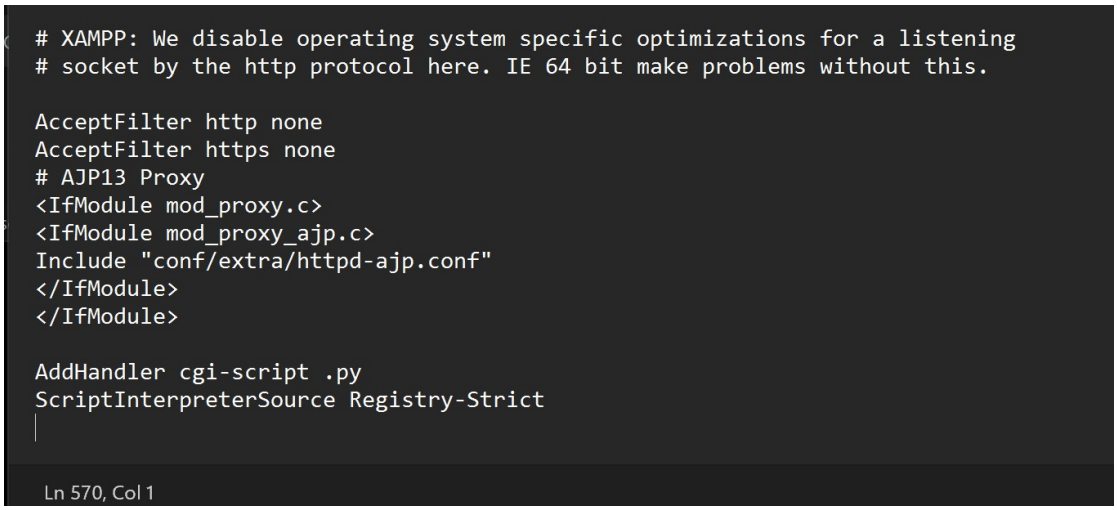
Part I. Database Interaction

1. Install Python 3.9.13.
    a. https://www.python.org/downloads/release/python-3913/
        i. Check on 'Add Python 3.9 to PATH'

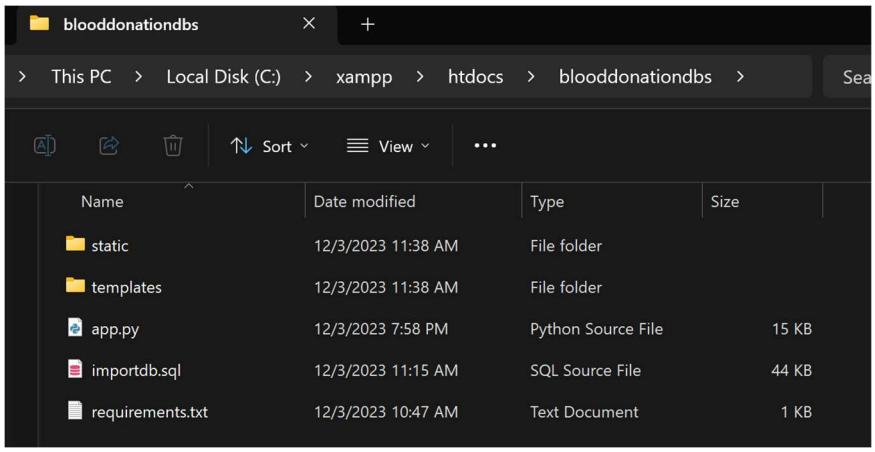
2. Configure XAMPP for Python.
    a. In XAMPP control panel, click 'Config' button of Apache
        i. Click 'Apache(httpd.conf)' → It will open text editor for httpd.conf.
        ii. At the end of the file, insert the following scripts and save the file

            AddHandler cgi-script .py

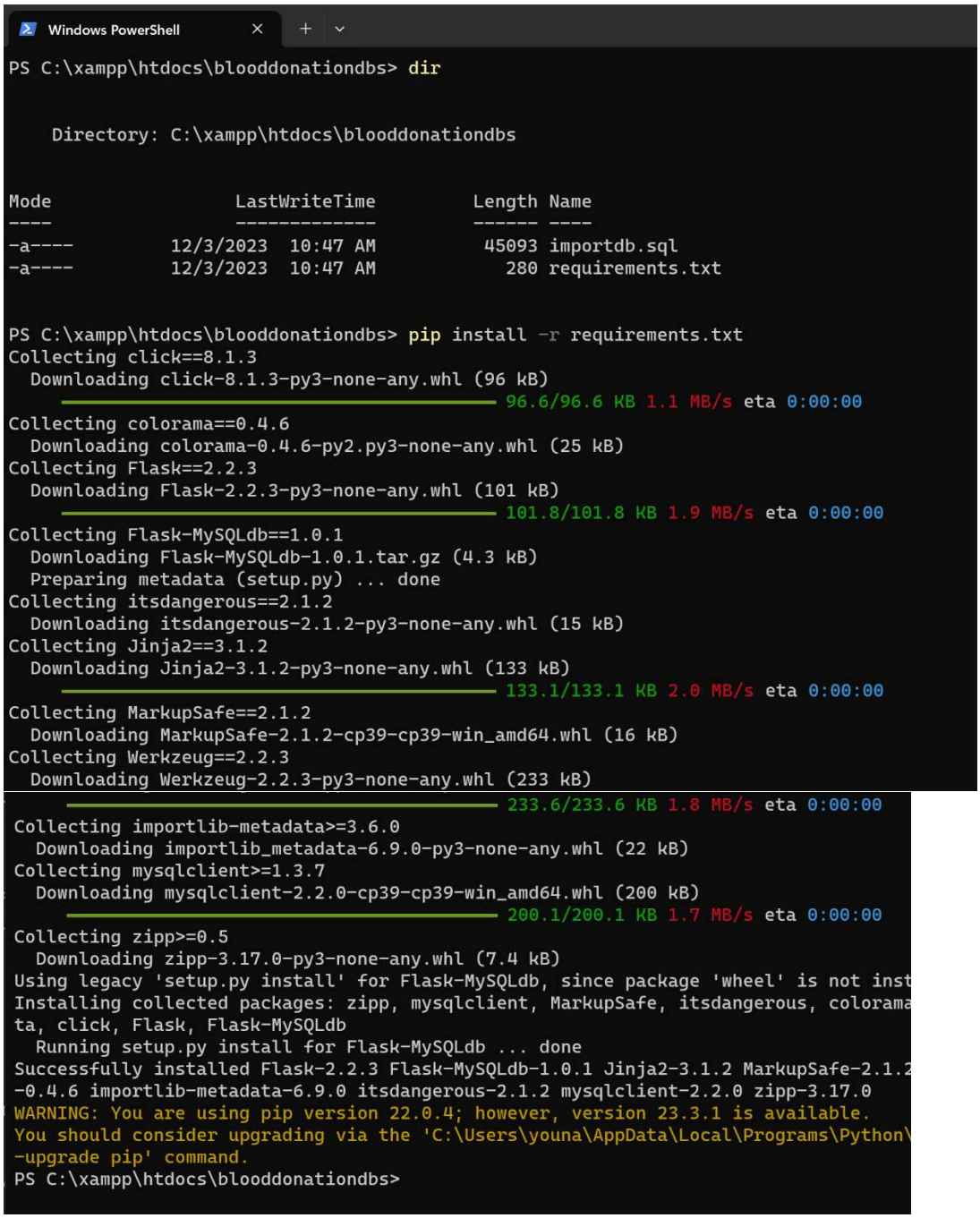
            ScriptInterpreterSource Registry-Strict



3. Create a folder (ex. blooddonationdbs) in XAMPP/htdocs folder and extract the provided zip file to the folder.



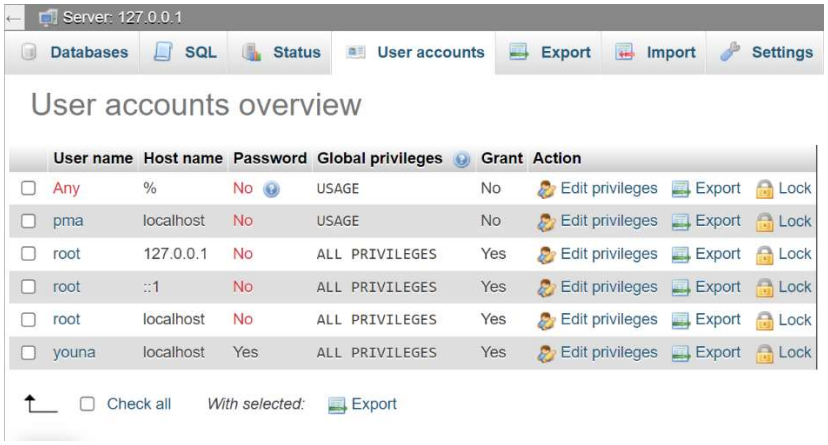
4. Open command for MAC or PowerShell for Windows
    a. Move to your project folder (i.e., blooddonationdbs)
        i. To list all the directories, use ‘dir’
        ii. To move to another directory, use ‘cd’
    b. Execute: pip install -r requirements.txt



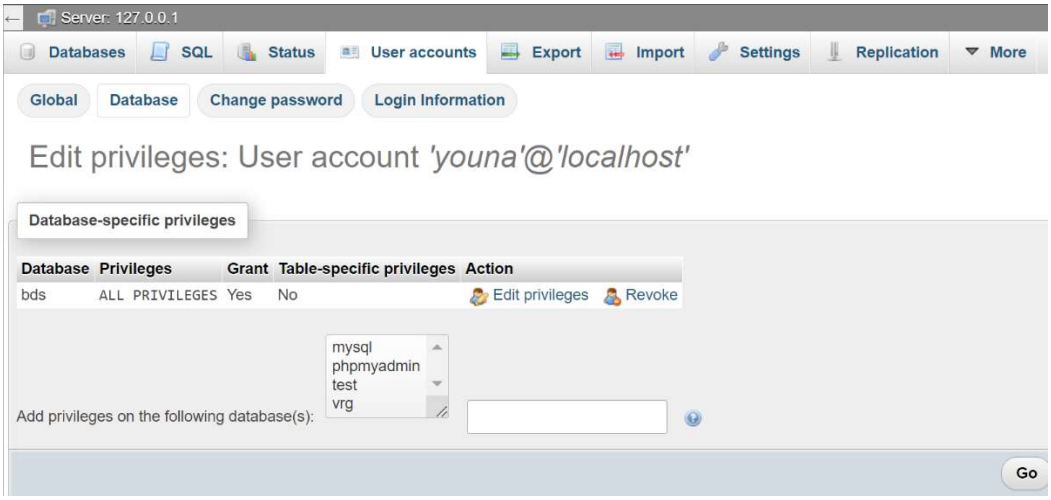
    c. Leave the PowerShell running!

5. Import Database
    a. Run SQL and Apache → Open Phpmysqladmin page
    b. Create a database 'bds'
    c. Select the created Database.
    d. Click the import tab.
        i. In the file to import section, choose file "importdb.sql" in your project folder that you created in step4.
        ii. In the 'Other options' section, **Disable** 'Enable foreign key checks' option.
        iii. Click 'Import'.
        iv. Then you can see tables in the database tree

6. Create Database user.
    a. Click 'Home' button in phpMyAdmin (top-left corner)
    b. Click the 'User'Accounts' tab.
    c. Click 'Add user account' button to create user account.
        i. Username – your firstname
        ii. Host name – Select 'Local'
        iii. Password – your own password
        iv. Check on 'Grant all Global privileges on wildcard name'
        v. In the 'Global privileges', check 'Check all' → hit 'Go'



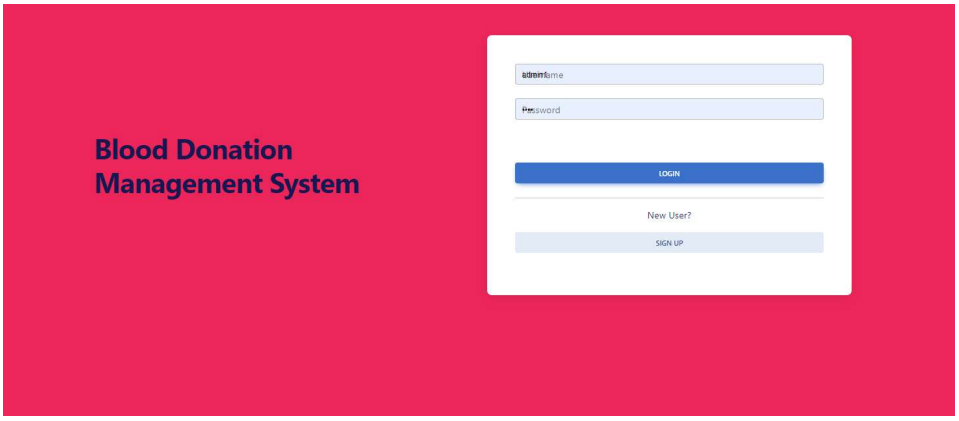
    d. Click the created user and assign Database (dbproject) to the user.
        i. Click the database button.
        ii. Select 'dbs' in the 'Add privileges on the following database(s)' → hit 'Go'→ 'Check all' → 'Go'



7. Go back to the command line window.
    a. Execute: python app.py -u <yourdbusername> -p <yourdbpassword>
    b. If successful, you can see the following message.

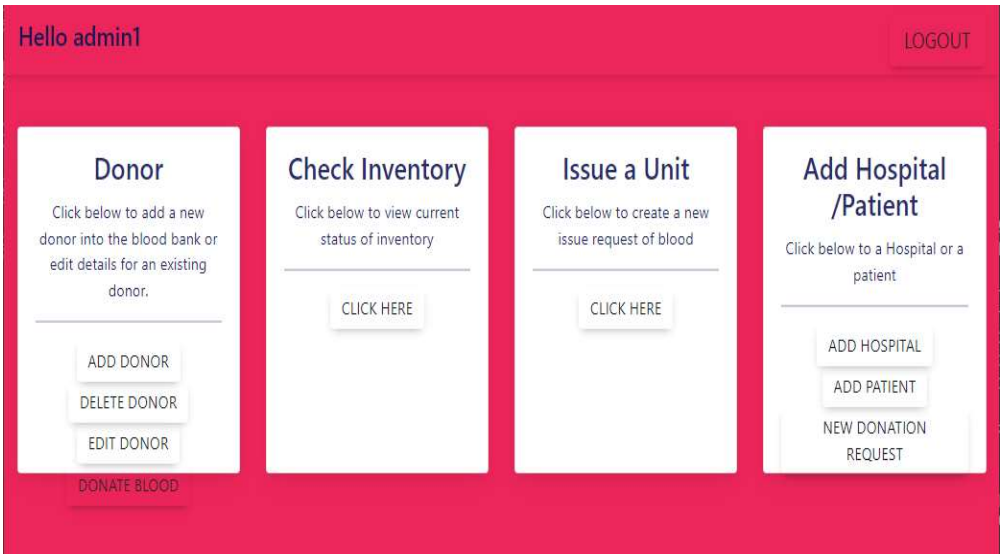
```
PS C:\xampp\htdocs\blooddonationdbs> python app.py -u youna -p 77dbskWkd!
 * Serving Flask app 'app'
 * Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment.
 Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 135-995-614
```

8. In web browser, type http://127.0.0.1:5000/ and check the following page is shown.



9. If successful, type “**admin1**” for username and “**root**” for password. Then Login.

10. If you can see the following page, then you are all set.



Part II. SQL Integration Problems

1. Let us assume you are at the main page (http://127.0.0.1:5000/profile/1/), following a successful login. Upon clicking the "CLICK HERE" button within the "Check Inventory" section, the goal is to display all the blood bags stored in the database. Examine the 'app.py' file to locate the code associated with the "Check Inventory" menu. Then, insert an accurate SQL query to execute when the "CLICK HERE" button in the "Check Inventory" section is pressed.

2. Upon clicking the "EDIT DONOR" button within the "Donor" section, the goal is to search for information about a particular donor using his/her phone number and then enable users to modify the donor information. Examine the 'app.py' file to locate the code associated with the 'Search Donor' button. Then, insert an accurate SQL query to execute when the "SEARCH DONOR" button in the "Edit Existing Donor Record" page is pressed.

3. The Blood Donation system incorporates multiple SQL procedures.

    A. Upon selecting "ADD DONOR" in the "Donor" Section on the main page and inputting essential details for a new donor on the 'Register a Donor' page, the system is required to create a new donor entry in the 'bds' database by invoking a procedure. Identify the procedure associated with the "REGISTER" button in the "Register a Donor" page and specify the procedure name within the corresponding code block in the 'app.py' file to facilitate the addition of a new donor to the database.

    B. Upon selecting "ADD HOSPITAL" in the "Add Hospital/Patient" Section on the main page and inputting essential details for a new hospital on the 'Add a Hospital' page, the system is required to create a new hospital entry in the 'bds' database by invoking a procedure. Identify the procedure associated with the "REGISTER" button in the "Add a Hospital" page and specify the procedure name within the corresponding code block in the 'app.py' file to facilitate the addition of a new donor to the database.