Where trainees meet real Support

Databricks & PySpark Course

Complete Python, Apache Spark, Databricks, DBFS & Delta Lake Masterclass

8 Weeks Duration • 40 Total Lectures (5 Lectures/Week) • Beginner to Advanced

About the Course

- **Structured Learning Journey:** From Python programming to enterprise Apache Spark, Databricks & Delta Lake
- **Career-Oriented:** Designed specifically for aspiring Big Data Engineers, ETL developers, and Cloud Data Architects
- **Comprehensive Scope:** Covers Python OOP, Spark RDDs/DataFrames, DBFS, ADLS Gen2 mounting, and Delta Lake
- **Hands-On Methodology:** 100% practical lab-driven learning on Databricks clusters with real Big Data datasets
- **Portfolio Capstone:** Architect a production Medallion Lakehouse pipeline with ACID transactions and SCD Type 2

Who Can Attend?

- Aspiring Big Data Engineers & Cloud Lakehouse Developers
- Python & SQL Developers scaling to Apache Spark ecosystems
- Data Analysts & BI Engineers upgrading to PySpark pipelines
- IT Professionals transitioning into Databricks & Delta Lake roles

**No previous Big Data experience required. Starting points are tailored to learner background.*

Course Outline (Part 1 - Python Foundations & Apache Spark Core)

Phase 1: Python Overview, Datatypes, Operators & Control Flow (Weeks 1–2)

- **Python Overview:** History & features of Python, Python vs. other programming languages, first Python program, basic syntax, Python development tools & packages
- **Environment Setup:** Installing Python, verify & setup local and cloud Python environments (IDEs, virtual environments, pip)
- **Python Datatypes:** Variables declaration, dynamic typing, primitive data types (int, float, bool), Strings operations & indexing, Type Casting
- **Python Operators:** Arithmetic operators, Relational (comparison) operators, Logical operators, Bitwise operators, Assignment operators
- **Flow Control & Loops:** Conditional branching (if, if-else, if-elif-else), while loops, do-while loop logic, and for loops with range()
- **Outcome:** Write structured Python programs and master core procedural programming fundamentals.

Phase 2: Functional Programming, Collections, File I/O, OOP & Exceptions (Week 3)

- **Functional Programming:** Function declarations, calling functions, functions call-by-name, named arguments, variable arguments (*args, **kwargs), default parameter values, and anonymous Lambda functions
- **Python Collections:** In-depth Lists, Tuples, Sets, and Dictionaries; common operations, methods, comprehensions & indexing
- **Python Files I/O:** Reading input from console, reading data from files (text/csv), writing and appending data to files, context managers (with)
- **Python Object Oriented Programming (OOP):** Classes & Objects, creating a simple class, class attributes vs. instance attributes, methods, constructors (__init__), and Inheritance (Single & Multiple)
- **Python Exception Handling:** Throwing exceptions (raise), try-except blocks, catching specific exceptions, handling multiple exceptions, and the finally clause for resource cleanup
- **Outcome:** Build robust, modular, object-oriented Python scripts capable of handling complex file manipulation and errors.

Phase 3: Introduction to Apache Spark & Spark Core Architecture (Week 4)

- **Spark Architecture:** Overview of Apache Spark architecture, Driver Node, Worker Nodes, Cluster Manager, Executors & Lazy Evaluation
- **Spark Data Abstractions:** RDD (Resilient Distributed Datasets) vs. DataFrames (DataFlow) vs. Datasets; schema immutability & execution plans
- **Transformations & Actions:** Narrow transformations (map, filter) vs. wide transformations (groupBy, join), stage boundaries, shuffling & Actions (count, collect, show, take)
- **Spark SQL:** Run SQL queries in Databricks using Spark SQL, register temporary views, write ANSI SQL queries, and perform aggregations
- **Outcome:** Understand distributed memory compute and execute high-performance analytical queries with Spark SQL.

SUPPORT ME
—BY THE DEVELOPERS' GROUP

Course Outline (Part 2 - PySpark, Databricks Platform & Delta Lake)

Phase 4: Databricks Workspace, Clusters & External Integrations (Week 5)

- **Introduction to Databricks:** Create Databricks Workspace, configure Single-node & Multi-node Clusters, Databricks Runtime (DBR), cluster policies & autoscaling
- **Databricks Notebooks:** Create & organize notebooks, multi-language magic commands (%python, %sql, %scala, %md), interactive markdown & cell execution
- **Security & Key Vault:** Azure KeyVault integration, Databricks Secret Scopes, securing access keys, tokens & connection credentials
- **Integrate with External Resources:** Create mount points with ADLS Gen2, Azure Blob Storage accounts, Azure SQL & Azure Synapse; read and write data directly from ADLS and SQL databases

Phase 5: Databricks File System (DBFS) & Databricks Utilities (dbutils) (Week 6)

- **Databricks File System (DBFS):** DBFS architecture, create, copy, and move files within DBFS; handle multiple files and directory structures; archive files and folders in DBFS
- **Databricks Utilities (dbutils) Deep Dive:** 1. **dbutils.fs:** File system operations (cp, mv, rm, ls, head, put, mounts) • 2. **dbutils.secrets:** Securely retrieve and pass secrets • 3. **dbutils.notebook:** Programmatic notebook execution, chaining workflows & exit parameter passing • 4. **dbutils.widgets:** Create interactive dropdowns, text boxes, and comboboxes for parameterized notebook execution

Phase 6: Introduction to PySpark & Multi-Format Ingestion (Weeks 6–7)

- **Multi-Source Data Ingestion:** Read and write data from CSV, JSON, Parquet, Avro, Azure SQL, DBFS, and ADLS Gen2 with defined StructType schemas
- **Transformations using PySpark:** Column manipulation (withColumn, withColumnRenamed, drop), filtering, sorting, handling nulls/NaNs (fillna, dropna), string/date-time functions, and regex expressions
- **Aggregations & Joins:** GroupBy, aggregate functions (count, sum, avg), and Joins (Inner, Left Outer, Right Outer, Full Outer, Cross)

Phase 7: Delta Lake Architecture, DML & SCD Type 1 and Type 2 (Weeks 7–8)

- **Delta Lake Overview:** Delta Lake architecture, difference between Data Lake & Delta Lake, Parquet storage with transaction log (_delta_log), ACID transactions compliance
- **Delta Tables Management:** How to create Delta Lake Tables (Managed vs. External), table properties, schema evolution & schema enforcement
- **DML Operations:** INSERT, UPDATE, DELETE, and MERGE INTO (atomic UPSERT operations) on Delta tables
- **Slowly Changing Dimensions (SCD):** How to manage SCD Type 1 (overwrite current values) and SCD Type 2 (historical versioning with start date, end date & active flag)
- **Time Travel & Auditing:** History logs (DESCRIBE HISTORY), querying historical snapshots by version/timestamp, restore tables, and VACUUM

Phase 8: Databricks Performance Optimization & Tuning (Week 8)

- **Cost Optimization Techniques:** Cluster sizing, auto-termination policies, spot instances, pool management & cost governance
- **Performance Optimization Engines:** Catalyst Optimizer architecture (Analysis, Logical Plan, Physical Plan, Code Generation)
- **Memory Management:** Caching & Persisting (Cache vs. Persist), StorageLevels (MEMORY_ONLY, MEMORY_AND_DISK), and unpersist()

Hands-on Capstone Projects

- **Enterprise End-to-End Lakehouse Ingestion Pipeline:** Ingest raw CSV/JSON files from ADLS Gen2 into DBFS using mount points & dbutils, transform with PySpark, and load into Delta Lake Medallion layers.
- **Production SCD Type 2 Customer Dimension Pipeline:** Implement robust MERGE INTO logic in Delta Lake to maintain historical audit trails with effective dating and active record flags.
- **Parameterized Orchestration & Tuning Engine:** Build automated multi-notebook workflows using dbutils.notebook, dbutils.widgets, Key Vault secrets, and benchmark query acceleration with Catalyst & Caching.

Tools & Technologies

Databricks, Apache Spark, PySpark, Python 3, Delta Lake, DBFS, Databricks Utilities (dbutils), Azure ADLS Gen2, Azure SQL Database, Azure Synapse, Azure Key Vault, Spark SQL, Parquet, JSON, CSV

High-Value POC Categorization

- **Beginner:** Python scripts, DBFS file management, basic notebooks
- **Intermediate:** Multi-format PySpark ETL, dbutils widgets, Delta MERGE
- **Advanced:** SCD Type 1 & 2, Mount points (ADLS/Synapse), Catalyst tuning

Why This Program?

- 100% hands-on cloud labs with real-world enterprise architectures
- Complete coverage from Python basics to advanced Delta Lake SCD 1/2
- Production-ready dbutils, Key Vault integration & performance tuning
- Targeted interview prep for Databricks & PySpark Data Engineer roles

How This Fits Your Full Learning Stack

- Frontend → UI • Backend → Java / Python • Cloud → AWS / Azure • Data → Databricks & PySpark Data Engineering
- **Final Result: Full Stack + AI + Cloud + Data Engineer**