Data Engineering Fundamentals Module

Part 1: Introduction to Data Engineering

- 1.1 What is Data Engineering?
    - Definition and scope.
- 1.2 Role of Data Engineer in Data Science and Analytics
    - Collaboration with data scientists and analysts.

Part 2: Data Systems and Storage Solutions

- 2.1 Overview of RDBMS
    - Key characteristics and use cases.
- 2.2 Overview of Data Warehouse and Data Lake
    - Definitions, differences, and practical applications.
- 2.3 What is a Data Warehouse?
    - Deep dive into Data Warehouse architecture and benefits.
- 2.4 Why Data Warehouse?
    - Strategic importance in data analytics.

Part 3: Data Modeling Fundamentals

- 3.1 Data Modeling Overview
    - Purpose and principles of data modeling.
- 3.2 Data Modeling Techniques
    - Various techniques used in the field.
- 3.3 Data Modeling Types
    - Conceptual, logical, and physical models.

Part 4: Advanced Data Modeling Concepts

- 4.1 Data Modeling - Facts
    - Understanding fact tables and their role in data warehousing.
- 4.2 Data Modeling - SCD (Slowly Changing Dimensions)
    - Types of SCDs and handling dimension changes over time.
- 4.3 Data Modeling - Different Keys
    - Exploring primary keys, foreign keys, and surrogate keys.
- 4.4 Data Modeling - OLTP vs OLAP
    - Contrasting transactional and analytical processing systems.

Part 5: Practical Data Engineering

- 5.1 Stages of Data Engineering
    - From data collection to data governance.
- 5.2 Data Modeling - Real World Example
    - Applying data modeling concepts to a practical scenario.

Real World E2E Projects

- Towards the end of the module, students will undertake a capstone project that encompasses the key learnings from the course. This project could involve designing a data warehouse schema based on given requirements, including the use of SCDs, and demonstrating the ETL process on a dataset to populate the warehouse.

Supplementary Materials

- Additional readings, case studies, and resources will be provided to deepen students' understanding of each topic.

Evaluation

- Mock Interview at the end of each week to test comprehension.
- Peer-reviewed assignments for practical sections.
- E2E project presentation and report submission.

Data Modelling

Part 1: Basic Concepts of Data Modelling

- Introduction to data modeling
- Conceptual, logical, and physical models
- Best practices in data modeling

Part 2: Business Data Requirements – Entities and Classes

- Identifying business data requirements
- Entities and classes in data modeling
- Entity-relationship modeling (ER modeling)

Part 3: Business Data Requirements – Attributes

- Types of attributes in data modeling
- Attribute domains and data types
- Constraints and naming conventions for attributes

Part 4: How To Link Things Together – Relationships

- Types of relationships in data modeling
- Cardinality and optionality in relationships
- Role names and associative entities

Part 5: Requirements Analysis

- Gathering and analyzing data requirements
- Functional and non-functional requirements
- Documentation of data requirements

Part 6: Conceptual Data Modeling

- Creating a conceptual data model
- Entity-relationship diagrams (ERDs)
- Representing business processes and data flows

Part 7: Logical Data Modeling

- Transforming conceptual model to logical model
- Tables, columns, normalization
- Primary and foreign keys in data modeling

Part 8: Physical Data Modelling

- Converting logical model to physical model
- Database schema, tables, denormalization
- Indexing and partitioning strategies

Part 9: Data Modelling Tools and Techniques

- Overview of data modeling tools (e.g., ERwin, PowerDesigner)
- Reverse engineering and forward engineering
- Techniques for data model manipulation

SQL Tutorial Module

Part 1: SQL Beginner Lessons

- 1.1 Introduction to SQL
  - History and importance of SQL in data management.
  - Overview of SQL syntax and structure.
- 1.2 Basic Data Retrieval
  - Using `SELECT` statements to query data.
  - Understanding `FROM`, `WHERE`, and `ORDER BY` clauses.
- 1.3 Working with Functions
  - Introduction to common SQL functions for data manipulation (e.g., `COUNT`, `SUM`, `AVG`, `MIN`, `MAX`).
- 1.4 Data Filtering and Sorting
  - Advanced use of `WHERE` clause.
  - Sorting results using `ORDER BY`.
- 1.5 Basic Data Manipulation
  - Inserting data with `INSERT`.
  - Updating data with `UPDATE`.
  - Deleting data with `DELETE`.
- Practice Questions: A set of **200+** exercises focused on querying and manipulating data in a simple database scenario.

Part 2: SQL Intermediate Lessons

- 2.1 Joins and Subqueries
  - Understanding different types of joins (`INNER JOIN`, `LEFT JOIN`, `RIGHT JOIN`, `FULL JOIN`).
  - Utilizing subqueries for complex data retrieval.
- 2.2 Grouping Data
  - Using `GROUP BY` to aggregate data.
  - Filtering aggregated data with `HAVING`.
- 2.3 Set Operations
  - Combining results using `UNION`, `INTERSECT`, and `EXCEPT`.
- 2.4 Working with Indexes
  - Introduction to indexes for performance optimization.
- 2.5 Data Definition Language (DDL)
  - Creating tables with `CREATE TABLE`.
  - Altering tables with `ALTER TABLE`.

    - Dropping tables with `DROP TABLE`.
- Practice Questions: Intermediate-level **200+** exercises that challenge learners to create more complex queries and understand the performance implications of various operations.

Part 3: SQL Advanced Lessons

- 3.1 Advanced Query Techniques
    - Window functions and their applications.
    - Common Table Expressions (CTEs) and recursive queries.
- 3.2 Database Design and Normalization
    - Basic concepts of database design and normalization to reduce redundancy and improve data integrity.
- 3.3 Transaction Control and Concurrency
    - Understanding transactions, `COMMIT`, and `ROLLBACK`.
    - Basics of concurrency control and locking mechanisms.
- 3.4 Performance Tuning and Optimization
    - Techniques for optimizing SQL queries for better performance.
- 3.5 Security and Permissions
    - Managing user access and roles with `GRANT` and `REVOKE`.
- Practice Questions: **200+** Advanced exercises that include optimizing query performance, implementing security measures, and designing efficient database schemas.

Part 4: Practice Questions

- Comprehensive Practice Set
    - A mixed set of **500+** practice questions covering beginner, intermediate, and advanced topics to solidify learners' understanding and prepare them for real-world SQL tasks.

Supplementary Materials

- Additional Resources
    - Recommended readings, online resources, and tools for practicing SQL beyond the classroom.
- SQL Best Practices
    - Guidelines for writing clean, efficient, and maintainable SQL code.
- Mock Interviews
    - **Unlimited** mock interviews with industry experts.

Python Tutorial Module

Part 1: Python Basics

- Introduction to Python programming language
- Variables, data types, and operators

Part 2: OOPs Concept

- Object-Oriented Programming (OOP)
- Classes, objects, inheritance, encapsulation, abstraction, polymorphism

Part 3: NumPy

- NumPy library
- Arrays and array operations
- Indexing, slicing, and reshaping arrays
- Mathematical functions in NumPy

Part 4: Pandas

- Pandas library for data analysis
- Series and DataFrame data structures
- Data cleaning and preprocessing with Pandas

Part 5: Data Visualization

- Introduction to data visualization
- Matplotlib library for basic plotting
- Seaborn library for statistical data visualization
- Plotly library for interactive visualizations

Part 6: File Handling

- Reading and writing files in Python
- Text file manipulation and processing
- File manipulation and file formats

Part 7: Exception Handling

- Introduction to exception handling
- Errors, exceptions, try-except blocks

- Handling specific exceptions

Part 8: Regular Expressions Fundamentals

- Introduction to regular expressions
- Pattern matching and searching in text
- Matching and replacing patterns

Explore Cloud Technologies (AWS)

AWS Basics

- Introduction to AWS (Amazon Web Services)
- Overview of cloud computing and its benefits
- Understanding AWS services and solutions
- Basics of AWS account setup and management

Amazon S3 (Simple Storage Service)

- Introduction to Amazon S3
- S3 bucket creation and management
- Uploading, downloading, and managing objects in S3
- Integrating S3 with other AWS services

Amazon Lambda

- Introduction to Amazon Lambda
- Explain serverless computing
- Creating Lambda Function
- Collecting, processing, and analyzing data with lambda

Amazon Kinesis

- Introduction to Amazon Kinesis
- Real-time streaming data processing
- Creating Kinesis data streams
- Collecting, processing, and analyzing streaming data

Amazon Firehose

- Introduction to Amazon Firehose
- Creating firehose
- Data processing with Firehose
- Firehose integration with AWS services

Amazon MSK (Managed Streaming for Apache Kafka)

- Introduction to Amazon MSK
- Apache Kafka basics
- Setting up and managing MSK clusters
- Streaming data ingestion and processing with MSK

AWS Glue

- Introduction to AWS Glue
- Data cataloging and metadata management
- Extract, Transform, Load (ETL) with Glue
- Building and managing ETL pipelines using Glue

DynamoDB

- Introduction to Amazon DynamoDB
- NoSQL database fundamentals
- Creating and managing DynamoDB tables
- Querying and scanning data in DynamoDB

AWS Redshift

- Introduction to AWS Redshift
- Columnar data warehousing with Redshift
- Provisioning and managing Redshift clusters
- Loading, querying, and optimizing data in Redshift

Amazon Athena

- Introduction to Amazon Athena
- Serverless querying and analysis of data
- Creating tables and querying data with Athena
- Optimizing performance and cost in Athena

Amazon Secrets Manager

- Introduction to AWS Secrets Manager

- Create and manage secrets
- Retrieve and rotate secrets
- Security in Secrets Manager

Amazon CloudWatch

- Introduction to Amazon Cloudwatch
- Setting up new Cloudwatch
- Check logs in Cloudwatch
- Setup alerts in Cloudwatch

Amazon SNS

- Introduction to Amazon SNS
- Setting up new SNS Topic
- Subscription to SNS Topics

Amazon SQS

- Introduction to Amazon SQS
- Setting up new SQS Queue
- Introduction to Dead Letter Queue (DLQ)
-

Amazon QuickSight

- Introduction to Amazon QuickSight
- Business intelligence and data visualization
- Creating visualizations and dashboards in QuickSight
- Sharing and presenting insights from QuickSight

Explore Cloud Technologies (Azure)

Introduction to Microsoft Azure

- Introduction to Microsoft Azure
- Introduction to ARM & Azure Storage
- Azure Virtual Machines
- Azure Networking – I

- Azure Networking – II

Authentication, Authorization, and Monitoring

- Authentication and Authorization in Azure using RBAC
- Microsoft Azure Active Directory
- Azure Monitoring

Data Storage and Integration

- Data Storage in Microsoft Azure
- Non-Relational Data Stores and Azure Data Lake Storage
- Data Lake and Azure Cosmos DB
- Relational Data Stores
- Why Azure SQL?
- Azure Data Lake Storage Gen2 and Data Streaming Solution
- Data Integration with Microsoft Azure Data Factory
- Designing Data Flows in Azure
- Using Azure Data Factory Pipelines to Copy Data
- Monitor Azure Data Factory using Azure Monitor

Azure Synapse Analytics and Databricks

- Introduction to Microsoft Azure Synapse Analytics
- Using Azure Synapse Analytics to Query Data Lake
- Optimizing Dedicated SQL Pools in Azure Synapse Analytics
- Data Warehousing with Microsoft Azure Synapse Analytics
- Data Engineering with MS Azure Synapse Apache Spark Pools
- Operational Analytics with Microsoft Azure Synapse Analytics
- Handling Slowly Changing Dimensions With Azure Synapse Analytics Pipelines
- Microsoft Azure Databricks for Data Engineering
- Running Spark on Azure Databricks
- Using Azure Databricks to Import and Analyze Data
- Introduction to Delta Lake on Azure Databricks

Azure Stream Analytics, and Azure Service Bus

- Introduction to Azure Stream Analytics
- Monitoring & Security
- Azure Functions
- Azure Service Bus

Explore Cloud Technologies (Snowflake)

Introduction to Snowflake

- **Overview of Snowflake Cloud Data Platform**
    - Introduction to Snowflake and its architecture.
    - Key features: Multi-cluster shared data architecture, separation of storage and compute.
    - Snowflake's position in the cloud data ecosystem.

Snowflake Data Warehousing

- **Setting Up Snowflake**
    - Creating a Snowflake account.
    - Overview of Snowflake editions (Standard, Enterprise, Business Critical).
    - Navigating the Snowflake user interface.
- **Virtual Warehouses**
    - Understanding virtual warehouses and scaling.
    - Best practices for performance and cost management.

Data Loading and Unloading

- **Loading Data into Snowflake**
    - Introduction to Snowflake's loading methods: Bulk loading with COPY, loading semi-structured data.
    - Working with external stages (AWS S3, Azure Blob, Google Cloud Storage).
- **Unloading Data**
    - Best practices for unloading data from Snowflake.
    - Using the COPY command to export data.

Snowflake SQL and Querying

- **Querying Data**
    - Snowflake SQL basics: SELECT statements, filtering, and sorting.
    - Working with semi-structured data (JSON, Parquet) using Snowflake.
- **Advanced Query Techniques**
    - Time travel and data retention in Snowflake.
    - Cloning and zero-copy cloning for efficient data management.

Security and Data Sharing

- **Snowflake Security Features**
    - Role-based access control (RBAC) and user management.
    - Data encryption and compliance features.

    - Managing security with Virtual Private Snowflake (VPS).
- **Data Sharing**
    - Introduction to Snowflake's secure data sharing capabilities.
    - Setting up and managing data shares with partners and customers.

Performance Tuning and Optimization

- **Best Practices for Query Performance**
    - Understanding query profiling in Snowflake.
    - Optimizing performance with clustering keys.
- **Resource Monitoring and Management**
    - Monitoring virtual warehouse performance and utilization.
    - Auto-suspend and auto-resume features for cost efficiency.

Snowflake Ecosystem and Integrations

- **Integrations with ETL Tools**
    - Connecting Snowflake with ETL tools like dbt, Apache Airflow.
- **Snowflake and BI Tools**
    - Integrating Snowflake with BI tools like Tableau, Power BI, and Looker.
- **Snowflake Marketplace**
    - Exploring the Snowflake Data Marketplace for third-party data.

Real-World Applications and Case Studies

- **Case Study: Building a Data Warehouse in Snowflake**
    - Designing and implementing a data warehouse solution.
    - Data ingestion, transformation, and reporting with Snowflake.
- **Real World Project: End-to-End Data Pipeline**
    - Developing a full data pipeline using Snowflake from data ingestion to analytics.

Spark

Introduction to Apache Spark

- Overview of big data processing and Apache Spark
- Spark architecture and components
- Introduction to Resilient Distributed Datasets (RDDs)
- Understanding Spark's distributed computing model

Spark SQL and Data Frames

- Introduction to Spark SQL module
- Working with structured and semi-structured data
- Data exploration and analysis using DataFrames
- Querying and manipulating data with SQL-like syntax

Apache Kafka and Flume

- Introduction to Apache Kafka and Apache Flume
- Streaming data ingestion using Kafka and Flume
- Integration of Kafka and Flume with Spark
- Real-time data processing and analysis

Spark Streaming

- Introduction to Spark Streaming
- Processing live data streams with Spark
- Windowed operations and aggregations
- Real-time analytics using Spark Streaming

Devops

Introduction to DevOps

- Understanding the DevOps culture and principles
- Benefits of DevOps in data engineering
- Overview of DevOps tools and practices

Git

- Introduction to version control systems
- Git fundamentals: repositories, branches, commits
- Collaborative development with Git
- Git workflows: branching strategies, pull requests, merging

Docker

- Introduction to containerization and Docker
- Docker architecture and components
- Building Docker images for data engineering applications
- Container orchestration with Docker Compose

Kubernetes

- Introduction to Kubernetes for container orchestration
- Kubernetes architecture and components
- Deploying and managing applications with Kubernetes
- Scaling, monitoring, and updating applications in Kubernetes

Jenkins

- Introduction to Jenkins for continuous integration and continuous delivery
- Jenkins installation and configuration
- Building and automating data engineering pipelines with Jenkins
- Integration with Git, Docker, and Kubernetes

Power BI

Introduction to Power BI

- Overview of Power BI and its role in data engineering
- Introduction to self-service business intelligence
- Understanding Power BI components: Power BI Desktop, Power BI Service, Power BI Mobile

Data Extraction

- Connecting to various data sources in Power BI
- Importing data from databases, files, web services, and other sources
- Configuring data refresh options and scheduling data updates

Data Transformation – Shaping & Combining Data

- Understanding data transformation concepts in Power BI
- Applying data shaping techniques: filtering, sorting, and removing duplicates
- Combining multiple data sources using merging and appending operations

Data Modeling & DAX (Data Analysis Expressions)

- Introduction to data modeling in Power BI
- Creating relationships between tables
- Implementing calculations and measures using DAX formulas

Data Visualization with Analytics

- Creating interactive visualizations using Power BI visuals
- Formatting and customizing visual elements
- Applying data analytics techniques: forecasting, clustering, and trend analysis

DBT(Data Build Tool)

DBT Cloud Overview

- Overview of DBT
- dbt, data platforms, and version control
- Setting up dbt Cloud and your data platform
- dbt Cloud IDE Overview
- Overview of dbt Cloud UI

Models

- What are models?
- Building your first model
- What is modularity?
- Modularity and the ref functions
- Quick history of data modeling
- Naming conventions
- Reorganize your project

Sources

- What are sources?
- Configure and select from sources
- Source freshness

Tests

- Why testing?
- What is testing?
- Generic tests
- Singular tests
- Testing sources
- The dbt Build command

Jinjas

- What is Jinja?
- Jinja Basics
- Jinja Applications

Macros

- What are macros?
- cents_to_dollars macro
- limit_data_in_dev macro
- DRY code vs. readability

Packages

- What are packages?
- Installing packages
- Packages with macros
- Packages with models

Materalizations

- What are materializations?
- Tables, views, and ephemeral models
- Incremental models
- What are snapshots?
- Implementing snapshots

Documentation

- Why is documentation important?
- What is documentation?
- Writing documentation and doc blocks
- Documenting sources
- Generate and view documentation

Deployment

- What is deployment?
- Setting up a dbt Cloud job

- Reviewing a dbt Cloud job

SQL FAANG Questions for Practice

- Duration: 2 Weeks
- Content:
    - Week 1: Introduction to SQL interview expectations at top tech companies. Practice with basic to intermediate SQL problems, focusing on data retrieval, aggregation, and filtering.
    - Week 2: Advanced SQL problem-solving involving joins, subqueries, window functions, and query optimization.

SQL Data Engineering Interview Prep

- Duration: 2 Weeks
- Content:
    - Week 1: Real-world SQL scenarios in data engineering, covering data modeling and ETL processes.
    - Week 2: Mock interviews simulating data engineering SQL interviews, including data warehouse and data lake querying.

Learning Outcomes

- Students will familiarize themselves with the types of SQL questions asked in FAANG and tech interviews.
- Gain practical experience with SQL through real-world scenarios relevant to data engineering.
- Develop strategies for solving complex SQL problems and optimizing queries.
- Experience mock interviews to improve interviewing skills with real-time feedback.

PySpark Module

Part 1: Introduction to PySpark

- **1.1 Introduction to PySpark**
  - Overview of PySpark and its role in big data processing.
  - Installation and setup of PySpark in different environments.
  - Understanding Spark architecture and cluster computing.
- **1.2 PySpark Core Components**
  - SparkContext, SparkSession, and their roles.
  - Introduction to RDD (Resilient Distributed Datasets) and DataFrames.
  - Key PySpark libraries and functions.

Part 2: Data Processing with RDDs

- **2.1 Introduction to RDDs**
  - Characteristics and structure of RDDs.
  - Creating and manipulating RDDs.
  - Transformations and actions in RDDs.
- **2.2 Advanced RDD Operations**
  - Aggregating data with reduce, aggregate, and fold.
  - Working with key-value RDDs (pair RDDs).
  - Using mapPartitions, flatMap, and coalesce for data partitioning.

Part 3: DataFrames and SQL in PySpark

- **3.1 Working with DataFrames**
  - Creating DataFrames from various data sources (CSV, JSON, Parquet).
  - Data exploration and filtering with DataFrame API.
  - Common DataFrame transformations (filter, select, groupBy).
- **3.2 SQL Queries on DataFrames**
  - Registering DataFrames as temporary tables.
  - Writing SQL queries in PySpark.
  - Combining SQL queries with DataFrame operations.

Part 4: Advanced DataFrame Operations

- **4.1 Data Cleaning and Preprocessing**
  - Handling missing values and duplicates.
  - Data type casting and string operations.
  - Applying user-defined functions (UDFs) in PySpark.

- **4.2 Joins and Aggregations**
    - Different types of joins (inner, outer, left, right).
    - Grouping, aggregating, and summarizing data.
    - Working with complex data structures (arrays, maps).

Part 5: Working with PySpark MLlib (Machine Learning Library)

- **5.1 Introduction to MLlib**
    - Overview of PySpark MLlib and its capabilities.
    - Building machine learning pipelines in PySpark.
- **5.2 Classification, Regression, and Clustering**
    - Implementing basic machine learning models (logistic regression, decision trees).
    - Exploratory data analysis and feature engineering.
    - Unsupervised learning with clustering algorithms (K-means).

Part 6: Structured Streaming in PySpark

- **6.1 Introduction to PySpark Structured Streaming**
    - Overview of streaming data and its importance.
    - Setting up and managing structured streams in PySpark.
- **6.2 Working with Real-time Data**
    - Reading and writing streaming data.
    - Aggregating, filtering, and transforming streaming data.
    - Applying window operations and handling late data.

Part 7: Performance Tuning and Optimization

- **7.1 Optimizing PySpark Jobs**
    - Partitioning strategies and best practices.
    - Caching and persisting RDDs and DataFrames.
    - Using broadcast variables and accumulators.
- **7.2 Advanced Optimizations**
    - Understanding the Catalyst Optimizer.
    - Working with Spark's Tungsten execution engine.
    - Avoiding shuffle operations and minimizing data movement.

Part 8: Deployment and Best Practices

- **8.1 Deploying PySpark Applications**
    - Deploying PySpark on various environments (local, cluster, cloud).

    - Setting up Spark configurations for optimal performance.
    - Managing resource allocation and monitoring PySpark jobs.
- **8.2 Best Practices in PySpark Development**
    - Writing efficient and maintainable PySpark code.
    - Debugging and troubleshooting common PySpark errors.
    - Version control and code modularity.

Practice and Project Work

- **Practice Exercises**:
    - A set of 300+ exercises across beginner, intermediate, and advanced levels covering key PySpark operations, transformations, and machine learning tasks.
- **Capstone Project**:
    - **End-to-End Data Pipeline**: Students will build an end-to-end data pipeline involving data ingestion, transformation, and analysis using PySpark.
    - **Real-Time Data Processing Project**: Setting up a structured streaming pipeline to process real-time data.

Evaluation

- **Mock Interviews**:
    - Weekly mock interviews to assess comprehension and problem-solving in PySpark.
- **Peer-reviewed Assignments**:
    - Practical assessments where peers review PySpark code for efficiency, scalability, and clarity.
- **Final Project Presentation**:
    - A presentation and report on the capstone project, demonstrating the implementation of learned PySpark concepts in a real-world scenario.