MASTERING AZURE DATA ENGINEERING

Complete 110-Hour Industry-Ready Curriculum with Generative AI ● DP-203 & DP-700 Prep

MODULE 1: Cloud & Azure Fundamentals for Data Engineers

Duration: 3 Hours | **Level:** Beginner

- What is Cloud Computing & Microsoft Azure
- Azure Global Infrastructure
  - Regions, Availability Zones, Resource Groups
  - Paired regions for disaster recovery
- Azure Free Account Setup & Portal Overview
- Azure CLI Setup & Configuration
- Azure Cloud Shell — Bash & PowerShell in browser
- Azure SDK for Python — azure-sdk Overview
- Azure Pricing Model & Cost Management
- Azure Resource Manager (ARM) — Infrastructure Overview
- Core Azure Services for Data Engineers — Compute, Storage, Analytics
- Azure Subscription & Management Groups Hierarchy

🔬 **Lab:** Create Azure Free Account, Set Up CLI, Create Resource Group

MODULE 2: Azure Identity, Security & Access Management

Duration: 4 Hours | **Level:** Beginner-Intermediate

- What is Azure Active Directory (Azure AD / Entra ID)
- Azure AD Users, Groups & Service Principals
- Managed Identities — System Assigned vs User Assigned
- Role-Based Access Control (RBAC) — Roles & Assignments
- Azure Key Vault — Store Secrets, Keys & Certificates
- Accessing Key Vault Secrets from Python (azure-keyvault SDK)
- Service Principal Authentication — Client ID & Client Secret
- IAM Best Practices for Data Engineering
  - Principle of least privilege, Managed Identity over Service Principal
- Encryption at Rest & Encryption in Transit — Azure Standards

- PII Data Masking & Sensitive Data Handling
- Network Security — Private Endpoints & VNet Integration

🔬 **Lab:** Create Service Principal, Assign RBAC, Store Secrets in Key Vault • Access Key Vault Secrets from Python using Managed Identity

MODULE 3: Azure Storage — ADLS Gen2 & Blob Storage

Duration: 4 Hours | **Level:** Beginner-Intermediate

- Azure Blob Storage — Containers, Blobs & Access Tiers
- Azure Data Lake Storage Gen2 — Overview & Architecture
- ADLS Gen2 vs Blob Storage — Key Differences
- Hierarchical Namespace — Folders & Files in ADLS
- ADLS Gen2 Access Control — RBAC & ACLs
- Storage Account Setup & Configuration
- Storage Access Methods
  - Access Keys, SAS Tokens, Service Principal, Managed Identity
- Connecting ADLS Gen2 from Python (azure-storage-blob SDK)
- Uploading & Downloading Files from ADLS using Python
- ADLS Gen2 Partitioning Strategy for Data Lakes
  - Partition by date, region, entity — best practices
- Lifecycle Management Policies — Hot, Cool, Archive Tiers
- ADLS Gen2 as Data Lake — Bronze, Silver, Gold Layers
- Private Endpoints for ADLS — Secure Access

🔬 **Lab:** Build Azure Data Lake (Bronze/Silver/Gold), Upload & Read Files with Python SDK

📝 **Quiz:** Azure Storage & IAM — 15 Questions

MODULE 4: Azure Data Factory (ADF) — Core Concepts

Duration: 3 Hours | **Level:** Intermediate

- What is Azure Data Factory & When to Use It
- ADF Architecture
  - Pipelines — Logical grouping of activities
  - Activities — Individual operations (Copy, Transform, Control)
  - Datasets — Define source & sink data structures
  - Linked Services — Connections to data sources
- ADF vs AWS Glue — Key Differences
- Creating ADF Instance & Studio Overview
- Integration Runtime — Azure IR, Self-Hosted IR, SSIS IR
- ADF Triggers
  - Schedule Trigger — Cron-based runs
  - Tumbling Window Trigger — Non-overlapping time windows
  - Event-Based Trigger — Fire on ADLS file arrival
  - Manual Trigger — On-demand runs
- ADF Pipeline Monitoring & Alerts

- ADF Pricing — Data Integration Units (DIU)

🔬 **Lab:** Create First ADF Pipeline — Copy CSV from HTTP to ADLS

MODULE 5: ADF — Data Ingestion & Copy Activity

Duration: 5 Hours | **Level:** Intermediate

- Copy Activity — Source to Sink Data Movement
- Supported Sources — ADLS, Blob, SQL, REST API, HTTP, SFTP, S3, Oracle
- Supported Sinks — ADLS, Synapse, SQL, Databricks Delta
- Copy Activity Performance Tuning — Parallelism & DIU
- Incremental Data Copy — Watermark Pattern
  - Track LastModifiedDate in a control/watermark table
  - Copy only new or changed records each run
- Incremental Copy using LastModifiedDate
- Incremental Copy using Change Tracking (SQL Server)
- Binary Copy — Moving Files Without Transformation
- Parameterised Pipelines — Dynamic Source & Sink Paths
- ADF Self-Hosted Integration Runtime — On-Premise Sources
  - Install SHIR on-premise, register with ADF
  - Use for SQL Server, Oracle, File shares behind firewall

🔬 **Lab:** Ingest Azure SQL → ADLS Gen2 Full Load, Incremental Copy with Watermark Pattern

MODULE 6: ADF — Mapping Data Flows (No-Code ETL)

Duration: 5 Hours | **Level:** Intermediate

- Data Flow Activity — Mapping Data Flows (No-Code ETL)
- Data Flow Transformations
  - Source & Sink — Read and write configuration
  - Filter & Select — Row and column filtering
  - Derived Column — Add or transform columns with expressions
  - Aggregate — groupBy, sum, avg, count
  - Join — Inner, Left, Right, Full outer joins
  - Lookup — Enrich data from reference table
  - Conditional Split — Route rows based on conditions
  - Union — Combine multiple data sources
  - Flatten — Handle Nested JSON & Array columns
  - Surrogate Key — Auto-Generate integer keys
  - Window Functions — Rank, Lead, Lag in Data Flows
- Wrangling Data Flows — Power Query in ADF
- Data Flow Debug Mode — Testing Transformations
- Data Flow Performance — Partitioning, Core count & Optimization

🔬 **Lab:** Build ETL Pipeline using Mapping Data Flows • Transform Nested JSON from REST API using ADF

MODULE 7: ADF — Control Flow & Pipeline Orchestration

Duration: 5 Hours | **Level:** Intermediate

- ForEach Activity — Loop Over Files or Tables Dynamically
- If Condition Activity — Conditional Branching in Pipeline
- Switch Activity — Multi-Branch Logic
- Until Activity — Retry Until Condition is Met
- Wait Activity — Add Delays Between Activities
- Get Metadata Activity — Check File Existence & Properties
- Set Variable & Append Variable Activities
- Execute Pipeline Activity — Call Child Pipelines
- Lookup Activity — Fetch Config Data from Database
- Web Activity — Call REST APIs from ADF
  - Call external APIs, trigger Logic Apps, webhook notifications
- Stored Procedure Activity — Execute SQL Procedures
- ADF Error Handling
  - On Failure path — send alert, write error log
  - Retry count & retry interval configuration

🔬 **Lab:** Dynamic Pipeline — Process Multiple Files with ForEach • Metadata-Driven Pipeline Framework — Config Table Approach

📝 **Quiz:** ADF Pipelines & Activities — 20 Questions

MODULE 8: ADF — Advanced Features, CI/CD & Testing

Duration: 5 Hours | **Level:** Advanced

- What is CI/CD in Data Engineering Context
- Git Branching Strategy — Feature, Dev, QA, Production
- ADF Git Integration — GitHub & Azure DevOps
- ADF Publishing — Dev → Test → Production promotion
- ADF ARM Template Export & Deployment
- Parameterised Linked Services & Global Parameters
- Azure DevOps Pipeline — Deploy ADF Across Environments
- ADF with Key Vault — Secure Credentials (no hardcoding)
- ADF Monitoring — Pipeline Runs, Activity & Trigger Runs
- ADF Alerts with Azure Monitor — Email/SMS on Failure
- Databricks Repos — Git Integration for Notebooks
- Databricks Asset Bundles (DAB) — Deploy Jobs & DLT Pipelines
- Infrastructure as Code — ARM Templates & Bicep Basics
- Terraform Basics — Deploy ADF & ADLS from Scratch
- Unit Testing PySpark — pytest Basics
- Integration Tests for ADF Pipeline Outputs

🔬 **Lab:** ADF CI/CD Dev → Production with Azure DevOps • Databricks CI/CD with GitHub Actions + DAB

MODULE 9: Azure Databricks — Platform & Workspace Setup

Duration: 4 Hours | **Level:** Intermediate

- What is Azure Databricks & Databricks Lakehouse
- Azure Databricks Architecture — Control Plane vs Data Plane
- Creating Azure Databricks Workspace
- Databricks Cluster Configuration on Azure
  - Standard vs High Concurrency Clusters
  - Auto-Scaling & Auto-Termination
  - Cluster Policies — Control Cost & Config
- Databricks Runtime Versions — Standard, ML, Photon
- Connecting Azure Databricks to ADLS Gen2
  - Service Principal Authentication
  - OAuth 2.0 with App Registration
  - Mounting ADLS Gen2 in Databricks
- Databricks Utilities (dbutils) — secrets, fs, widgets
- Databricks Cost Optimization
  - Spot Instances vs On-Demand
  - Auto-Termination Best Practices
  - Cluster Sizing Guidelines & DBU Pricing

🔬 **Lab:** Set Up Azure Databricks Workspace & Connect ADLS Gen2 • Configure Cluster Policies for Cost Control

MODULE 10: Azure Databricks — PySpark on Azure

Duration: 5 Hours | **Level:** Intermediate

- Reading Data from ADLS Gen2 in PySpark
- Writing Data to ADLS Gen2 — Parquet, Delta, CSV
- PySpark DataFrame Operations on Azure
  - select, filter, groupBy, agg, join, withColumn, window functions
- Working with Azure SQL Database from Databricks
  - JDBC Connection to Azure SQL
  - Reading & Writing SQL Tables efficiently
- Working with Azure Synapse from Databricks
  - Synapse Connector — Optimised Bulk Read/Write
- Databricks Secrets — Store ADLS & SQL Credentials Securely
  - Secret Scopes backed by Azure Key Vault
- Unit Testing PySpark Code
  - pytest with PySpark — local test runs
  - Writing testable transformation functions
  - Mocking Spark DataFrames in tests
  - Running tests in Databricks & locally (CI-friendly)

🔬 **Lab:** PySpark ETL Pipeline ADLS → Transform → Delta Lake • Write Unit Tests for PySpark Transformations using pytest

MODULE 11: Azure Databricks — Delta Lake on Azure

Duration: 5 Hours | **Level:** Intermediate-Advanced

- Delta Lake on Azure Databricks
- Creating Delta Tables on ADLS Gen2
- Delta Operations — Update, Delete, Merge (Upsert)
- Schema Evolution & Enforcement in Delta
- Delta Time Travel — Query Historical Data
  - VERSION AS OF, TIMESTAMP AS OF
  - Restore previous versions with RESTORE command
- Delta OPTIMIZE & Z-Ordering on Azure
- Liquid Clustering on Azure Databricks
- Change Data Feed (CDF) — Incremental Processing
- Medallion Architecture on Azure — Bronze, Silver, Gold on ADLS
- Delta Lake Metrics & Health Monitoring
  - Transaction log analysis, Table statistics, OPTIMIZE history
- Delta Live Tables (DLT) on Azure Databricks
  - Creating DLT Pipelines — Streaming & Batch modes
  - Data Quality with Expectations (expect, expect_or_drop, expect_or_fail)
  - DLT Deployment Patterns — Dev, Staging, Prod
  - Monitoring DLT Runs — Event log & pipeline graph

🔬 **Lab:** Full Medallion Pipeline ADLS Bronze → Delta Silver → Gold • Deploy DLT Pipeline across Dev and Production environments

📝 **Quiz:** Delta Lake & DLT — 20 Questions

MODULE 12: Azure Databricks — Production Engineering

Duration: 6 Hours | **Level:** Advanced

- Databricks Job Scheduling & Orchestration
  - Creating Jobs & Multi-Task Workflows
  - Job Clusters vs All-Purpose Clusters
  - Retry Policies, Error Handling & Job Parameters
  - Monitoring Job Runs — Metrics & Logs
- Databricks Workflows vs Apache Airflow — When to Use What
- Unity Catalog — Enterprise Data Governance
  - Unity Catalog Architecture — Metastore → Catalog → Schema → Table
  - Unity Catalog vs Legacy Hive Metastore
  - Access Control — Users, Groups, Service Principals
  - Table & Column Level Permissions
  - Row-Level Security & Column Masking for PII
  - Data Lineage — Track Data Flow Across Pipelines
  - Tagging & Data Classification
  - Audit Logs in Unity Catalog
  - External Locations & Storage Credentials
  - Delta Sharing — Share Data Across Organizations

- MLflow Basics for Data Engineers
  - Tracking Experiments, Logging Metrics, Model Registry Overview
- Databricks Observability & Monitoring
  - Spark UI Analysis for Data Engineering Jobs
  - Custom Metrics with Azure Monitor, Alerts on Job Failures

🔬 **Lab:** Multi-Task Databricks Workflow, Unity Catalog Setup & PII Masking, Azure Monitor Alerts

MODULE 13: ADF + Azure Databricks Integration

Duration: 5 Hours | **Level:** Advanced

- Why Combine ADF & Databricks — Orchestration vs Transformation
- ADF Databricks Activity Types
  - Notebook Activity — Run Databricks Notebook from ADF
  - Jar Activity — Submit Spark JAR Job
  - Python Activity — Submit Python Script
- Passing Parameters from ADF to Databricks Notebook
- Receiving Output from Databricks back to ADF (via dbutils.notebook.exit)
- ADF Pipeline — Trigger Databricks on File Arrival
- ADF + Databricks — Full Medallion Pipeline
  - ADF ingests raw data to ADLS Bronze layer
  - Databricks DLT transforms Bronze → Silver → Gold
  - ADF monitors completion & triggers downstream activities
- Error Handling — Retry Databricks from ADF on Failure
- End-to-End Pipeline Observability
  - ADF pipeline run tracing with Correlation IDs
  - Databricks job metrics
  - Azure Monitor end-to-end dashboard

🔬 **Lab:** ADF Pipeline calling Databricks Notebook with Parameters • Full Pipeline — SQL Server → ADF → ADLS → Databricks DLT → Synapse

MODULE 14: Azure Synapse Analytics

Duration: 4 Hours | **Level:** Intermediate

- What is Azure Synapse Analytics
- Synapse vs Databricks — When to Use What
- Synapse Architecture
  - Dedicated SQL Pool — Enterprise Data Warehouse
  - Serverless SQL Pool — Query ADLS directly with SQL (no cluster needed)
  - Synapse Spark Pool — Managed Apache Spark
- Creating External Tables on ADLS from Synapse Serverless SQL
  - OPENROWSET — Query Parquet, Delta, CSV on ADLS
  - External data sources, file formats, external tables
- Synapse Link — Connect Cosmos DB to Synapse (no ETL)
- Loading Data into Synapse — COPY INTO Command
- Synapse with Power BI — Direct Integration for Reporting

- Synapse Analytics Security — Column-Level & Row-Level Security

🔬 **Lab:** Query ADLS Data Lake with Synapse Serverless SQL • Load Delta Tables from Databricks Gold Layer into Synapse

MODULE 15: Azure SQL Database & Change Data Capture

Duration: 4 Hours | **Level:** Intermediate

- Azure SQL Database — Overview & Use Cases
- Azure SQL Tiers — DTU vs vCore — choosing the right tier
- Connecting Azure SQL from Python — pyodbc & sqlalchemy
- Azure SQL as Source in ADF — Full & Incremental Load
- Azure SQL as Sink — Writing from Databricks & ADF
- Change Tracking in Azure SQL — Lightweight CDC
- Change Data Capture (CDC) Pattern
  - Capture inserts, updates & deletes at row level
  - CDC with ADF — Copy changed records to ADLS
  - CDC with Databricks Delta Merge
- SCD Type 1 & Type 2 with CDC + Delta Merge
  - SCD Type 1 — Overwrite current value
  - SCD Type 2 — Track full history with effective dates
  - Implementing SCD Type 2 with Delta MERGE + valid_from/valid_to

🔬 **Lab:** ADF Incremental CDC Pipeline from Azure SQL to ADLS

MODULE 16: Azure Event Hubs & Real-Time Streaming

Duration: 6 Hours | **Level:** Advanced

- What is Azure Event Hubs — Streaming Architecture
- Event Hubs vs Apache Kafka — Similarities & Differences
- Event Hubs Namespace, Topics (Event Hubs) & Partitions
- Sending Events to Event Hubs from Python
- Reading Events in Databricks Structured Streaming
- Exactly-Once Semantics with Event Hubs + Delta Lake
- Watermarking — Handling Late-Arriving Data
  - allowed lateness, drop vs include late events
- Backpressure Handling in Structured Streaming
- Stateful vs Stateless Streaming — When to Use Each
- Checkpointing & Fault Recovery
- Azure Stream Analytics — Real-Time SQL on Streams
  - Stream Analytics Input — Event Hubs, IoT Hub
  - Stream Analytics Output — ADLS, SQL, Power BI
  - Windowing — Tumbling, Hopping, Session windows

🔬 **Lab:** Event Hubs → Databricks Structured Streaming → Delta Lake • Late Data Handling — Watermark + Allowed Lateness • Stream Analytics Real-Time Aggregation to Power BI

📝 **Quiz:** Streaming & Real-Time Processing — 20 Questions

MODULE 17: ADF Real-World Pipeline Patterns

Duration: 5 Hours | **Level:** Advanced

- Pattern 1 — Metadata-Driven Generic Pipeline
  - Config table drives which tables/files to load
  - ForEach loops dynamically over table list
- Pattern 2 — Incremental Load with Watermark
  - Track LastModifiedDate in control table
  - Copy only new/changed records each scheduled run
- Pattern 3 — File-Based Event Trigger Pipeline
  - Event-based trigger on ADLS file arrival
  - ADF picks up file → calls Databricks notebook
- Pattern 4 — Full Medallion Architecture Pipeline
  - ADF ingests raw → Bronze ADLS
  - Databricks DLT → Silver layer
  - Databricks aggregates → Gold layer
  - Synapse Serverless queries Gold for Power BI reporting
- Pattern 5 — REST API to Data Lake Pipeline
  - Web Activity calls REST API, stores JSON to ADLS Bronze
  - Databricks flattens JSON → Silver → Gold
- Pattern 6 — CDC Pipeline with Merge
  - SQL CDC → ADF → ADLS → Databricks Delta Merge
  - SCD Type 2 with history tracking

🔬 **Lab:** Metadata-Driven Framework from Scratch, CDC Pipeline with Delta Merge

MODULE 18: Azure Monitor, Observability & End-to-End Tracing

Duration: 4 Hours | **Level:** Advanced

- Azure Monitor Overview for Data Pipelines
- ADF Monitoring — Pipeline, Activity & Trigger Runs
- Databricks Cluster & Job Monitoring
- Delta Lake Metrics — Table Health & Performance
- Azure Log Analytics Workspace Setup
- Kusto Query Language (KQL) — Queries for Data Engineers
  - Query ADF pipeline failures
  - Query Databricks job durations
  - Query ADLS access logs
- End-to-End Pipeline Tracing
  - Correlation IDs across ADF & Databricks
  - Track data flow from source to Gold layer
- Setting Up Alerts — Email & SMS on Pipeline Failure

- Azure Monitor Dashboards — Pipeline Health Overview
- Custom Metrics from Databricks to Azure Monitor

🔬 **Lab:** End-to-End Monitoring Dashboard for Complete Pipeline • KQL Queries to Analyse Pipeline Failures & Durations

MODULE 19: End-to-End Capstone Projects (Graded)

Duration: 12 Hours | **Level:** Advanced

PROJECT 1 — Batch Medallion Data Platform

Must submit: Code on GitHub + CI/CD pipeline + Monitoring dashboard

- Source: Azure SQL Database (sales transactions)
- Ingestion: ADF Metadata-Driven Pipeline → ADLS Bronze Layer
- Transform: Databricks DLT Pipeline → Silver Layer (clean data)
- Aggregate: Databricks → Gold Layer (sales by region & product)
- Governance: Unity Catalog — permissions, lineage, PII masking
- Warehouse: Synapse Serverless SQL queries on Gold Layer
- Reporting: Power BI Dashboard connected to Synapse
- CI/CD: GitHub Actions + Databricks DAB deployment
- Monitoring: Azure Monitor dashboard + failure alerts

PROJECT 2 — Real-Time Streaming Pipeline

Must submit: Streaming code + late data proof + Delta table output

- Source: Python app sending e-commerce events to Event Hubs
- Processing: Databricks Structured Streaming with watermarking
- Late Data: Handle late events up to 2 hours
- Exactly-Once: Delta Lake guarantees with checkpointing
- Storage: Delta Lake on ADLS Gen2 partitioned by date
- Output: Synapse Serverless → Power BI Real-Time Dashboard

PROJECT 3 — CDC & SCD Type 2 Pipeline

Must submit: CDC config + Delta merge code + test results

- Source: On-Premise SQL Server via Self-Hosted Integration Runtime
- CDC: Capture insert, update, delete from source tables
- Ingestion: ADF CDC pipeline → ADLS Bronze with change records
- Transform: Databricks Delta Merge — SCD Type 2 implementation
- History: Track full customer history with effective dates
- Tests: pytest unit tests for merge logic + integration tests

MODULE 20: Generative AI for Azure Data Engineering

Duration: 6 Hours | **Level:** Advanced

Introduction to Generative AI in Data Engineering

- What is Generative AI & Large Language Models (LLMs)
- LLM Capabilities for Data Engineers — Code Generation, Documentation, Debugging
- Prompt Engineering Fundamentals
  - Zero-shot, Few-shot, and Chain-of-Thought prompting
  - Writing effective prompts for technical tasks
  - Context window management and token limits
- AI-Assisted vs AI-Generated Code — Best Practices
- Security & Privacy Considerations with AI Coding Tools
  - Not sharing sensitive data, credentials, or PII with AI tools
  - Code review requirements for AI-generated code

Claude.ai for Data Engineering

- Claude Overview — Anthropic's AI Assistant
- Claude Models — Opus, Sonnet, Haiku (when to use each)
- Using Claude for PySpark Development
  - Generating PySpark transformation code from requirements
  - Explaining complex Spark operations and optimization
  - Debugging PySpark errors with stack traces
  - Converting SQL queries to PySpark
- Using Claude for Azure Data Factory
  - Generating ADF pipeline JSON from requirements
  - Creating Data Flow transformation logic
  - Writing ARM template parameters
- Using Claude for Documentation
  - Auto-generating README files for data pipelines
  - Creating data dictionary from schema
  - Writing technical design documents
- Claude Projects — Organizing code and context for teams
- Claude Artifacts — Interactive code and data visualizations

GitHub Copilot for Azure Data Engineering

- GitHub Copilot Overview — AI Pair Programmer
- Installing & Configuring GitHub Copilot in VS Code
- GitHub Copilot for PySpark
  - Autocomplete for DataFrame transformations
  - Inline suggestions for window functions, aggregations
  - Generating test cases for PySpark functions
- GitHub Copilot for Python (Azure SDK, pandas, pytest)
  - Autocomplete for azure-storage-blob, azure-keyvault
  - Generating pytest fixtures and test functions
- GitHub Copilot Chat — Conversational Coding
  - Asking questions about code inline in editor
  - Explaining code blocks, functions, classes
  - Refactoring suggestions and code improvements
- Copilot Slash Commands

    - /explain — Explain selected code
    - /fix — Fix bugs or errors
    - /tests — Generate unit tests
    - /doc — Generate documentation
- GitHub Copilot Workspace — AI-Powered Development Environment

Cursor IDE for Data Engineering

- Cursor Overview — AI-First Code Editor (fork of VS Code)
- Installing & Setting Up Cursor
- Cursor AI Features for Data Engineering
    - Cmd+K — Inline code generation and editing
    - Cmd+L — Chat with AI about your codebase
    - @-mentions — Reference specific files, folders, docs
- Cursor Composer — Multi-file AI editing
- Using Cursor for PySpark Development
    - Generate entire ETL pipeline from requirements
    - Refactor legacy Spark code to modern patterns
    - Add error handling and logging across files
- Cursor Tab — AI Autocomplete (faster than Copilot)
- Cursor vs GitHub Copilot — Feature Comparison
- Codebase Indexing — Cursor learns your entire project
- Using Custom AI Models in Cursor (Claude, GPT-4, etc.)

ChatGPT for Data Engineering Tasks

- ChatGPT Overview — OpenAI's Conversational AI
- GPT-4o vs GPT-4 Turbo vs GPT-3.5 — Model Selection
- Using ChatGPT for PySpark Code Generation
    - Prompt: 'Write PySpark code to read CSV, deduplicate, write to Delta'
    - Iterative refinement with follow-up prompts
- Using ChatGPT for Azure ARM/Bicep Templates
    - Generate ARM templates for ADF, ADLS, Databricks
    - Convert ARM to Bicep and vice versa
- ChatGPT Code Interpreter — Run Python in Chat
    - Upload CSV and generate PySpark transformation code
    - Analyze data schema and suggest optimizations
- ChatGPT Custom GPTs for Data Engineering
    - Create custom GPT trained on your company's standards
    - PySpark Style Guide GPT, Delta Lake Best Practices GPT

Claude Code — Agentic Coding from Terminal

- What is Claude Code — AI Agent for Command Line Development
- Installing Claude Code CLI
- Using Claude Code for Data Engineering Workflows
    - 'claude code create a PySpark ETL pipeline for customer data'

    - Claude Code writes files, runs tests, iterates on errors
    - Multi-step tasks — ADF pipeline + Databricks notebook + tests
- Claude Code Agents — Autonomous Task Execution
    - Agent modes: architect, implement, debug, document
    - Monitoring agent actions and approving changes
- Using Claude Code with Git
    - Auto-commit with meaningful messages
    - Create branches for feature development

AI-Powered Productivity Tools

- Claude Cowork — AI Desktop Agent for File & Task Management
    - Automate file organization for data engineering projects
    - Batch rename, move, convert files with natural language
- AI-Powered Code Reviews with Claude & Copilot
    - Automated code review comments on Pull Requests
    - Suggesting PySpark performance optimizations
    - Identifying security issues (hardcoded credentials, etc.)
- AI for Documentation Generation
    - Auto-generate docstrings for Python functions
    - Create Markdown documentation from code
    - Generate Data Catalog entries from Delta tables

Practical AI Workflows for Azure Data Engineers

- Workflow 1: Generate PySpark Code with Claude → Test with Copilot → Deploy
    - Use Claude to draft initial transformation logic
    - Use Copilot to add error handling and tests
    - Deploy to Databricks with CI/CD
- Workflow 2: Debug Production PySpark Job
    - Copy error stack trace into ChatGPT or Claude
    - Ask for root cause analysis and fix
    - Apply fix and validate with unit tests
- Workflow 3: Convert SQL Stored Procedure to PySpark
    - Paste SQL into Cursor and ask 'Convert to PySpark'
    - Review logic, add comments, test output
- Workflow 4: Create ADF Pipeline from Requirements
    - Describe pipeline in natural language to ChatGPT
    - Generate ADF JSON, import into Azure Data Factory
    - Use Copilot to parameterize and optimize

Best Practices & Limitations

- Always Review AI-Generated Code
    - Verify correctness, performance, security
    - Test thoroughly before production deployment
- Understand the Limitations

    - AI tools may hallucinate (generate incorrect code)
    - AI does not understand your business logic fully
    - Keep humans in the loop for critical decisions
- Data Privacy & Compliance
    - Do not paste production data, PII, or credentials into AI tools
    - Use enterprise AI tools with data residency guarantees
- Version Control for AI-Generated Code
    - Commit AI code with clear attribution in commit messages
    - Use .ai or .copilot tags for traceability

🔬 **Lab:** Use Claude to generate a complete PySpark ETL pipeline • Use Copilot to add pytest unit tests • Use Cursor to refactor and optimize code • Deploy to Databricks and validate output

BONUS: DP-203 & DP-700 Certification Preparation

Bonus Content — 6 Hours Self-Study

- DP-203 Azure Data Engineer Associate — Exam Overview
- Exam Domains: Data Storage (40-45%), Data Processing (25-30%), Security (10-15%), Optimization (10-15%)
- DP-700 Fabric Data Engineer — New 2024 Certification
- Practice Questions — 100 DP-203 Mock Test Questions
- Exam Tips & Time Management Strategy
- Top 50 Azure Data Engineering Interview Questions & Answers

🎯 WHAT YOU'LL MASTER

✓ Design enterprise data lakes on ADLS Gen2 with Bronze/Silver/Gold zones

✓ Build metadata-driven ADF pipelines with full control flow

✓ Engineer Delta Lake pipelines on Azure Databricks with DLT

✓ Implement SCD Type 1 & Type 2 with Delta MERGE

✓ Apply Unity Catalog governance — PII masking, lineage, auditing

✓ Build real-time pipelines with Event Hubs + Structured Streaming

✓ Deploy multi-environment CI/CD with GitHub Actions + DAB

✓ Monitor end-to-end pipelines with Azure Monitor + KQL

✓ **Use GenAI tools (Claude, Copilot, Cursor) to 10x productivity**

✓ **Complete 3 graded capstone projects**

✓ **Pass DP-203 and DP-700 certification exams**

📞 ENROLLMENT & CONTACT

🌐 **Website:** databrickstraining.in

📧 **Email:** info@databrickstraining.in

📱 **Phone:** +91-8500002025

Training & Placement Excellence | Trusted by 5000+ Data Engineers