01 Introduction to SQL and Relational Databases

Relational Databases

- A **relational database** is a type of database that stores data in a structured format, using rows and columns
- Each table in a relational database represents a specific entity (like customers, orders, products), and each row in a table represents a unique record
- The relationship between these tables is established through **keys** (Primary and Foreign keys)

Key Concepts in Relational Databases

- **Table**
  - The primary structure in a relational database, consisting of rows and columns
  - Each table stores data about a specific entity
- **Row (Record)**
  - A single, data-filled instance in a table
  - Each row represents a unique item or entity
- **Column (Field)**
  - A vertical entity in a table that contains all information associated with a specific field (like Name, Age, Address).
- **Primary Key**
  - A unique identifier for each record in a table
  - It ensures that each record can be uniquely identified
- **Foreign Key**
  - A field in a table that links to the primary key of another table, establishing a relationship between the two tables
- **Normalization**
  - The process of organizing data to minimize redundancy
  - It involves dividing large tables into smaller, related tables and using foreign keys to establish relationships
- **Index**
  - A database object that improves the speed of data retrieval operations
  - It allows the database to find rows more quickly without scanning the entire table

Advantages of Relational Databases

- **Data Integrity** : Through the use of primary and foreign keys, relational databases maintain accurate and consistent data
- **Flexibility** : Users can query, filter, and manipulate data in various ways
- **Scalability** : Suitable for both small and large datasets
- **Security** : Provides mechanisms to control access to data

SQL

- **SQL (Structured Query Language)** is a standardized programming language specifically designed for managing and manipulating relational databases
- It allows users to query, insert, update, and delete data within a database
- SQL is widely used in applications ranging from small-scale desktop software to large-scale enterprise systems

SQL Basics

1. **SELECT Statement**: Used to query data from one or more tables

```
SELECT column1, column2 FROM table_name WHERE condition;
```

2. **INSERT INTO Statement**: Used to insert new records into a table

```
INSERT INTO table_name (column1, column2) VALUES (value1, value2);
```

3. **UPDATE Statement**: Used to modify existing records in a table

```
UPDATE table_name SET column1 = value1 WHERE condition;
```

4. **DELETE Statement**: Used to delete records from a table

```
DELETE FROM table_name WHERE condition;
```

5. **JOINs**: Used to combine rows from two or more tables based on a related column between them
    - **INNER JOIN**: Returns records that have matching values in both tables
    - **LEFT JOIN**: Returns all records from the left table, and the matched records from the right table
    - **RIGHT JOIN**: Returns all records from the right table, and the matched records from the left table

- **FULL JOIN**: Returns all records when there is a match in either left or right table

6. **GROUP BY Statement**: Used to arrange identical data into groups

```
SELECT column, COUNT(*)
FROM table_name
GROUP BY column;
```

7. **ORDER BY Statement**: Used to sort the result-set in ascending or descending order

```
SELECT column1, column2 FROM table_name ORDER BY column1 ASC;
```

8. **Constraints**: Rules enforced on data columns to ensure the accuracy and reliability of the data within the database. Examples include `NOT NULL`, `UNIQUE`, `CHECK`, `DEFAULT`