

ECM1417: Web Development

Workshop 05

Matt Collison

Wednesday, February 17th, 2021

In the following workshop you will learn how to program stateful websites. To this end you will:

- Install and configure a MySQL DBMS.
- Execute SQL queries on your database.
- Integrate SQL queries into PHP using `mysqli`

1 Database

For this exercise you first need to install MySQL:

- First install the `mariadb-server` and `php-mysql` packages from the commandline on your Azure VM using the command below.

```
sudo apt-get install mariadb-server php-mysql
```

- When the MySQL DBMS (`mariadb-server`) package is installed, a configuration script is added to your path. Execute the installation script below to setup your MySQL installation: `mysql_secure_installation`.

When asked for the root password just hit enter, skip the setting of a new root password ('N'), and confirm all the remaining questions with 'Y'.

- Then, launch `mysql` using `sudo` and execute the following:
 - First, create a new database "webdev" using

```
CREATE DATABASE webdev;
```
 - Then, create a new user for "webdev" using

```
GRANT ALL ON webdev.* TO '<user>'@'localhost'
IDENTIFIED BY '<password>' WITH GRANT OPTION;
```
 - Finally execute `FLUSH PRIVILEGES;` to update the permissions and exit `mysql`.

Now you can login to `mysql` using the new user by executing `mysql -u <user> -p`. Then do the following:

1. Create two tables:

Users contains an automatically incremented `userid` and a `username`

```
CREATE TABLE Users(
  userid int NOT NULL AUTO INCREMENT PRIMARY KEY,
  username varchar(255),
  unhashedpassword varchar(255)
);
```

ToDo contains a name for a task and a deadline date

```
CREATE TABLE ToDo(
  task varchar(255),
  deadline date
);
```

2. Now, connect the two tables in a way that a user can have multiple todos but todos cannot be shared. `ALTER TABLE ToDo ADD userid int, ADD FOREIGN KEY ToDo(userid) REFERENCES Users(userid) ON DELETE CASCADE;`
3. Now populate the table with some data
 - Add three different users. One user example is provided below:
`INSERT INTO Users(username,unhashedpassword) VALUES ('Matt','UnsecurePassword');`
 - Add three todos. Again only one example is provided.
`INSERT INTO ToDo VALUES ("Complete ECM1417 Coursework","2022-04-01", (SELECT userid FROM Users WHERE username = "Matt"));`
 - Add two todos to one user and one to another. One user remains without todos.
4. Create a SQL query to list all users and their todos (Users with no todos should also be displayed).
`SELECT * FROM Users JOIN ToDo ON Users.userid=ToDo.userid;`
5. Create a second SQL query to list all todos which are due in two weeks from now.
`SELECT task FROM ToDo WHERE deadline > '2022-02-14';`

2 PHP and DB

For this exercise you should create a simple PHP application using the database from the last exercise:

- First, create an HTML form in `action.php` which allows a user to enter a username, password and task description and submit it to `action.php` as a post request using the code below.

```
<form action="/action.php" method="post">
<label for="uname">Username:</label><br>
<input type="text" id="username" name="uname" placeholder="Enter username"><br>
<label for="pass">Password:</label><br>
<input type="password" id="password" name="pass" Placeholder="Password"><br>
<label for="task">Task:</label><br>
<input type="text" id="task" name="task" placeholder="Enter task description"><br><br>
<input type="submit" value="Submit">
</form>
```

- Second, add the php code at the top of `action.php` to connect to the database and submit the fields set in the form.
- Next, edit `action.php` so when the form is submitted all the tasks in the `ToDo` table are displayed and the form is not displayed.
- Finally, create a session cookie and store the username only if the correct password is used in the `action.php` form. Make the page display all the tasks for the user when the session variable is set in and display the form if not.