SQL BASICS

* What is SQL ?

ans) SQL (Structured Query Language) is used to store, manipulate, and retreive data in relational databases like MySQL, Postgre SQL, SQLite, Oracle.

* Create Database :

suppose we're using a School database.

```
CREATE DATABASE School;
```

* Use Database :

```
USE school;
```

* Create Table :

```
CREATE TABLE Students (
id INT PRIMARY KEY,
name VARCHAR (50) NOT NULL,
age INT,
grade VARCHAR (5)
);
```

CRUD functions (Create, Read, Update, Delete)

* Insert Data (CREATE)

```
- INSERT INTO Students Values (1, 'Alice', 20, 'A');
```

* Read Data (SELECT)

```
- SELECT * FROM STUDENTS;
- SELECT name, age FROM STUDENTS;
```

* Update Data (UPDATE)

```
- UPDATE Students
  SET age = 21
  WHERE id = 1;
```

* Delete Data (DELETE)

```
DELETE FROM Students
WHERE id = 1;
```

→ WHERE Clause:

```
SELECT * FROM Students
WHERE age > 18;
```

* ORDER BY Clause

s/x =>

```
SELECT * FROM table_name
ORDER BY column_name ASC / DESC
```

Sort table by column in a sorted order.

* LIMIT Clause

s/x ;

```
SELECT * FROM table_name
LIMIT number;
```

shows the numbers up to a limit.

* ADD a column

s/x =>

```
ALTER TABLE table_name
ADD column_name datatype;
```

*. Delete a column

s/x =>

```
ALTER TABLE table_name
DROP column_name COLUMN column_name;
```

* Constraints (Rules on Columns)

- → PRIMARY KEY - Unique ID for each row
- → FOREIGN KEY - Link to primary key in another table.
- → NOT NULL - Column cannot be empty.
- → UNIQUE - Values in column must be unique.
- → CHECK - Validates values based on condition
- → DEFAULT - Assigns a default value if none provided.

* AGGREGATE FUNCTIONS

- → COUNT() - count rows
- → SUM() - Sum values
- → AVG() - Average value
- → MAX() - Highest value
- → MIN() - Lowest value

* Grouping & Filtering Groups

1) GROUP BY clause:

s/x:

```
SELECT column, COUNT(*)
```

FROM table_name

GROUP BY column ;

2) HAVING clause :

s/n =>

Eg : SELECT grade, COUNT(*)

FROM Students

GROUP BY grade

HAVING COUNT(*) > 2 ;

*. JOINS - Combines Data from Tables ;

1) INNER JOIN

SELECT A.name , B. course_name

FROM Students A

INNER JOIN Courses B ON A.id = B. student_id ;

2) LEFT JOIN

SELECT A. name, B. course_name

FROM Students A

LEFT JOIN Courses B on A.id = B.student_id

Data Ingestion

- Is the process of gathering, managing & utilizing data efficiently.
- plays a foundational step in the data processing pipeline.
- It involves the seamless importation, transfer or loading of raw data from diverse external resources into a centralized system or storage infrastructure, where it awaits further processing & analysis.

- Key steps in the Ingestion Process.

1. Data Collection - Gather raw data from various sources.
2. Data Transformation - Clean, normalise & enrich the data.
3. Data loading - moves the transformed data into the target system,

There are three type of Ingestion:

Type	Description	Use Cases
Batch	Data is collected & processed at scheduled intervals.	Daily reports, Pay roll, backups.
Real-Time	Data is ingested as its generated, enabling Instant processing	Fraud detection, Live Dashboards.
Micro-batching	a hybrid approach small batches processed frequently.	IOT data, Semi-live analytics.

Data Acquisition

→ what is data acquisition?

ans) The process of collecting, measuring & storing data from various sources (sensors, devices, websites, databases, APIs, etc) into a usable format for further analysis.

→ Types of Data Acquisition;

- Manual Acquisition;
  Hand-typed, manually collected (Surveys, Spreadsheets)
- Automated Acquisition;
  Programmatically extracted (web scraping, sensors, APIs)
- Real-Time Acquisition;
  continously streaming data (IOT sensors)
- Batch Acquisition;
  collected at fixed intervals (Data Dumps, log files).

→ Data Acquisition Process:

1) Identify Data Source - know where your data coming from.

2) Connect to the source - Use APIs, file readers or sensors

3) Collect the data - pull the data using tools or scripts.

4) pre-process - clean, convert, normalize the data.

5) Store - Save to database, file or cloud Storage.

*. ELT & ETL

In managing and analysing data; two primary approaches ie, ETL (Extract, Transform, Load) and ELT (Extract, Load and Transform) are commonly used to move data from various sources into a data warehouse.

→ ELT Process

Extraction, Load and Transform (ELT) is the technique of extracting raw data from the source.

storing it in the data warehouse of the target server and preparing it for end-stream user.

ELT operations;

1) Extract;

Extracting data is the process of identifying data from one or more sources. The sources may include databases, files, ERP, CRM, or any other useful source of data.

2) Load;

Loading is the process of storing the extracted raw data in a warehouse or data lake.

3) Transform;

Data Transformation is the process in which the raw data from the source is transformed into the target format required for analysis.

ETL Process

In this process there are three operations.

1) Extract;

It is the process of extracting raw data from all available data sources such as databases, files, ERP, CRM or any other.

2) Transform;

The extracted data is immediately transformed as required by the user.

3). Load;

The transformed data is then loaded into the data warehouse from the users can access it.

Data Warehousing concepts

→ What is Data Warehousing

ons) Data warehousing is a central repository of integrated data from multiple sources.

It supports reporting, analysis, business intelligence, decision making.

- optimised for read heavy operations.
- Stores historical data
- Used in OLAP Systems.

→ Benefits of Data Ware Housing;

- combines data from multiple sources.
- Supports historical Analysis.
- Improves data quality & consistency.
- Enables faster decision-making.

→ How OLTP and OLAP Work Together in Data Warehousing?

1). OLTP ;

collects daily transactional data.

2). ETL ;

Data is extracted from OLTP systems, transformed (cleaned & structured) and loaded into the warehouse.

3). Data Warehouse ;

stores large volumes of historical data.

4) OLAP ;

performs multi dimensional queries on this stored data for analysis and reporting.

*. What is OLTP?

ans) OLTP (Online Transaction Processing) systems are used for real time, day to day transaction processing.

characteristics:

- fast insert/update/delete operations.
- Handles high volume of short transactions.
- used by customers, clerks, employees.
- Data is current and highly normalised.
- Ensures data integrity

Eg: Banking System, e-commerce, etc.

* what is OLAP?

ans) OLAP (Online Analytical processing) systems are used for complex queries, analysis and reporting.

→ characteristics:

- used for decision-making and trend analysis.
- involves read heavy operations.
- Queries are often complex and multi-dimensional
- stores historical data
- uses denormalised schema (star/snow-flake).

* Star Schema & Snow flake Schema

→ Star Schema ;

Star schema is a denormalised structure in which a central fact table is connected directly to the dimension tables.

→ Structure ;

- Looks like a Star
- One fact table in the center
- Dimension table radiates around it

→ Fact Table ;

- stores measurable data
- contains foreign keys pointing to dimension tables.

→ Dimension tables ;

- Store descriptive attributes
- Not normalised (data may repeat)

→ Performance ;

- Faster querying due to fewer joins.

. ideal for reporting and dashboards.

Storage ;

. More storage used ~~in~~ the due to data redundancy.

. Easier to end-users to understand.

Use Case ;

Suitable for simple data models with high query performance needs.

→ Snowflake Schema ;

Snowflake schema is a normalised structure where dimension tables are further split in to sub dimensions.

Structure ;

. Looks like Snow flake

. Fact table at the center

. Dimension tables are normalised into related tables

3 Fact table;

- Same as star schema
- Stores metrics & foreign keys.

→ Dimension Tables;

- Linked to other related dimension Tables.
- Normalised to reduce redundancy

→ Performance;

- slower querying due to more joins.
- requires complex SQL Queries.

→ Storage;

- Less storage needed due to normalisation
- Better data Integrity.

→ Use Cases;

- suitable for complex data models with need for data consistency.

* Dimensional Modelling :

Dimensional Modelling is a design concept used in data warehouses to make data easy to understand and query.

It consists of ;

→ A. fact Table ;

- contains numerical measures.
- Foreign keys to dimensional tables.

B. Dimension Table;

- Descriptive attributes used to slice and dice data
- Textual, categorical data.