WeHelp

Assignment - Week 5

For a full functional service, we have to build a database server as a data storage solution. Install MySQL server in your computer and use SQL statements to manage data in MySQL.

Task 1: Install MySQL server

Download and install MySQL community server 8.x from the official website. Don't forget your username and password setup in the installation procedure.

Download Link: https://dev.mysql.com/downloads/mysql/

Installation Tutorial: Windows、Mac

Task 2: Create database and table in your MySQL server

Connect to your MySQL server through Command Line Interface. Use SQL statements to complete requests described below:

- Create a new database named website.
- Create a new table named member, in the website database, designed as below:

Column Name	Data Type	Additional Settings	Description
id	bigint	primary key, auto increment	Unique ID
name	varchar(255)	not null	Name
username	varchar(255)	not null	Username
password	varchar(255)	not null	Password
follower_count	int unsigned	not null, default to 0	Follower Count
time	datetime	not null, default to current time	Signup Time

Task 3: SQL CRUD

Write SQL statements to complete following requests based on the database and table created in Task 2. **Reference:** W3Schools SQL Tutorial

- INSERT a new row to the member table where name, username and password must be set to test. INSERT additional 4 rows with arbitrary data.
- SELECT all rows from the member table.
- SELECT all rows from the member table, in descending order of time.
- SELECT total 3 rows, second to fourth, from the member table, in descending order of time. Note: it does not mean SELECT rows where id are 2, 3, or 4.
- SELECT rows where username equals to test.
- SELECT rows where name includes the es keyword.
- SELECT rows where both username and password equal to test.
- UPDATE data in name column to test2 where username equals to test.

Task 4: SQL Aggregation Functions

Write SQL statements to complete following requests based on the database and table created in Task 2.

- SELECT how many rows from the member table.
- SELECT the sum of follower_count of all the rows from the member table.
- SELECT the average of follower_count of all the rows from the member table.
- SELECT the average of follower_count of the first 2 rows, in descending order of follower_count, from the member table.

Task 5: SQL JOIN

Complete the following requests to support a message board system.

- Create a new table named message, in the website database. designed as below:

Column Name	Data Type	Additional Settings	Description
id	bigint	primary key, auto increment	Unique ID
member_id	bigint	not null, foreign key refer to id column in the member table	Member ID for Message Sender
content	varchar(255)	not null	Content
like_count	int unsigned	not null, default to 0	Like Count
time	datetime	not null, default to current time	Publish Time

- SELECT all messages, including sender names. We have to JOIN the member table to get that.
- SELECT all messages, including sender names, where sender username equals to test. We have to JOIN the member table to filter and get that.
- Use SELECT, SQL Aggregation Functions with JOIN statement, get the average like count of messages where sender username equals to test.
- Use SELECT, SQL Aggregation Functions with JOIN statement, get the average like count of messages GROUP BY sender username.

How to Prepare and Submit Tasks:

1. Use mysqldump command to export the website database to a file named data.sql, putting it in the week5 folder.
2. For task 2, 3, 4, and 5: create/edit README.md file in the week5 folder of your GitHub Repository. For every request, write down your answer, including SQL statement and screenshot of the executing result in MARKDOWN syntax.