**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

Інститут **КНІТ**

Кафедра **ПЗ**

ЗВІТ

До лабораторної роботи № 6

**З дисципліни:** *“* *Основні типи та функції доступу до БД MySQL”*

**На тему:** *“* *Використання React JS та Bootstrap”*

**Лектор:**

проф. каф. ПЗ

Мельник Р. А.

**Виконав:**

ст. гр. ПЗ-23 Михалевич П.-І. В.

**Прийняв:**

доц. каф. ПЗ

Грицай О.Д.

« » 2022 р.

∑= .

Львів – 2022

**Тема роботи:** Основні типи та функції доступу до БД MySQL.

**Мета роботи:** Засвоїти елементи створення, модифікації, читання та занесення даних з таблиць БД засобами РНР.

**ТЕОРЕТИЧНІ ВІДОМОСТІ**

Для використання MySQL-сервера баз даних необхідно компілювати php програми з відповідною підтримкою (опція --with-mysql). Якщо шлях до mysql не вказується, php використовуватиме вбудовані в mysql бібліотеки клієнта. Специфікація шляху до mysql: -- with-mysql=/path/to/mysql. Тоді php використовує відповідні бібліотеки.

Для використання MySQL-сервера баз даних необхідно компілювати php програми з відповідною підтримкою (опція --with-mysql). Якщо шлях до mysql не вказується, php використовуватиме вбудовані в mysql бібліотеки клієнта. Специфікація шляху до mysql: --with-mysql=/path/to/mysql. Тоді php використовує відповідні бібліотеки.

Сучасна версія PHP 5 підтримує доступ до бази даних MySQL командами, які в своїй назві використовую термін MySQLi для позначення об’єктно-орієнтованого та процедурного варіантів використання. Таке позначення використовується якщо реалізується доступ тільки до MySQL і не передбачається переналаштування на іншу базу даних. Якщо ж в програмі можливі зміни джерела даних, раціонально використати об’єктно-орієнтований інтерфейс доступу PDO (PHP Data Objects), який передбачає підключення до 12 різних баз даних.

**ЗАВДАННЯ**

Додати до попередньо розгорнутого проекту Laravel:

● Перевірити налаштування до бд;

● створити міграцію для додання у базу таблички ‘gamer’ та його полів (хоча б ім’я

та вік) які відображені на формі даних гравця із сторінки з полем для шахів

● сворити модель гравця та написати збереження даних гравця у контролері

написаному в попередній лабораторній

● додати на сторінку із полем для шахів функцію збереження даних гравця, яка

надсилатиме дані та виводитеме отриману відповідь як alert()

**ХІД РОБОТИ**

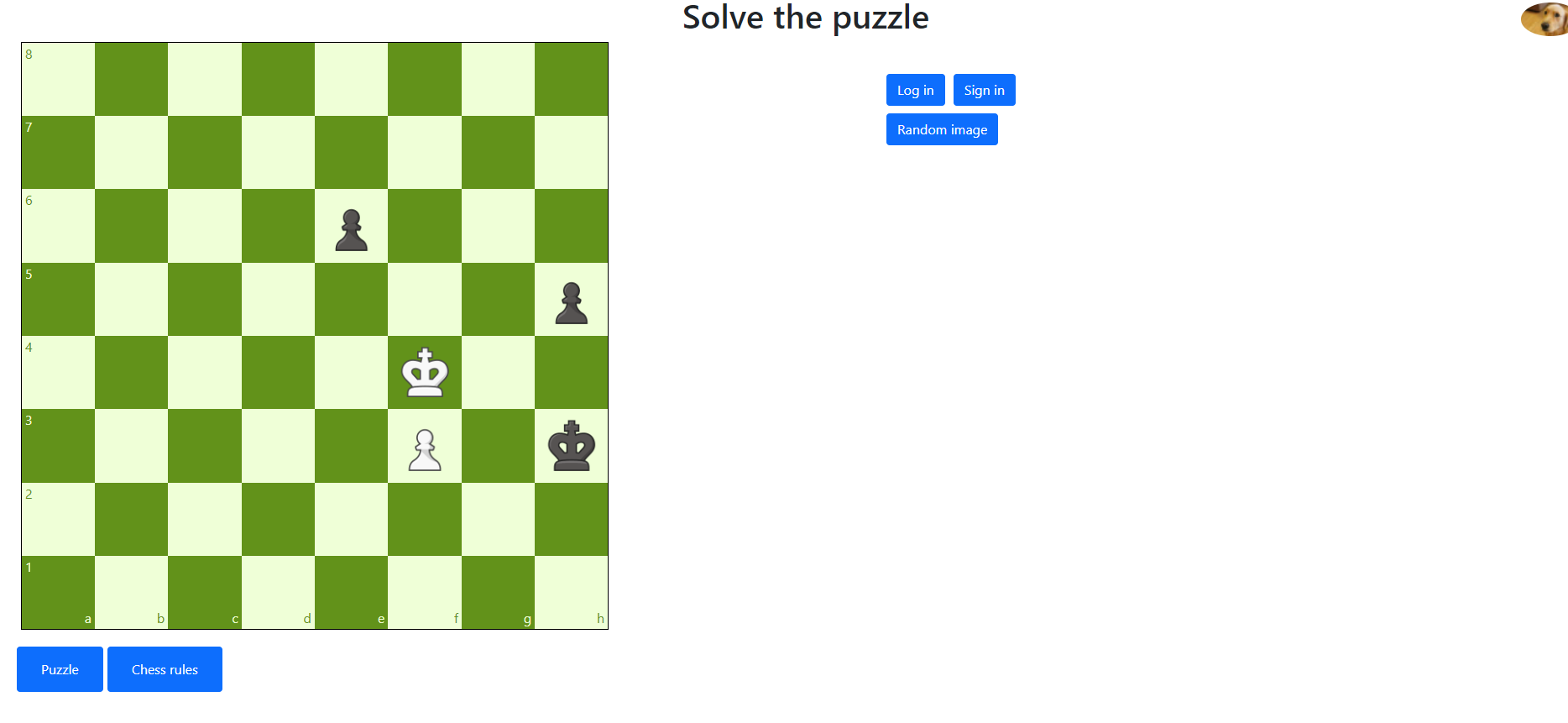
****

Рис.1. Початкова сторінка

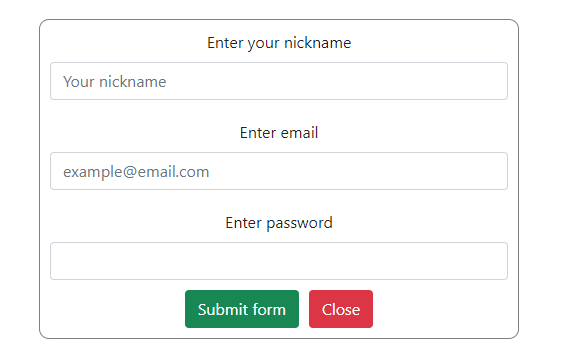


Рис.2. Форма входу для гравця

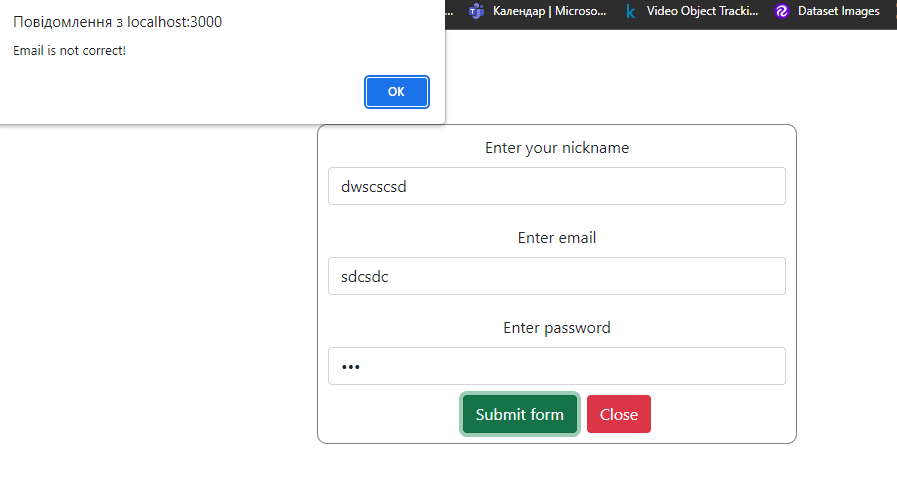


Рис.3. Валідація форми

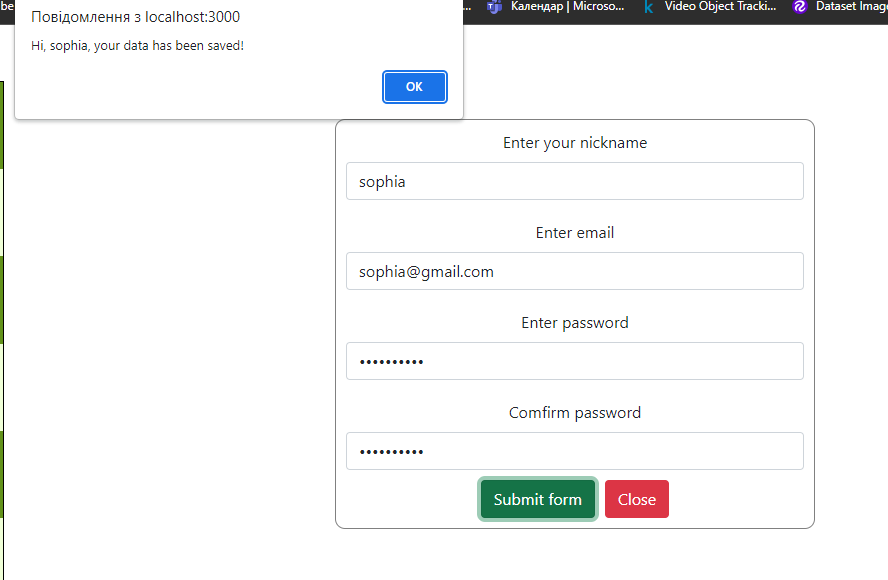


Рис. 4. Успішне збереження даних у БД

Код програми:

Login.php:

<?php

header('Access-Control-Allow-Origin: \*');

$servername = "localhost";

$username = "root";

$password = "root";

$dbname = "chess\_players";

// Create connection

$conn = mysqli\_connect($servername, $username, $password, $dbname);

$response = array('sever\_error' => false,

                  'server\_error\_text' => "",

                  'validated'=>true,

                  'validation\_error\_text' => '',

                  'nickname' => '');

// Check connection

if (!$conn) {

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "Connection failed: " . mysqli\_connect\_error();

}

$nickname = $\_POST['nickname'];

$email = $\_POST['email'];

$pwd = $\_POST['password'];

//Validation

if(empty($nickname) or empty($email) or empty($pwd)){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Not all data was provided!";

}

if(!filter\_var($email, FILTER\_VALIDATE\_EMAIL)  and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Email is not correct! ";

}

if(!preg\_match("/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])[0-9a-zA-Z]{8,}$/",$pwd)  and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Password must have at least 1 upper and lower case letter, 1 number and consist of 8 characters! ";

}

if(strlen($nickname) < 3 and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Nickname must be at least 3 characters! ";

}

$db\_answer = mysqli\_query($conn, "SELECT \* FROM chess WHERE (username = '$nickname')");

if (db\_answer) {

  $user = mysqli\_fetch\_assoc($db\_answer);

  if($user != null){

    if($pwd != $user['password']){

      if($response['validated'] != false){

        $response['validated'] = false;

        $response['validation\_error\_text'] = "Invalid password! ";

      }

    }else{

      $response['nickname'] = $nickname;

    }

  }else if($response['validated'] != false){

    $response['validated'] = false;

    $response['validation\_error\_text'] = "No such nickname registered! ";

  }

} else if($response['sever\_error'] != true){

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "DB error! ";

}

$conn->close();

echo json\_encode($response);

?>

Registration.php:

<?php

header('Access-Control-Allow-Origin: \*');

$servername = "localhost";

$username = "root";

$password = "root";

$dbname = "chess\_players";

// Create connection

$conn = mysqli\_connect($servername, $username, $password, $dbname);

$response = array('sever\_error' => false,

                  'server\_error\_text' => "",

                  'validated'=> true,

                  'validation\_error\_text' => '',

                  'nickname' => '');

// Check connection

if (!$conn) {

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "Connection failed: " . mysqli\_connect\_error();

}

$nickname = $\_POST['nickname'];

$email = $\_POST['email'];

$pwd = $\_POST['password'];

//Validation

if(empty($nickname) or empty($email) or empty($pwd)){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Not all data was provided!";

}

if(!filter\_var($email, FILTER\_VALIDATE\_EMAIL) and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Email is not correct! ";

}

if(!preg\_match("/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])[0-9a-zA-Z]{8,}$/",$pwd) and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Password must have at least 1 upper and lower case letter, 1 number and consist of 8 characters! ";

}

if(strlen($nickname) < 3 and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Nickname must be at least 3 characters! ";

}

$db\_answer = mysqli\_query($conn, "SELECT \* FROM chess WHERE (username = '$nickname')");

if ($db\_answer) {

  $user = mysqli\_fetch\_assoc($db\_answer);

  if($user != null){

    if($response['validated'] != false){

      $response['validated'] = false;

      $response['validation\_error\_text'] = "Such nickname already exist! ";

    }

  }else{

    $result = mysqli\_query($conn, "INSERT INTO chess (username, email, password) VALUES ('$nickname', '$email', '$pwd')");

    if ($result) {

      $response['nickname'] = $nickname;

    } else {

      if($response['sever\_error'] != true){

        $response['sever\_error'] = true;

        $response['server\_error\_text'] = "DB error! Can't insert! ";

      }

    }

  }

} else {

  if($response['sever\_error'] != true){

    $response['sever\_error'] = true;

    $response['server\_error\_text'] = "DB error! Can't get data! ";

  }

}

$conn->close();

echo json\_encode($response);

?>

**Висновок**

На даній лабораторній роботі було отримано базові знання про будову баз даних, засвоєно елементи створення, модифікації, читання та занесення даних з таблиць БД засобами РНР, навчився використовувати SQL запити. За допомогою сервісу OpenServer було створено базу даних на піднятому локальному сервері.