Database and SQL Fundamental

Press Space for next page →

Agenda

1. Data VS Information
2. Database
3. Examples of Databases
4. Common Database Types
5. Components of a Databases
6. Common types of keys in a database
7. Download and Install MySQL
8. Execute SQL Queries
9. Data Types in MySQL
10. SQL Commands
11. Create a Table in MySQL
12. Insert Data into a Table
13. Select Data from a Table

Data VS Information

DATA	INFORMATION
Data refers to raw facts that have no specific meaning	Information refers to processed data that has a purpose and meaning
The data is independent of the information	Information is dependent on data
Data or raw data is not enough to make a decision	The information is sufficient to help make a decision in the respective context

Database

> A database is a collection of organized data, information, and records that can be easily accessed and managed. It is a structured repository that stores data in a way that allows for efficient retrieval and manipulation.

Key Characteristics:

Organized Data: Data is stored in a structured format, making it easy to access and manage.

Collection of Records: A database can contain multiple records, each with its own set of attributes.

Easy Access and Management: Data can be easily retrieved, updated, and manipulated using database management systems (DBMS).

Examples of Databases

- **Personal Database::** A list of phone numbers, addresses, or personal contacts.
- **Business Database:** A collection of customer information, sales records, or inventory data.
- **Educational Database:** A repository of student records, course information, or exam results.

Common Database Types

- **Relational Databases:** Organizes data into tables with rows and columns.

  Example: MySQL, PostgreSQL, Oracle.
- **NoSQL Databases:** Stores data in a non-tabular format, such as key-value pairs or document stores.

  Example: MongoDB, Cassandra, Redis.
- **Key-Value Database:** Stores data as key-value pairs.

  Example: Redis, DynamoDB.

Components of a Databases

- **Tables:** A collection of related data, organized into rows and columns.
- **Record: :** A single entry in a table, representing a single instance of data.
- **Fields:** A single column in a table, containing a specific piece of data for each record.

ROLL NO	NAME	AGE	GPA
101	Alice	20	3.5
102	Bob	21	3.2

Common types of keys in a database

Primary Key: A unique identifier for each record in a table.

- Uniquely identifies each record.
- No duplicate values or NULL values allowed.
- Typically indexed for faster search and retrieval.
- Example: Employee ID in an Employees table.

Foreign Key: A field in a table that refers to the primary key of another table.

- Establishes relationships between tables.
- Can contain NULL values and duplicate values.
- Used to link data across tables.
- Example: Order ID in an Orders table referencing the Customer ID in a Customers table.

Download and Install MySQL

- Download XAMPP from: apachefriends.org
- Install XAMPP on your computer.
- Start the Apache and MySQL services.
- Open phpMyAdmin in your browser: http://localhost/phpmyadmin/

Execute SQL Queries

- Open phpMyAdmin in your browser.
- Click on the SQL tab.
- `SHOW DATABASES;` to list all databases.
- `CREATE DATABASE mydatabase;` to create a new database.
- `USE mydatabase;` to switch to the new database.
- `DROP DATABASE mydatabase;` to delete the database.

Data Types in MySQL

Numeric Data Types: INT, FLOAT, DOUBLE, DECIMAL.

- ***INT:*** Integer values.

```
CREATE TABLE employees (id INT PRIMARY KEY, salary INT);
```

- ***DECIMAL:*** Fixed-point numbers.

```
CREATE TABLE products (price DECIMAL(10, 2));
```

- ***FLOAT:*** An approximate floating-point number.

```
CREATE TABLE products (price FLOAT);
```

Date and Time Data Types: DATE, TIME, DATETIME.

- ***DATE:*** Date values. Example format 'YYYY-MM-DD'

```
CREATE TABLE employees (dob DATE);
```

- ***DATETIME:*** Date and time values. Example format 'YYYY-MM-DD HH:MM:SS'

```
CREATE TABLE employees (created_at DATETIME);
```

String Data Types: CHAR, VARCHAR, TEXT.

- ***VARCHAR:*** Variable-length string.

```
CREATE TABLE users (username VARCHAR(50), email VARCHAR(100));
```

SQL Commands

DDL (Data Definition Language)

- ***CREATE TABLE:*** Creates a new database, table, index, or view
- ***ALTER TABLE:*** Modifies an existing database, table, or view
- ***DROP TABLE:*** Deletes a database, table, index, or view
- ***TRUNCATE TABLE:*** Removes all records from a table, but keeps the table structure

DML (Data Manipulation Language):

Example: INSERT, UPDATE, DELETE

DQL (Data Query Language):

Example: SELECT, SELECT DISTINCT, SELECT INTO

Create a Table in MySQL

```sql
CREATE TABLE students (
  id INT PRIMARY KEY,
  name VARCHAR(50) NOT NULL,
  class INT,
  section CHAR(1),
  fees INT,
  house VARCHAR(20)
);
```

Insert Data into a Table

```
INSERT INTO students (id, name, class, section, fees, house) VALUES
(101, 'Alice', 10, 'A', 5000, 'Red'),
(102, 'Bob', 11, 'B', 6000, 'Blue'),
(103, 'Charlie', 12, 'C', 7000, 'Green');
```

Rename a Table and Drop a Table

```
RENAME TABLE students TO new_students;
DROP TABLE new_students;
```

Select Data from a Table

Example 1: Select Specific Columns

```
SELECT name, class, fees FROM students;
```

This query selects the name, class, and fees columns from the students table.

Example 2: Select All Columns

```
SELECT * FROM students;
```

This query selects all columns from the students table.

Example 3: Select with WHERE Clause

```
SELECT * FROM students WHERE class = 10;
```

This query selects all columns from the students table where the class is 10.

Example 4: Select with AND Operator

```
SELECT * FROM students WHERE class = 10 AND section = 'A';
```

This query selects all columns from the students table where the class is 10 and the section is 'A'.

Thank you ❤️

qa-june-2024-automation-with-java-slides