Full Stack Web Development
Mid Term Practical Examination
(6th Semester CSE A/B)
Max. Time - 2.25 Hrs (9 AM - 11:15 AM)
Prepared By - Deepak Uniyal, Asst. Prof. CSE

Instructions -

1. All the questions are compulsory.
2. You are allowed to use the Internet but not allowed to copy the code directly from the Internet.
3. No one should copy the code from others. If someone is found doing so or any plagiarism is found in the code, a strict penalty will be applied.
4. Create a directory having the format (classRollNo_Section_UniversityRollNo_FirstName_LastName) and keep it inside directory **CSE_A or CSE_B** corresponding to your section**.**
5. Keep all your codes inside the directory you created in the format stated above.
6. Anything you commit after the time allocated for the exam will not be evaluated.
7. No pull requests should be made after the designated time.
8. You have to perform a writeup and implement the code in the given time.

1. Please use the JSON file that contains some records of photos - You need to read the JSON data and do the following things - Create an object of albumId, Id, last portion of URL. **For example** - for 1st two you should get the following
   photosData = {**1**:[**1**,"**92c952**"], **2**:[**1**,"**771796**"]}

   See the following portion of color-coded JSON data from the file to understand the mapping of data from JSON to Javascript Object photosData.

```
{
   "albumId": 1,
   "id": 1,
   "title": "accusamus beatae ad facilis cum similique qui sunt",
   "url": "https://via.placeholder.com/600/92c952",
   "thumbnailUrl": "https://via.placeholder.com/150/92c952"
 },
 {
   "albumId": 1,
   "id": 2,
   "title": "reprehenderit est deserunt velit ipsam",
   "url": "https://via.placeholder.com/600/771796",
   "thumbnailUrl": "https://via.placeholder.com/150/771796"
 },
```

   If you try to access the record of the photo with id 1, you will get the corresponding album id and the last portion of the URL or thumbnail URL in a list. **[***Make one commit here stating the following message -* **process JSON data to create an object of photos]**

2. Develop a program that does the following -
   Create an HTML form that takes the following values as input - ***username*** (should not be empty), ***email*** (should be a valid email address), ***contact*** of 10 digits (It should not allow anything other than digits), ***gender*** (use radio box to select the gender from following values - male, female, other), ***city*** (use a drop-down to select the city from the list of following 10 cities [Mumbai, Delhi, Jaipur, Mussoorie, Nanital, Chennai, Pune, Bangalore, Hyderabad, Lucknow]).

   a) Try submitting data from the form using the **get** method and take its screenshot to show the way data is being sent from the browser. You need to save this screenshot with the name "**get_data.XXXX**" where XXXX is the format of the image saved. **[***Make one commit here stating the following message* - **create an HTML form with the get method]**
   b) Try submitting data while some values are not in the required format (**1.** contact having a length less than 10 **2.** Email not in proper format **3.** Empty username) and take multiple screenshots showing that form is indicating an error. You could save screenshots in the following format - **contact_invalid_js.XXXX**, **email_invalid_js.XXXX**, **username_invalid_js.XXXX** etc.
   c) Use the post method to submit the data which will be further processed by the PHP. **[***Make another commit here stating the following message* - **modify the HTML form to use post method]**
   d) Create a PHP file named **configuration.php** that is able to connect to the database. **[***Make another commit here stating the following message* - **create a database connection using PHP]**
   e) The database should be named **WebPractial** and the table should be named **users** with the following columns - ***id, username, email, contact, gender, and city.*** The column ***id*** should be the **primary key** with the **auto-increment** feature. Take a screenshot of the database server having a database and table showing all columns and save it with the name **empty_table.XXXX**.
   f) *For questions **(f-i)** you should write your code in a file named **index.php**. You can use suitable functions of your choice to make the code more readable and following the modular approach.* Write a PHP code to retrieve the form data but ensure that form is not submitted while refreshing the page. Take the following variables to assign the data - $username, $email, $contact, $gender, and $city **[***Make another commit here stating the following message* - **retrieve form data while avoiding form submission on refreshing the page]**
   g) Once you are done retrieving the form data, pass it to another function for the following validation. **1.** username, email, contact, gender, and the city should not be empty **2.** contact should have a length of 10. **[***Make another commit here stating the following message* - **validating data using PHP]**
   h) If everything is fine till now, write a PHP script to insert the data into the **users** table. **[***Make another commit here stating the following message* - **insert data into database using PHP script]**
   i) Now you have inserted the data successfully, try retrieving the data from the database using a PHP script and show it on a web page in the following

format - **[***Make another commit here stating the following message -* **retrieve data from users table using PHP script]**

Sample Output - Following is the output for the details of two users. The left side shows the headings while the right side shows the original values for those columns.

Users Data -

1. **Details of user with ID = id**
   - UserName - username
   - Email - email
   - Contact - contact
   - Gender - gender
   - City - city

2. **Details of user with ID = id**
   - UserName - username
   - Email - email
   - Contact - contact
   - Gender - gender
   - City - city

j) Take a screenshot of the table along with its data and user details on the web page. You can open two tabs side by side and take a single screenshot showing the data in the database and web page. Save the screenshot with the name **final_data.XXXX.** Now you can save all the screenshots into a directory named **output**. **[***Make another commit here stating the following message -* **screenshots of output results]**

k) In the end, you need to make a pull request on the given GitHub repository.

All The Best