Basic Database Concepts: Key Terms and Principles in Database Systems

1. Introduction

This lesson introduces the essential terminology and foundational principles of database systems. To effectively work with databases, it's important to understand the language and concepts that define their structure and function. This section will break down key terms like schemas, queries, and keys, which form the building blocks of any database.

---

2. Core Database Components

Every database system is made up of a few fundamental components that work together.

- **Data**: The actual facts, figures, and information stored in the database.
- **Hardware**: The physical devices, like servers and hard drives, that store the data.
- **Software**: The **Database Management System (DBMS)** is the software that allows you to create, manage, and use the database. Examples include MySQL, Oracle, and Microsoft SQL Server.
- **Procedures**: The rules and instructions that govern how the data is used and managed, such as data entry and backup protocols.
- **Database Access Language**: The language used to communicate with the DBMS, with **SQL** (Structured Query Language) being the most common.

---

3. Key Database Terminology

Schema vs. Instance

- **Schema**: The **blueprint** or **structure** of the database. It defines the tables, the columns in each table, the data types they can hold, and the relationships between them. A schema is a logical design that rarely changes.
- **Instance:** The actual data stored in the database at a specific moment in time. The instance is a **snapshot** of the data that exists at any given second. An instance is dynamic and changes constantly as data is added, deleted, or modified. Think of a schema as the empty form, and the instance as the filled-out form.

Tables, Rows, and Columns

These terms are used in relational databases to describe how data is organized.

- **Table**: A collection of related data organized in a two-dimensional grid of rows and columns.

- **Row** (also called a **record** or **tuple**): A single entry or a complete set of related values within a table. For example, in a "Customers" table, a single row would represent one customer.
- **Column** (also called a **field** or **attribute**): Represents a specific type of data within a table. For example, in a "Customers" table, you might have columns for "Customer ID," "Name," and "Email."

Keys

Keys are used to uniquely identify records and establish relationships between tables.

- **Primary Key**: A column or a set of columns in a table that uniquely identifies each row. A primary key must be **unique** and **cannot be null**. You can think of it as a person's Social Security Number; no two people have the same one.
- **Foreign Key**: A column in one table that refers to the **primary key** in another table. Foreign keys create links between tables, which is the foundation of the relational database model. For instance, an "Orders" table might have a foreign key called "CustomerID" that links each order to the correct customer in the "Customers" table.

Query

A **query** is a **request for data** from a database. Using a query language like SQL, you can ask the database to perform specific actions, such as:

- **Retrieving** data (e.g., "Find all customers who live in Olongapo City").
- **Adding** new data.
- **Updating** existing data.
- **Deleting** data.

Queries are the primary way users and applications interact with a database to get information or make changes.