12 STEPS OF DATA ENGINEERING

End-to-End GCP Data Engineering Activities

1. Source Data Understanding & Access

- Identify the **type of source**: Data warehouse, Databases, APIs, IoT, Cloud, File systems, CDC.
- Determine the **server type**: Unix, Windows.
- **Location**: UK, IN, etc.
- **Connection details**: Data location, database name (e.g., sales, products, payments).
- **Sample data**: CSV, JSON, XML, or other formats.
- **Dataset details**:
  - File naming format (e.g., bmw_sales_uk_data_09-12-2024.csv).
  - File size/number of records per dataset.
- **Schema details**: Number of tables, schema structure.
- **Data frequency**: Daily, hourly, weekly, or near real-time (NRT).
- **Historical data load process**.

2. Choose Your Tools

- **Landing zone**: Shared drive, cloud storage.
- **Connection & access**:
  - **Access permissions** (e.g., 777).
  - **Tools**: PuTTY, WinSCP, Python, Git, VS Code, IntelliJ, Notepad++.
  - **Third-party tools**: NiFi, LeapLogic, DataDog, Juniper, Talend, Make, etc.
- **Cloud account & project setup**:
  - Google Cloud Storage (GCS).
  - BigQuery.
  - Cloud Composer (Airflow).
  - DataFlow, Pub/Sub.
- **CDC tools**: IBM Infosphere, Attunity, DebeziumDB.
- **Mainframe**: V-Series, I-Series, EBCDIC.
- **Messaging systems**: RabbitMQ, JMS, Kafka, Confluent, Pub/Sub.

3. Setup Environment

- **Raise requests for tool access**.
- **Set up**:
  - SDK environment.
  - Git repository.
  - On-prem servers access, IAM roles.
  - Service accounts & API enablement.
  - Snow/Jira ticketing system for tracking requests.
  - Confluence projects for documentation.
  - Sprint definition, POD setup.
  - MS Office tools (Word, Excel, PPT).
  - VPN, VPC, security tools.
- **Development environment configurations**.

4. Develop Your Data Pipeline

- **Storage setup**: SDK, buckets, objects.
- **BigQuery (BQ) setup**: Datasets, tables, queries.
- **Develop scripts**:
  - **Data load scripts**: BQ SDK (-t, -d, -v, etc.).
  - **Data recovery**: Time travel, snapshot, failsafe.
  - **Security policies**: Row/column-level access, audit tables, views.
  - **DAG scripts**: Airflow-based ETL orchestration.
  - **Dataflow jobs**: Beam templates, Python/Java jobs.
  - **Dataproc**: PySpark scripts & jobs.
  - **DWH concepts**: Star schema, Snowflake schema, Slowly Changing Dimensions (SCD).

5. Data Transformations

1. **Data Cleaning**:
   - Handle null values, regex validation, remove invalid/missing data.
   - Default value corrections (trim, upper/lowercase, replace, COALESCE).
2. **Data Conversion**:

   - Typecasting, safe cast, parsing dates/timestamps.

3. **Data Aggregation**:
   - Grouping, count, sum, avg, min, max.
4. **Data Filtering**:
   - WHERE, HAVING, LIKE, BETWEEN, <, >, !=.
5. **Data Joining**:
   - Joins between multiple tables.
6. **Partitioning for Cost Optimization**:
   - **By Date/Timestamp**, **Integer Range**.
   - Techniques: Avoid caching, reduce subqueries, optimize joins, minimize temp tables, create multiple tables/views.
7. **Data Windowing Functions**:
   - COUNT OVER(), RANK, DENSE_RANK, ROW_NUMBER, NTILE().
8. **Data Sharding**:
   - Distribute data across multiple tables for scalability.

6. Data Loading

- **Load data into BigQuery (GCS → BQ)**
  - Full load, truncate load, incremental/delta load.
  - Update & insert (upsert) using MERGE.
- **On-prem DWH to BigQuery migration**
  - Direct load via Python/Scala scripts.
  - DBT for transformation & loading.
- **Cloud-to-cloud migration (C2C)**
  - Example: Redshift → BigQuery.

7. Test Your Data Pipeline (Unit Testing / SIT)

- Validate each pipeline step manually.
- **Follow 21-step testing checklist**.
- Deploy scripts from Git to **Dev** and **SIT environments**.

8. Monitor Data Pipeline

- Use **Composer, Cloud Scheduler, Cloud Run, Stackdriver** for monitoring.

- **Check DAG details**:
    - Job status, task status, task dependencies.
- **Analyze logs** to detect errors & challenges.

9. Optimize Your Data Pipeline

- **Query optimization**: Improve joins, grouping, query execution plans.
- **Data optimization**: Partitioning, clustering.
- **Connection validation**.
- **Data Governance**:
    - Implement policies for masking, quality checks, data profiling, data fabric.

10. Deployment Process

- **Change request (CR) process**:
    - Starts **2 weeks before deployment**.
    - Deployment window: **3-5 days**.
- **Coordinate with all teams**:
    - Engineering, IT, Platform, Source, Governance, DevOps (CI/CD).

10A. Post-Production Validation

- **Validate source vs. target** using BigQuery history & views.

11. Handover to Business

- **Deliverables**:
    - Verify **DIS Sheet**, ensure product owner requirements align with authorized views.
    - **Prepare documentation** & submit to business stakeholders.
    - Send **project delivery mail** to update all business teams.

12. Production Support & Knowledge Transfer (KT)

- **Handover key documentation**:
    - Confluence pages.
    - Git scripts.
    - Dev process & testing methodologies.
- **Discuss common issues** & resolution approaches.
- **Reverse KT (Knowledge Transfer)** for future teams.