UNDERSTANDINGS & CONCLUSIONS

NANDINI BHATTACHARYA

MODULE – 7

28. Database

28.1 MYSQL

- A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.
- Nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called **relational database** because all the data is stored into different tables and relations are established using **primary keys** or other keys known as **Foreign Keys**.

- **MySQL** is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database.
- MySQL is open-source and free software.
- MySQL is supported by Oracle Company.
- MySQL is ideal for both small and large applications
- MySQL is very fast, reliable, scalable, and easy to use
- MySQL is cross-platform

- MySQL follows the working of Client-Server Architecture. This model is designed for the end-users called clients to access the resources from a central computer known as a server using network services.
- The core of the MySQL database is the MySQL Server. This server is available as a separate program and responsible for handling all the database instructions, statements, or commands.
- The working of MySQL database with MySQL Server are as follows:

  1. MySQL creates a database that allows you to build many tables to store and manipulate data and defining the relationship between each table.
  2. Clients make requests through the GUI screen or command prompt by using specific SQL expressions on MySQL.
  3. Finally, the server application will respond with the requested expressions and produce the desired result on the client-side.

- **MySQL Data Types –**
- **1. Numerical type –** INT, FLOAT(m,d),DOUBLE(m,d),DECIMAL(m,d),BOOL
- **2. Date and Time type –** DATE, TIME,DATETIME
- **3. String type –** CHAR(size), VARCHAR(size)

- SQL keywords are NOT case sensitive: select is the same as SELECT
- Semicolon is the standard way to separate each SQL statement in database systems that allow more than one SQL statement to be executed in the same call to the server.

- **Commands of MySQL –**

DESCRIPTION	COMMAND
Create a database	CREATE DATABASE database_name;
Select a database	USE database_name;
List the databases	SHOW DATABASES;
Drop/delete a database	DROP DATABASE database_name;
Create a table in a database	CREATE TABLE table_name(column_definition1, column_definition2,.....,);

List the tables	SHOW TABLES;
Rename table	RENAME old_name TO new_name;
Truncate table (date without the structure)	TRUNCATE TABLE table_name;
Drop table (data with structure)	DROP TABLE table_name;
Add data into a table	INSERT INTO table_name VALUES(field1,field2..);
Modify data of table	UPDATE table_name SET column_name1 = new_value1 WHERE [condition];
Delete row from table	DELETE FROM table_name WHERE [condition];
Clause to remove duplicate records and fetch only unique records from table	SELECT DISTINCT value FROM table_name WHERE condition;
Clause to sort the records in ascending or descending order	SELECT value FROM table_name WHERE condition ORDER BY value [desc,asc];
Clause to collect data from multiple records and group the result by one or morecolumn	SELECT value1, value2 Aggregate_func(value) FROM table_name WHERE condition GROUP BY value1, value2;
JOIN : to fetch data from multiple table [INNER|LEFT|RIGHT]	SELECT table1.column1,table1.column2,table2.column1 FROM table1 [INNER|LEFT|RIGHT]JOIN table2 ON table1.column=table2.column;

- **Use and Connect to MySQL Database in .NET Application using MySQLConnector-**
- **Step-1 :** Download and install the MySQLConnector in order to connect to the MySQL database in .Net.
- **Step-2** : Create a database
- **Step-3:** Create a project in visual studio
- **Step-4 :** Now Create the new website and save it
- **Step -5:** Now open the .aspx form and drag some Labels, text boxes and button.
- **Step-6:** Now create the connection

28.2 Workbench Overview

- MySQL Workbench is a unified visual database designing or graphical user interface tool used for working with database.
- It is developed and maintained by Oracle.
- It is available for all major operating systems like Mac OS, Windows, and Linux.
- MySQL Workbench fully supports MySQL Server version v5.6 and higher.
- It provides SQL development, data modeling, data migration, and comprehensive administration tools for server configuration, user administration, backup, and many more.

- MySQL Workbench covers **five main functionalities**:

  1. SQL Development: This functionality provides the capability that enables you to execute SQL queries, create and manage connections to the database Servers with the help of built-in SQL editor.

  2. Data Modelling (Design): This functionality provides the capability that enables you to create models of the database Schema graphically, performs reverse and forward engineering between a Schema and a live database, and edit all aspects of the database using the comprehensive Table editor.

  3. Server Administration: This functionality enables you to administer MySQL Server instances by administering users, inspecting audit data, viewing database health, performing backup and recovery, and monitoring the performance of MySQL Server.

4. Data Migration: This functionality allows you to migrate from Microsoft SQL Server, SQLite, Microsoft Access, PostgreSQL, Sybase ASE, SQL Anywhere, and other RDBMS tables, objects, and data to MySQL. It also supports migrating from the previous versions of MySQL to the latest releases.

- **MySQL Enterprise Supports:** This functionality gives the support for Enterprise products such as MySQL firewall, MySQL Enterprise Backup, and MySQL Audit.

- MySQL Workbench is mainly available in **three editions**:

  1. The **Community Edition** is an open-source and freely downloadable version of the most popular database system. It came under the GPL license.

  2. Standard Edition is the commercial edition that provides the capability to deliver high-performance and scalable Online Transaction Processing (OLTP) applications.

  3. Enterprise Edition is the commercial edition that includes a set of advanced features, management tools, and technical support to achieve the highest scalability, security, reliability, and uptime. This edition also reduces the risk, cost, complexity in the development, deployment, and managing MySQL applications.

- **Create Database:** Go to the Navigation tab and click on the Schema menu. Here, you can see all the previously created databases.
- For new database, right-click under the Schema menu and select **Create Schema** or click the database icon (red rectangle).
- The new Schema window screen open. Enter the new database name and use default **Collation**.
- A new popup window appears, click Apply->Finish button to create a new database.
- If you want to see more information about the database, select mytestdb database, and click on the 'i' icon. The information window displays several options, like Table, Column, Functions, Users, and many more
-
- **Drop Database:** Right-click database->click on drop schema->drop schema
-
- **Export Database:** go to the Menu bar, click on Server, and select the Data Export option.It will open a new window of **data export** settings and options.
- In the Export option, you can select the export path of your choice,
- Click the Start Export button, which displays the progress bar and log. Now, open the Document folder in your system to locate the export files.

- **Import Databases:** Go to the Menu bar, click on Server, and select the Data Import option. It will open a new window of data import settings and options.
- Here, you can see the two radio options to import databases and tables, which are:
  Import from Dump Project Folder
  Import by using Self-Contained File
- Select a database from the Data Import option and also select the corresponding product table.

28.3 CRUD Operation

- CRUD stands create, read, update and delete operations.
- We can perform basic CRUD operation on the database with some HTTP method as shown
- **Create**

  CREATE DATABASE database_name;

  CREATE TABLE table_name(column1 datatype(size), column2 datatype(size),....);

  INSERT INTO table_name VALUES(value1,value2...);
- **Read**

SELECT*FROM table_name;

SELECT FROM table_name WHERE condition;

- **Update**

  UPDATE table_name SET column1=value WHERE condition;

- **Delete**

  DELETE FROM table_name WHERE condition;