Introduction to Database Systems

Homework 3: Login System

Introduction

In this homework, you need to create a simple login system using MySQL and Flask with Python.

Run the Project

1. **Install the Required Libraries** Run the following command to install the necessary libraries:

   ```
   pip install flask mysql-connector-python
   ```

2. **Create a New Database** Execute the following SQL command to create a new database:

   ```
   CREATE DATABASE hw3;
   ```

3. **Create a `users` Table** Create a table to store user information:

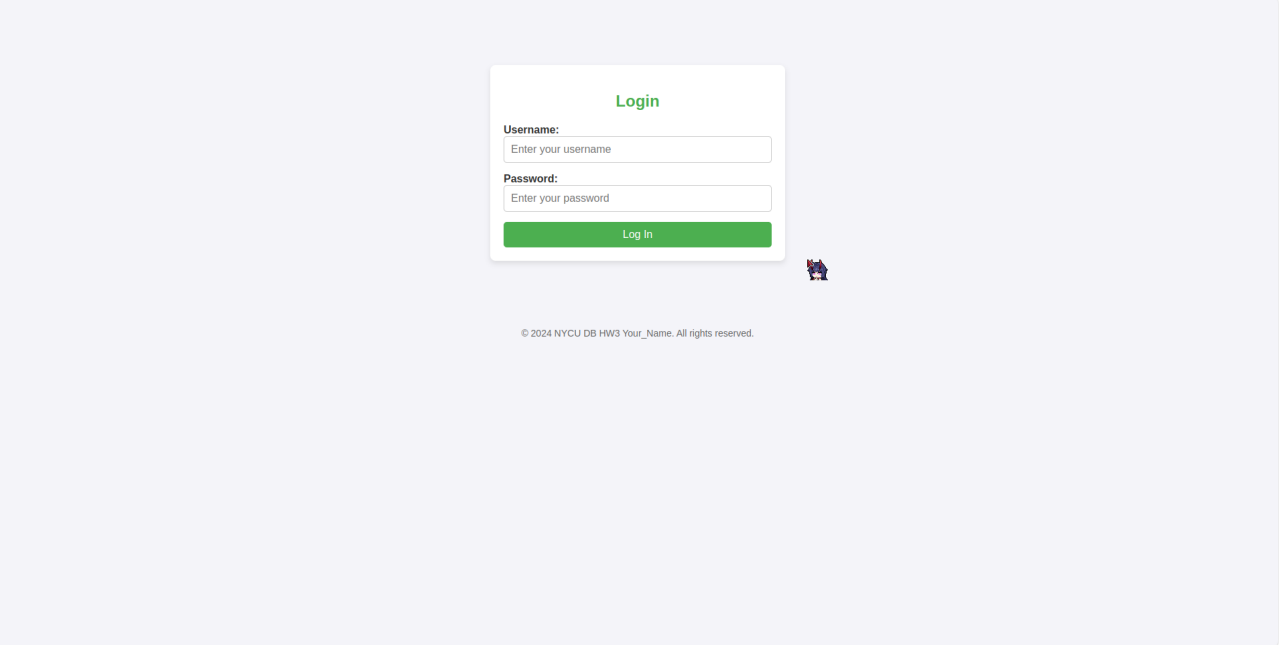
   ```
   CREATE TABLE users (
       id INT AUTO_INCREMENT PRIMARY KEY,
       username VARCHAR(255) NOT NULL UNIQUE,
       password VARCHAR(64) NOT NULL
   );
   ```

4. **Add a User** Insert a user into the table:

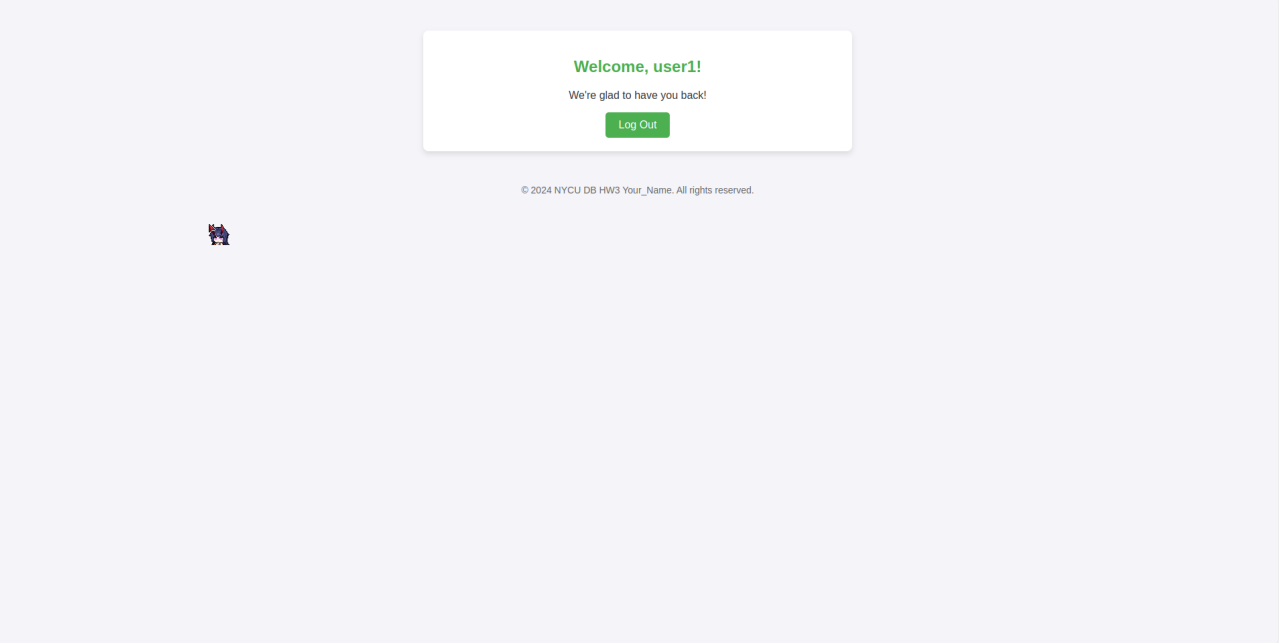
   ```
   INSERT INTO users (username, password)
   VALUES ('admin', '1234');
   ```

5. **Run the Flask Application** Launch the application by running `main.py`. Ensure you update the `db_config` at the top of `main.py` with your MySQL credentials.

   After starting the Flask server, you should see the login page:



Enter username and password to access the welcome page:



Coding Part (70%)

1. Update Footer on Every Page (10%)

Update the footer of every webpage to display the following text:

```
2024 NYCU DB HW3 {Your_ID} {Your_Name}. All rights reserved.
```

---

2. Complete the Login Function (20%)

Implement the login function to verify whether the username and password are correct.

---

3. Complete the Signup Function (20%)

Implement the signup function to allow users to register by adding new records to the database.

---

4. Hash the Password Using SHA-256 (10%)

To secure sensitive data, never store passwords in plaintext. Use the SHA-256 hashing function to hash passwords before storing them in the database.

- Example: The SHA-256 hash of the password `1234` is:

  ```
  03ac674216f3e15c761ee1a5e255f067953623c8b388b4459e13f978d7c846f4
  ```

  Use this hash value to debug your code.

---

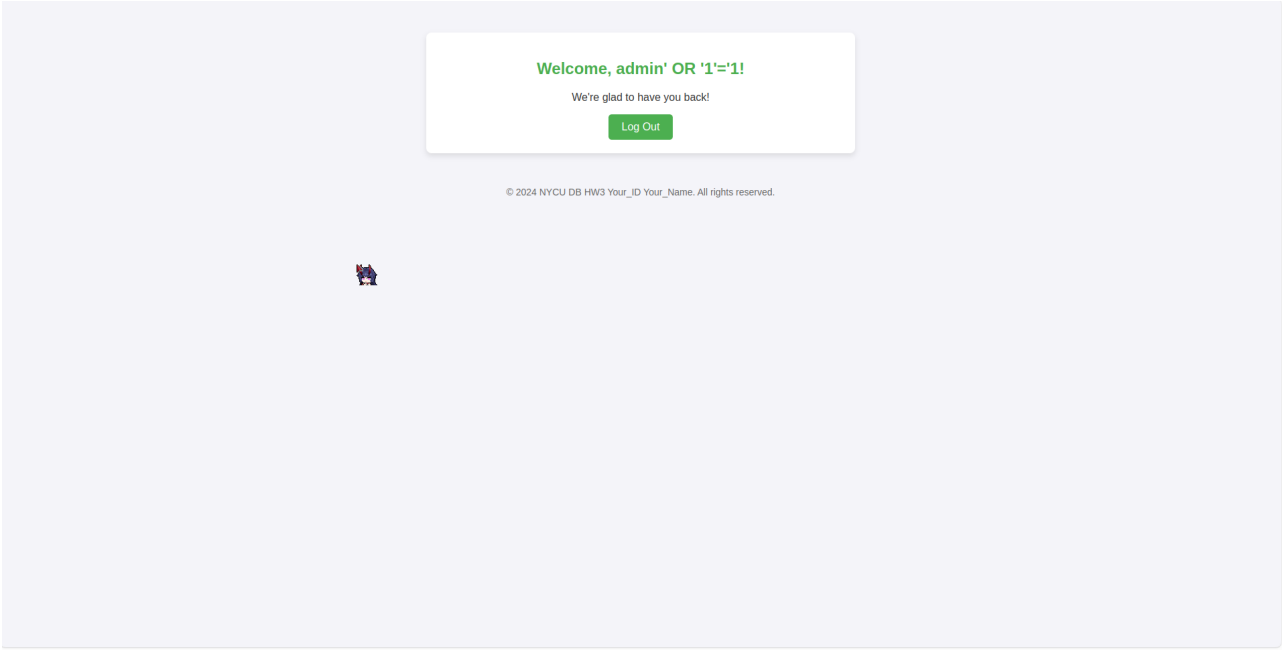
5. Prevent SQL Injection (10%)

What is SQL Injection?

SQL Injection is a web security vulnerability that allows attackers to manipulate database queries by injecting malicious SQL code into input fields or URLs. This can allow attackers to bypass authentication, access or modify data, and even compromise the entire database server.

- Example: Entering the following into the username field and using any password may bypass authentication:

  ```
  admin' OR '1'='1
  ```



To learn more, check out Wikipedia's SQL Injection page.

Report (30%)

1. Screenshot of Footers (5%)

Include screenshots of the updated footer on all three pages (login, signup, and welcome pages).

2. Screenshot and Explanation of Login Function (5%)

- Provide a screenshot of your login function implementation.
- Include screenshots of the login page and the welcome page after successful login.

3. Screenshot and Explanation of Signup Function (5%)

- Provide a screenshot of your signup function implementation.
- Include screenshots of the signup page and the login page for both successful and failed attempts.

4. Screenshot of Hashed Passwords (5%)

- Provide a screenshot of your password hashing code.
- Show the hashed password by executing:

```
SELECT * FROM users;
```

5. Screenshot and Explanation of SQL Injection Prevention (10%)

- Test your code with the input `admin' OR '1'='1` and provide screenshots of the results.
- Explain how your method prevents SQL injection and why it works.

Discussion

TAs had opened a channel HW3 討論區 on Teams of the course, you can post questions about the homework on the forum. TAs will answer questions as soon as possible.

Discussion rules:

1. Do not ask for the answer to the homework.
2. Check if someone has asked the question you have before asking.
3. We encourage you to answer other students' questions, but again, do not give the answer to the homework. Reply the messages to answer questions.
4. Since we have this discussion forum, do not send email to ask questions about the homework unless the questions are personal and you do not want to ask publicly.

Submission

1. The deadline of this homework is **12/9 (Mon.) 23:59:00.**

2. The submission requires all files (including login.html signup.html welcome.html and main.py) and a PDF report file name as **HW3_XXXXXX_report.pdf.**
3. You should put all the files into one folder, the folder should be named as **"HW3_XXXXXXX"** where XXXXXXX is your student ID. ex: **HW3_123456** Then compress your folder into one `zip` file. Submit it to New E3 System with the format **HW3_XXXXXXX.zip** where XXXXXXX is your student ID. ex: **HW3_123456.zip**

Your submission Format should be like:

```
HW3_123456.zip
└── HW3_123456
    ├── templates
    │   ├── login.html
    │   ├── signup.html
    │   └── welcome.html
    ├── HW3_123456_report.pdf
    └── main.py
```

We only accept one zip file, each wrong format or naming format causes -10 points to your score (after considering late submission penalty). 4. Late submission lead to score of (original score)*0.85days, for example, if you submit your homework right after the deadline, you get (original score)*0.85 points. 5. **0 points will be given to Plagiarism.** If the codes are similar to other people and you can't explain your code properly, you will be identified as plagiarism. TAs will strictly examine your code. It is important that you must write your code by yourself. 6. If there is anything you are not sure about submission, ask in the discussion forum.