- ✓ Azure Versus Fabric DWH

Ch 2: Fabric Account, Workspace

- ✓ Need for Fabric Workspace
- ✓ Workspace Creation Process
- ✓ ETL, Storage, Analytical
- ✓ Streaming, Monitoring
- ✓ Compute & Separation

Ch 3: Fabric OneLake Architecture

- ✓ Intelligent Data Foundation
- ✓ Polaris Distributed Engine
- ✓ Stateless & Stateful
- ✓ Cache, Metadata, Xact & Data
- ✓ Fabric Tasks, Inputs & DAG
- ✓ State Machine & Statistics
- ✓ Hot Spot Recovery

Ch 4: Fabric Warehouse

- ✓ Fabric Warehouse Creation
- ✓ Fabric Warehouse Features
- ✓ Fabric Warehouse Properties
- ✓ Fabric Warehouse Limitations
- ✓ DWH Internal Operations
- ✓ Default Schemas & Objects
- ✓ SSMS Connections

Ch 5: Fabric Data Types

- ✓ Realtime use of Fabric Houses
- ✓ Exact, Approximate Numbers
- ✓ Date and Time Data Types
- ✓ Fixed & Variable Length

- ✓ Binary & String Data Types
- ✓ Fabric Type Limitations
- ✓ Save As table, Save As View

Ch 6: Fabric Caching

- ✓ Fabric Caching Process
- ✓ In-memory Cache, Disk Cache
- ✓ Cache Types: LRU /MRU
- ✓ Cold Cache / Cold Run
- ✓ Realtime use of Caching
- ✓ Performance Advantages
- ✓ Warehouse Optimizations

Ch 7: Fabric Statistics

- ✓ Query Engine Options
- ✓ Statistics Types
- ✓ Leverage Statistics
- ✓ Auto, Manual Statistics
- ✓ Update Statistics
- ✓ Statistics Consistency
- ✓ Statistics Lists & Reports

Ch 8: Time Travel

- ✓ Continuous Data Protection
- ✓ Data Storage, Retention
- ✓ FOR TIMESTAMP AS OF
- ✓ Time Travel Scenarios
- ✓ Time Travel Implementation
- ✓ Time Travel on Queries
- ✓ Time Travel Limitations

Ch 9: Zero Copy Cloning

- ✓ User Layer, Storage Layer
- ✓ Cloning & Parquet Files
- ✓ Synapse Data Warehouse
- ✓ Data History Retention
- ✓ Point In Time , Schema Level
- ✓ Zero Copy Cloning Limitations

Ch 10: Fabric Security

- ✓ Workspace Security
- ✓ Warehouse Security
- ✓ Item Security & Roles
- ✓ Adding AD Users
- ✓ Item Security Limitations
- ✓ MFA & Client Security

Ch 11: Fabric Copy Job

- ✓ ETL Implementation Options
- ✓ Copy Job Item
- ✓ Data Loads with Copy Job
- ✓ Full Loads
- ✓ Testing Copy Jobs

Ch 12: Fabric Copy Job

- ✓ Incremental Loads with Copy Job
- ✓ Business Key Concept
- ✓ DWH: Data Storage
- ✓ Testing Initial Loads
- ✓ Testing Incremental Loads
- ✓ Copy Job Limitations

Ch 13: Fabric Data Factory

- ✓ Need for Fabric Data Factory
- ✓ ETL Operations in FDF
- ✓ Data Sources, Transformations
- ✓ Activities and Connections
- ✓ Data Destinations (Sinks)
- ✓ Creating Pipelines

Ch 14: Fabric Pipelines Design

- ✓ Creation Options for Pipelines
- ✓ Azure SQL DB Data Loads
- ✓ Creating Data Sets
- ✓ Copy Command Usage
- ✓ Run ID & Monitoring
- ✓ Pipeline Creation, Verification

Ch 15: Data Loads with Azure

- ✓ Azure Data Lake Storage (ADLS)
- ✓ Azure BLOB Containers
- ✓ Fabric Data Loads From Azure Files
- ✓ Fabric Warehouse with Azure
- ✓ Run IDs and Activity
- ✓ Compressions & Advantages

Ch 16: ETL Staging

- ✓ Staging: Advantages
- ✓ Caching & Storing Concept
- ✓ Staging Types in Fabric
- ✓ Workspace & External
- ✓ External Stages in Pipelines
- ✓ Pipeline Trigger, Monitor

Ch 17: Fabric Aggr Data Loads

- ✓ Aggregation Scenarios
- ✓ Creating Views in TSQL
- ✓ Using Views in FDF Pipelines
- ✓ Using Pipeline Editor
- ✓ Data Loads to Warehouse
- ✓ Pipeline Verifications

Ch 18: Fabric Incremental Loads - 1

- ✓ Upsert (Incremental Loads)
- ✓ Business Key Concept
- ✓ SCD: Slowly Changing Dimension
- ✓ Full Loads Vs Incr Loads
- ✓ Testing Incrementing Loads
- ✓ Pipeline Execution Tests

Ch 19: Fabric Incremental Loads - 2

- ✓ Control Tables
- ✓ Watermark Columns
- ✓ Lookup Activity
- ✓ Stored Procedure Activity
- ✓ Parameters & Runs
- ✓ Concurrency & Batch Count

Ch 20: OnPrem Gateways

- ✓ Need for On-Premise Gateway
- ✓ Installing & Configuring
- ✓ Authentication, Usage
- ✓ On-Premise Connections
- ✓ Pipelines for Data Loads
- ✓ Warehouse Data Storage
- ✓ Data Refresh with Gateways

Ch 21: Data Factory Pipeline Tuning

- ✓ Intelligent Throughput
- ✓ DOCP & Optimizations
- ✓ Staging & In-Memory
- ✓ Spark Compute Options
- ✓ Concurrent Connections
- ✓ ETL Partitions in Real-world

Part 2: Fabric Data Flow, Lake House

Ch 1: Fabric Lakehouse

- ✓ Fabric Lakehouse Architecture
- ✓ Files and Tables Storage
- ✓ Direct Lake & AI

- ✓ Creating Lakehouse
- ✓ Azure SQL Database Source
- ✓ UI: Limitations

Ch 2: Lakehouse File Loads

- ✓ Creating Lakehouse
- ✓ Incremental Refresh
- ✓ Computed Tables
- ✓ Reusable Transformations
- ✓ Scheduling
- ✓ Monitoring
- ✓ Best Practices

Ch 3: Power Query Level 1

- ✓ Power Query Concept
- ✓ Fabric Lake House
- ✓ ETL, ELT Process with AI
- ✓ Data Combine: Union, Append
- ✓ Duplicate / Reference Queries
- ✓ Warehouse Data Loads

Ch 4: Power Query Level 2

- ✓ Table Transformations
- ✓ Group By, Transpose
- ✓ Header Row Promotion
- ✓ Reverse Rows, Count Rows
- ✓ Any Column Transformations
- ✓ Data Type, Fill & Pivot

Ch 5: Power Query Level 3

- ✓ Text Transformations
- ✓ Format, SubString
- ✓ Number Transformations
- ✓ Date Time Transformations
- ✓ Add Column Transformation

Ch 6: Power Query Level 4

- ✓ Column From Examples
- ✓ Conditional Column
- ✓ Index Transformation
- ✓ Duplicate Rows, Errors
- ✓ Advanced Editor

Ch 7: Power Query Level 5

- ✓ ETL Parameters
- ✓ Big Data Access
- ✓ Static Parameters
- ✓ Dynamic Parameters

- ✓ List Queries
- ✓ M Language Expressions

Ch 8: Stream House, KQL

- ✓ Need for Stream House
- ✓ Auto creation of KQL
- ✓ Manual KQL Databases
- ✓ Differences with Warehouse
- ✓ Differences with Lakehouse

Ch 9: KQL Query Sets

- ✓ KQL Database Extraction
- ✓ File Imports - on Premises
- ✓ Metadata Edit Options
- ✓ Query Analytics
- ✓ Exports, Visualizations
- ✓ Query Sets Versus Notebooks

Ch 10: Fabric Data Activator

- ✓ Need for Alerts, Notifications
- ✓ Fabric Data Activator Options
- ✓ Alert Conditions, Thresholds
- ✓ Email Notifications
- ✓ Events & Notifications
- ✓ Edit / Enable / Disable

Ch 11: Mirror Database

- ✓ Need for Mirror Databases
- ✓ Configure Mirror Databases
- ✓ Data Replication
- ✓ Schema Replication
- ✓ Connections & Usage
- ✓ Mirror DB Practical uses

Part 3: Python, PySpark, DWH

Ch 1: Fabric Notebooks

- ✓ Need for Notebooks
- ✓ Fabric Notebook Types
- ✓ Creating Environment
- ✓ Creating Spark Clusters
- ✓ Standard, High Concurrency
- ✓ Magic Command
- ✓ Freeze Cells

Ch 2: Spark SQL - 1

- ✓ Spark SQL Notebooks
- ✓ Creating Schemas

- ✓ Delta Tables
- ✓ Parquet Tables
- ✓ Spark Joins
- ✓ Data Partitioning
- ✓ Union, Views in Spark

Ch 3: Spark SQL - 2

- ✓ Math, Sort Functions
- ✓ String, Date Time Functions
- ✓ Conditional Statements
- ✓ Data Recovery & Undo
- ✓ Version Number
- ✓ Describe Extended

Ch 4: Python Intro & Print

- ✓ Python Introduction
- ✓ Python Versions
- ✓ Python in Spark (PySpark)
- ✓ Python Print()
- ✓ Single, Multiline Statements

Ch 5: Python Variables

- ✓ Defining Variables
- ✓ Using Variables
- ✓ Printing Variables
- ✓ Display Variables
- ✓ Variable Types
- ✓ Multi Value Variables
- ✓ If ... Else Statement

Ch 6: Python Operators

- ✓ Integer Operators
- ✓ String Operators
- ✓ Arithmetic Operators
- ✓ Assignment Operators
- ✓ Comparison Operators
- ✓ Formatted Strings
- ✓ Indexing Operators
- ✓ ELIF, ELSE IF Statements

Ch 7: Python Data Types

- ✓ Python Data Types
- ✓ Integer / Int Data Types
- ✓ Float, String Data Types
- ✓ List Data Type
- ✓ List Items, Indexes
- ✓ Tuple Data Type
- ✓ Dictionary Data Type

Ch 8: Python Dataframes

- ✓ Pandas Module (Python)
- ✓ Dataframes from Lists
- ✓ Dataframe from Dict
- ✓ Pandas Dataframes
- ✓ Dataframe print, display

Ch 9: Python Dataframes Transformations

- ✓ Append
- ✓ Append with NoIndex
- ✓ Merge with ON, KIND
- ✓ spark.read.csv()
- ✓ spark.read.format()

Ch 10: Medallion Architecture

- ✓ Bronze, Gold and Silver
- ✓ Raw Data
- ✓ Data Preparation (Prepping)
- ✓ Temporary Views
- ✓ Big Data Analytics

Ch 11: PySpark: Medallion Loads 1

- ✓ Data Prep (Silver)
- ✓ Filtering DataFrame Records
- ✓ Removing Duplicate Records
- ✓ Sorting and Limiting Records
- ✓ Spark SQL Dataframes
- ✓ Gold Layer Implementation
- ✓ Testing Aggregated Loads

Ch 12: PySpark: Medallion Loads 2

- ✓ Azure SQL DB Connections
- ✓ SQL Queries in PySpark
- ✓ Data Prep (Silver)
- ✓ Filtering Null Values
- ✓ Grouping and Aggregating
- ✓ Spark SQL Dataframes
- ✓ Gold Layer Implementation
- ✓ Notebook Utilities

Ch 13: PySpark: SCD

- ✓ Slowly Changing Dimension
- ✓ Merge Into Statement
- ✓ Error Handling
- ✓ Merge with OLTP Data Sources

Ch 14: PySpark: Widgets

- ✓ Notebook Parameters
- ✓ Text Widgets

- ✓ Parameters & JSON
- ✓ Notebook Schedules, Retry
- ✓ Modular Notebook Design
- ✓ Notebook Chaining

Ch 15: LakeHouse Architecture Optimizations

- ✓ Delta Log
- ✓ Delta Versioning
- ✓ Partition Strategy
- ✓ Small File Problem
- ✓ Adaptive Query Execution
- ✓ Explain Plan, Spark UI
- ✓ mssparkutils

Ch 16: LakeHouse Optimizations

- ✓ VACUUM, OPTIMIZE
- ✓ ZORDER
- ✓ Broadcast Join
- ✓ Cache, Persist
- ✓ Shuffle Optimization

Ch 17: Semantic Models

- ✓ Creating Semantic Model
- ✓ Spark SQL: DDL, DML
- ✓ Adding References, Keys
- ✓ Using Model Layouts

Ch 18: Fabric Security

- ✓ Workspace Security
- ✓ Lakehouse Security
- ✓ Notebook Security
- ✓ Security Principals
- ✓ Authentication Options
- ✓ MFA (Multi Factor Authentication)

Module 4: Real-Time Projects

Project 1 : E-Commerce Domain on Databricks

Project Title

Retail Sales Analytics Platform (Medallion Architecture)

Project Overview

Students will build a production-style data platform using Lakehouse Architecture (Bronze, Silver, and Gold), implementing modern ETL practices, data quality validation, incremental processing, orchestration, optimization, and reporting.

Business Scenario

A multinational retail company receives daily data from multiple operational systems. The company wants to build a centralized analytics platform to:

- Consolidate data from different business domains
- Improve data quality and consistency
- Track sales performance across regions
- Analyse customer purchasing behaviour
- Monitor inventory levels
- Generate business KPIs for decision-makers
- Deliver Power BI dashboards with near real-time insights

Source Systems

Students will work with multiple datasets including:

- Customer Master
- Product Master
- Orders
- Inventory
- Region Master

Technologies Covered

- Apache Spark
- PySpark
- Spark SQL
- Delta Lake
- Unity Catalog
- Databricks Workflows
- Git Integration
- Fabric ETL
- Fabric ELT
- OneLake

Resume Highlights

After completing this project, you can confidently showcase experience in:

- PySpark & Spark SQL
- Delta Lake
- Medallion Architecture
- ETL Pipeline Development
- Incremental Data Processing
- Data Quality & Validation
- Unity Catalog
- Workflow Automation
- Performance Optimization

Project 2 : Microsoft Fabric Data Engineering Business Scenario

A multinational e-commerce company wants to centralize sales, customers, products, orders, inventory, logistics and payment data into Microsoft Fabric.

Goal:

Create a scalable Fabric Data Platform capable of processing millions of records daily.

Skills Gained:

- Fabric Pipelines
- Fabric Notebooks
- Fabric Data Flow
- Semantic Models
- Medallion Architecture (Bronze/Silver/Gold)
- Real-Time Industry Experience
- Fabric APIs (REST, Workspace)

Components For Project (From Resume Perspective):

- Bronze
- Silver
- Gold
- PySpark
- CI/CD
- Deployment
- End to End Integrations
- DP-700 Exam: Complete Guidance

Bonus Module: Databricks Data Engineer Certification Guidance

- ✓ Exam Q & A, Scenarios
- ✓ Certification Exam Objectives
- ✓ Topic-Wise Revision
- ✓ Scenario-Based Questions
- ✓ Practice Assessments
- ✓ Mock Exams
- ✓ Databricks Interview Questions
- ✓ Real-Time Scenario Discussions
- ✓ Resume Project Explanation
- ✓ Technical Mock Interview

Why Choose This Training?

-  Industry-Oriented Curriculum
-  End-to-End Real-Time Projects & Solutions
-  Medallion Architecture Implementation
-  100% Hands-on Practical Sessions
-  Production-Ready ETL Pipelines
-  PySpark & Spark SQL from Basics to Advanced

-  Interview Preparation
-  Assignments & Practice Labs
-  100% Practical • No Unnecessary Theory
-  Learn Like You're Working in a Real Company

EXPLORE OUR FREE DATABRICKS LEARNING SERIES

Watch practical Databricks content before you enroll:

- ✓ Technical Tutorials
 - ✓ Short & Full Demo Classes
 - ✓ Real-Time Project Walkthroughs
 - ✓ PySpark, Spark SQL & Delta Lake
 - ✓ Interview Questions & Scenarios
 - ✓ Job Roles & Career Guidance
-  **Free Videos at:** <https://www.youtube.com/playlist?list=PLT3eLlIc1sgg>

BUILD REAL SKILLS. WORK ON REAL PROJECTS. BECOME JOB READY.

At SQL School, our focus goes beyond completing a syllabus. You will learn through hands-on labs, production-style scenarios, end-to-end projects, assignments, interview preparation, and certification guidance designed to build practical Databricks Data Engineering skills.

- ✓ Practical Skill Development
- ✓ Real-Time Project Guidance
- ✓ Resume & Interview Preparation
- ✓ Certification Guidance

-  **Ready to Experience the Training?**
-  Join our **FREE Live Demo Session.**

-  **Call/WhatsApp: +91 9951440801**
-  **Register: www.sqlschool.com/register**