**Лабораторна робота №5**

**Тема: Робота з базою даних MySQL та з бібліотекою PDO у мові РНР**

**Мета роботи:** ознайомитися з бібліотекою PDO для роботи з базами даних у мові PHP

Хід роботи:

Завдання 1:

1. Створіть MySQL базу даних lab5 і в ній таблицю users, яка зберігатиме інформацію про користувачів.

2. На головній сторінці розмістіть форму для введення логіну та пароля і посилання на сторінку реєстрації.

3. Реалізуйте форму реєстрації користувачів та внесення даних до таблиці users. При цьому потрібно перевіряти коректність заповнення полів та заборонити повторну реєстрацію користувачів.

4. При заповненні форми входу на сайт, потрібно перевірити, чи є відповідний запис у таблиці users. Якщо є, то записати у сесію ознаку того, що користувач успішно аутентифікований і надалі для нього не відображати форму входу на сайт та форму реєстрації.

5. Для користувача, який успішно пройшов аутентифікацію потрібно на головній сторінці відобразити вітання та надати можливість зміни своїх даних, які він ввів при реєстрації. При зміні даних вони повинні змінюватися і в таблиці users.

6. Реалізуйте можливість виходу користувача.

7. Реалізуйте можливість видалення власного профілю користувачем.

Лістинг:

<?php

session\_start();

$dsn = "mysql:host=localhost;dbname=lab5;charset=utf8";

$dbUser = "root";

$dbPass = "";

try {

    $pdo = new PDO($dsn, $dbUser, $dbPass);

    $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $ex) {

    exit("Помилка з'єднання з БД: " . $ex->getMessage());

}

$message = "";

function getPostValue($key) {

    return isset($\_POST[$key]) ? trim($\_POST[$key]) : '';

}

// Реєстрація

if (isset($\_POST['register'])) {

    $username = getPostValue('login');

    $password = getPostValue('password');

    if ($username && $password) {

        $query = $pdo->prepare("SELECT id FROM users WHERE login = ?");

        $query->execute([$username]);

        if ($query->fetch()) {

            $message = "Такий користувач вже існує.";

        } else {

            $passwordHash = password\_hash($password, PASSWORD\_DEFAULT);

            $pdo->prepare("INSERT INTO users (login, password) VALUES (?, ?)")->execute([$username, $passwordHash]);

            $message = "Успішна реєстрація.";

        }

    } else {

        $message = "Будь ласка, заповніть усі поля.";

    }

}

// Авторизація

if (isset($\_POST['login\_btn'])) {

    $username = getPostValue('login');

    $password = getPostValue('password');

    $stmt = $pdo->prepare("SELECT \* FROM users WHERE login = ?");

    $stmt->execute([$username]);

    $user = $stmt->fetch(PDO::FETCH\_ASSOC);

    if ($user && password\_verify($password, $user['password'])) {

        $\_SESSION['user'] = $user;

    } else {

        $message = "Невірні дані для входу.";

    }

}

// Вихід

if (isset($\_GET['logout'])) {

    session\_destroy();

    header("Location: index.php");

    exit;

}

// Видалення акаунта

if (isset($\_POST['delete'])) {

    $userId = $\_SESSION['user']['id'];

    $pdo->prepare("DELETE FROM users WHERE id = ?")->execute([$userId]);

    session\_destroy();

    header("Location: index.php");

    exit;

}

// Оновлення даних

if (isset($\_POST['update'])) {

    $userId = $\_SESSION['user']['id'];

    $newLogin = getPostValue('login');

    $newPassword = getPostValue('password');

    $newHash = password\_hash($newPassword, PASSWORD\_DEFAULT);

    $pdo->prepare("UPDATE users SET login = ?, password = ? WHERE id = ?")

        ->execute([$newLogin, $newHash, $userId]);

    $\_SESSION['user']['login'] = $newLogin;

    $message = "Дані успішно оновлено.";

}

?>

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <title>Користувацька система</title>

</head>

<body>

<h2>Лабораторна робота №5</h2>

<?php if (isset($\_SESSION['user'])): ?>

    <p>Привіт, <strong><?= htmlspecialchars($\_SESSION['user']['login']) ?></strong>!</p>

    <form method="post">

        <label>Новий логін: <input type="text" name="login" required></label><br>

        <label>Новий пароль: <input type="password" name="password" required></label><br>

        <button type="submit" name="update">Оновити</button>

    </form>

    <form method="post">

        <button type="submit" name="delete">Видалити акаунт</button>

    </form>

    <p><a href="?logout">Вийти</a></p>

<?php else: ?>

    <h3>Вхід</h3>

    <form method="post">

        <label>Логін: <input type="text" name="login" required></label><br>

        <label>Пароль: <input type="password" name="password" required></label><br>

        <button type="submit" name="login\_btn">Увійти</button>

    </form>

    <h3>Реєстрація</h3>

    <form method="post">

        <label>Логін: <input type="text" name="login" required></label><br>

        <label>Пароль: <input type="password" name="password" required></label><br>

        <button type="submit" name="register">Зареєструватися</button>

    </form>

<?php endif; ?>

<?php if (!empty($message)) echo "<p style='color: red;'>$message</p>"; ?>

</body>

</html>

<?php

session\_start();

if (isset($\_SESSION["user"])) {

    header("Location: userProfile.php");

    exit;

}

$message = "";

if ($\_SERVER['REQUEST\_METHOD'] === "POST") {

    $login = trim($\_POST['login'] ?? '');

    $password = trim($\_POST['password'] ?? '');

    if ($login && $password) {

        try {

            $pdo = new PDO("mysql:host=localhost;dbname=lab5;charset=utf8", "root", "");

            $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

            $stmt = $pdo->prepare("SELECT \* FROM users WHERE login = ?");

            $stmt->execute([$login]);

            $user = $stmt->fetch(PDO::FETCH\_ASSOC);

            if ($user && password\_verify($password, $user['password'])) {

                $\_SESSION['user'] = $user;

                header("Location: userProfile.php");

                exit;

            } else {

                $message = "Невірний логін або пароль!";

            }

        } catch (PDOException $e) {

            $message = "Помилка БД: " . $e->getMessage();

        }

    } else {

        $message = "Будь ласка, заповніть всі поля.";

    }

}

?>

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <title>Вхід</title>

</head>

<body>

<h2>Вхід</h2>

<form method="POST">

    <label>Логін: <input type="text" name="login" required></label><br>

    <label>Пароль: <input type="password" name="password" required></label><br>

    <button type="submit">Увійти</button>

</form>

<p>Немає акаунту? <a href="register.php">Зареєструватись</a></p>

<?php if (!empty($message)) echo "<p style='color:red;'>$message</p>"; ?>

</body>

</html>

<?php

session\_start();

if (isset($\_SESSION["logged"]) && $\_SESSION["login"]) {

    header("Location: userProfile.php");

    exit;

}

$message = "";

if ($\_SERVER['REQUEST\_METHOD'] == 'POST') {

    $name = trim($\_POST['name'] ?? '');

    $login = trim($\_POST['login'] ?? '');

    $password = trim($\_POST['password'] ?? '');

    if ($name && $login && $password) {

        try {

            $pdo = new PDO("mysql:host=localhost;dbname=lab5;charset=utf8", "root", "");

            $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

            $stmt = $pdo->prepare("SELECT id FROM users WHERE login = ?");

            $stmt->execute([$login]);

            if ($stmt->rowCount() === 0) {

                $hashedPassword = password\_hash($password, PASSWORD\_DEFAULT);

                $insert = $pdo->prepare("INSERT INTO users (name, login, password) VALUES (?, ?, ?)");

                $insert->execute([$name, $login, $hashedPassword]);

                $\_SESSION["logged"] = 1;

                $\_SESSION["login"] = $login;

                header("Location: login.php");

                exit;

            } else {

                $message = "Такий логін вже існує.";

            }

        } catch (PDOException $e) {

            $message = "Помилка БД: " . $e->getMessage();

        }

    } else {

        $message = "Ви не ввели всі значення.";

    }

}

?>

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <title>Реєстрація</title>

</head>

<body>

    <h2>Реєстрація</h2>

    <form method="post">

        <table>

            <tr>

                <td>Ім’я:</td>

                <td><input type="text" name="name" value="<?= htmlspecialchars($name ?? '') ?>"></td>

            </tr>

            <tr>

                <td>Логін:</td>

                <td><input type="text" name="login" value="<?= htmlspecialchars($login ?? '') ?>"></td>

            </tr>

            <tr>

                <td>Пароль:</td>

                <td><input type="password" name="password"></td>

            </tr>

            <tr>

                <td></td>

                <td><button type="submit">Зареєструватися</button></td>

            </tr>

        </table>

    </form>

    <div>

        Вже є акаунт? <a href="login.php">Увійти</a>

    </div>

    <?php if (!empty($message)) echo "<p style='color:red;'>$message</p>"; ?>

</body>

</html>

<?php

session\_start();

function checkLogged(): ?array

{

    if (!isset($\_SESSION["logged"]) || !isset($\_SESSION["login"])) {

        header("Location: login.php");

        exit;

    }

    try {

        $pdo = new PDO("mysql:host=localhost;dbname=lab5;charset=utf8", "root", "");

        $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        $stmt = $pdo->prepare("SELECT \* FROM users WHERE login = ? LIMIT 1");

        $stmt->execute([$\_SESSION["login"]]);

        $user = $stmt->fetch(PDO::FETCH\_ASSOC);

        return $user ?: [];

    } catch (PDOException $e) {

        exit("Помилка БД: " . $e->getMessage());

    }

}

$accountData = checkLogged();

extract($accountData); // $id, $name, $login, $password

if ($\_SERVER['REQUEST\_METHOD'] === "POST") {

    try {

        $pdo = new PDO("mysql:host=localhost;dbname=lab5;charset=utf8", "root", "");

        $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        if (isset($\_POST['edit'])) {

            $newName = trim($\_POST['name'] ?? '');

            $newLogin = trim($\_POST['login'] ?? '');

            $newPassword = trim($\_POST['password'] ?? '');

            if ($newName && $newLogin) {

                $set = "name = :name, login = :login";

                $params = [

                    ':name' => $newName,

                    ':login' => $newLogin,

                    ':id' => $id,

                ];

                if (!empty($newPassword)) {

                    $set .= ", password = :password";

                    $params[':password'] = password\_hash($newPassword, PASSWORD\_DEFAULT);

                }

                $sql = "UPDATE users SET $set WHERE id = :id";

                $stmt = $pdo->prepare($sql);

                $stmt->execute($params);

                $\_SESSION['login'] = $newLogin;

                header("Location: userProfile.php");

                exit;

            }

        }

        if (isset($\_POST['exit'])) {

            session\_destroy();

            header('Location: login.php');

            exit;

        }

        if (isset($\_POST['delete'])) {

            $stmt = $pdo->prepare("DELETE FROM users WHERE id = ?");

            $stmt->execute([$id]);

            session\_destroy();

            header('Location: login.php');

            exit;

        }

    } catch (PDOException $e) {

        exit("Помилка: " . $e->getMessage());

    }

}

?>

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <title>Профіль користувача</title>

</head>

<body>

<form method="POST">

    <table>

        <tr>

            <td>Привіт, <?= htmlspecialchars($name) ?>!</td>

            <td><input type="submit" value="Вийти" name="exit"></td>

        </tr>

        <tr>

            <td>Ім’я</td>

            <td>Логін</td>

            <td>Новий пароль</td>

        </tr>

        <tr>

            <td><input type="text" name="name" value="<?= htmlspecialchars($name) ?>"></td>

            <td><input type="text" name="login" value="<?= htmlspecialchars($login) ?>"></td>

            <td><input type="password" name="password" placeholder="Не змінювати"></td>

            <td><input type="submit" value="Оновити" name="edit"></td>

        </tr>

        <tr>

            <td><input style="color: red;" type="submit" value="Видалити акаунт" name="delete"></td>

        </tr>

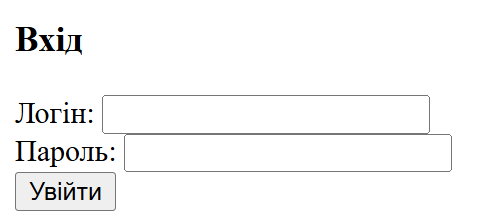
    </table>

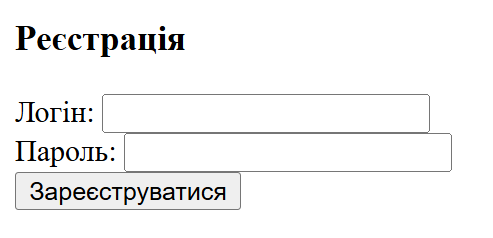
</form>

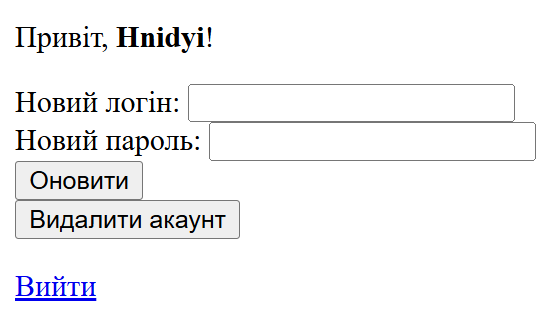
</body>

</html>

Результат виконання:







Завдання 2:

1. У розділі Користувачі додайте нового користувача homeuser з усіма правами.
2. Перевірте конфігураційний файл php.ini. Розширення php\_pdo\_mysql.dll має бути підключене.
3. Створіть папку Lab5. У ній файл index.php.
4. Підключіться до бази даних:
5. Для обробки виключень заключіть код в блок try…catch(). Опишіть в даному блоці дії, які необхідно виконувати при виникненні помилок. Наприклад:
6. До бази даних, створеної в попередньому завданні додайте ще одну таблицю з назвою tov: У поле Ім'я ввести tov; У поле Поля – число 5; Кнопка Ок.
7. Додайте 10 записів до таблиці tov. Тематику товарів придумайте самостійно.
8. Виведіть на екран всі дані з таблиці tov.
9. Оформіть виведення полів з таблиці tov у вигляді таблиці або блоків. Наприклад:
10. Розмістіть 2 кнопки Додати запис і Вилучити запис і текстове поле для вказівки № запису, що віддаляється.
11. Створіть файл insert.php, у якому розмістити форму для введення нових даних у таблицю.
12. Створіть файл delete.php, у якому опишіть код для видалення запису за заданим номером. Приклад для поновлення запису (аналогічний видаленню):
13. Експортуйте створену базу даних до файлу та скопіюйте до папки Lab6.

Лістинг:

<?php

try {

    $conn = new PDO("mysql:host=localhost;dbname=lab5", "root", "");

    $conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

    echo "Помилка підключення: " . $e->getMessage();

    exit;

}

if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {

    try {

        $stmt = $conn->prepare("INSERT INTO tov (name, price, quantity, note) VALUES (?, ?, ?, ?)");

        $stmt->execute([

            $\_POST['name'],

            $\_POST['price'],

            $\_POST['quantity'],

            $\_POST['note']

        ]);

        echo "<p style='color:green;'>Товар додано!</p>";

    } catch (PDOException $e) {

        echo "<p style='color:red;'>Помилка додавання товару: " . $e->getMessage() . "</p>";

    }

}

?>

<form method="post">

  <input name="name" placeholder="Назва" required><br><br>

  <input name="price" type="number" step="0.01" placeholder="Ціна" required><br><br>

  <input name="quantity" type="number" placeholder="Кількість" required><br><br>

  <input name="note" placeholder="Примітка"><br><br>

  <button type="submit">Додати</button>

</form>

<a href="index.php" style="display:block; margin-top:10px;">Назад до списку товарів</a>

<?php

try {

    $conn = new PDO("mysql:host=localhost;dbname=lab5", "root", "");

    $conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

    echo "Помилка підключення: " . $e->getMessage();

    exit;

}

// Виведення товарів у вигляді таблиці

$res = $conn->query("SELECT \* FROM tov");

echo "<table border='1' cellpadding='5'>";

echo "<tr><th>ID</th><th>Назва</th><th>Ціна</th><th>Кількість</th><th>Примітка</th></tr>";

foreach ($res as $row) {

    echo "<tr>";

    echo "<td>" . htmlspecialchars($row['id']) . "</td>";

    echo "<td>" . htmlspecialchars($row['name']) . "</td>";

    echo "<td>" . htmlspecialchars($row['price']) . " грн</td>";

    echo "<td>" . htmlspecialchars($row['quantity']) . "</td>";

    echo "<td>" . htmlspecialchars($row['note']) . "</td>";

    echo "</tr>";

}

echo "</table>";

// Форма для видалення запису

echo '<form action="delete.php" method="post" style="margin-top:20px;">

    <label>Номер запису для видалення: <input type="number" name="id\_to\_delete" required></label>

    <button type="submit">Вилучити запис</button>

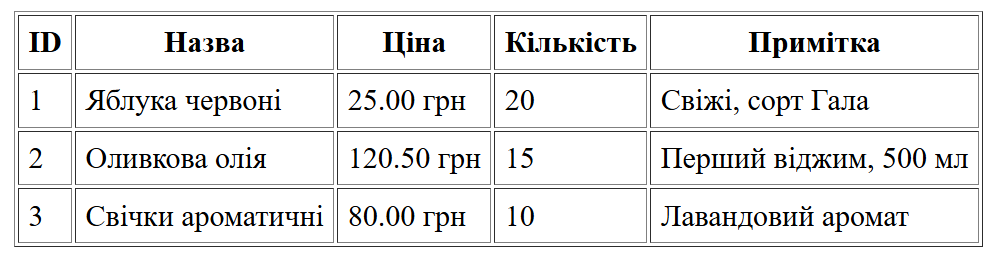
</form>';

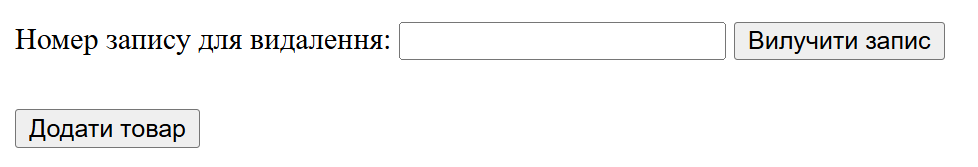
// Кнопка для переходу на додавання товару

echo '<a href="add\_product.php"><button style="margin-top:10px;">Додати товар</button></a>';

?>

Результат виконання:





Завдання 3:

1. Створіть базу даних MySQL за допомогою PhpMyAdmin і назвіть її "company\_db".
2. Створіть таблицю "employees" з наступними полями: id (INT, AUTO\_INCREMENT, PRIMARY KEY), name (VARCHAR), position (VARCHAR), і salary (DECIMAL).
3. Вставте кілька записів в таблицю "employees" через PhpMyAdmin.
4. Створіть PHP-сторінку для виведення всіх записів з таблиці "employees".
5. Реалізуйте можливість додавання нового працівника через веб-форму на PHP-сторінці.
6. Створіть функціонал для редагування існуючого запису працівника (зміна імені, посади або заробітної плати).
7. Реалізуйте можливість видалення запису про працівника з бази даних.
8. Створіть PHP-скрипт, який виводить статистику, наприклад, середню заробітну плату всіх працівників чи кількість працівників на кожній посаді.

Лістинг:

<?php

try {

    $pdo = new PDO('mysql:host=localhost;dbname=company\_db;charset=utf8', 'root', '');

    $pdo->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

    die("Помилка підключення: " . $e->getMessage());

}

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['add'])) {

    $stmt = $pdo->prepare("INSERT INTO employees (name, position, salary) VALUES (?, ?, ?)");

    $stmt->execute([$\_POST['name'], $\_POST['position'], $\_POST['salary']]);

    header('Location: employees.php');

    exit;

}

if (isset($\_GET['delete'])) {

    $stmt = $pdo->prepare("DELETE FROM employees WHERE id = ?");

    $stmt->execute([$\_GET['delete']]);

    header('Location: employees.php');

    exit;

}

if ($\_SERVER['REQUEST\_METHOD'] === 'POST' && isset($\_POST['edit'])) {

    $stmt = $pdo->prepare("UPDATE employees SET name = ?, position = ?, salary = ? WHERE id = ?");

    $stmt->execute([$\_POST['name'], $\_POST['position'], $\_POST['salary'], $\_POST['id']]);

    header('Location: employees.php');

    exit;

}

$editEmployee = null;

if (isset($\_GET['edit'])) {

    $stmt = $pdo->prepare("SELECT \* FROM employees WHERE id = ?");

    $stmt->execute([$\_GET['edit']]);

    $editEmployee = $stmt->fetch(PDO::FETCH\_ASSOC);

}

$employees = $pdo->query("SELECT \* FROM employees")->fetchAll(PDO::FETCH\_ASSOC);

$avgSalary = $pdo->query("SELECT AVG(salary) AS avg\_salary FROM employees")->fetch(PDO::FETCH\_ASSOC)['avg\_salary'];

if (is\_null($avgSalary)) {

    $avgSalary = 0;

}

$positionCounts = $pdo->query("SELECT position, COUNT(\*) AS count FROM employees GROUP BY position")->fetchAll(PDO::FETCH\_ASSOC);

?>

<!DOCTYPE html>

<html lang="uk">

<head>

    <meta charset="UTF-8">

    <title>Працівники компанії</title>

</head>

<body>

<h2>Список працівників</h2>

<table border="1" cellpadding="5" cellspacing="0">

    <tr>

        <th>ID</th><th>Ім'я</th><th>Посада</th><th>Зарплата</th><th>Дії</th>

    </tr>

    <?php foreach ($employees as $emp): ?>

    <tr>

        <td><?= htmlspecialchars($emp['id']) ?></td>

        <td><?= htmlspecialchars($emp['name']) ?></td>

        <td><?= htmlspecialchars($emp['position']) ?></td>

        <td><?= number\_format($emp['salary'], 2) ?> грн</td>

        <td>

            <a href="?edit=<?= $emp['id'] ?>">Редагувати</a> |

            <a href="?delete=<?= $emp['id'] ?>" onclick="return confirm('Видалити працівника?')">Видалити</a>

        </td>

    </tr>

    <?php endforeach; ?>

</table>

<hr>

<?php if ($editEmployee): ?>

<h3>Редагування працівника ID <?= $editEmployee['id'] ?></h3>

<form method="post">

    <input type="hidden" name="id" value="<?= $editEmployee['id'] ?>">

    <label>Ім'я: <input name="name" value="<?= htmlspecialchars($editEmployee['name']) ?>" required></label><br><br>

    <label>Посада: <input name="position" value="<?= htmlspecialchars($editEmployee['position']) ?>" required></label><br><br>

    <label>Зарплата: <input name="salary" type="number" step="0.01" value="<?= htmlspecialchars($editEmployee['salary']) ?>" required></label><br><br>

    <button type="submit" name="edit">Зберегти</button>

    <a href="employees.php">Скасувати</a>

</form>

<?php else: ?>

<h3>Додати нового працівника</h3>

<form method="post">

    <label>Ім'я: <input name="name" required></label><br><br>

    <label>Посада: <input name="position" required></label><br><br>

    <label>Зарплата: <input name="salary" type="number" step="0.01" required></label><br><br>

    <button type="submit" name="add">Додати</button>

</form>

<?php endif; ?>

<hr>

<h3>Статистика</h3>

<p>Середня заробітна плата: <?= number\_format($avgSalary, 2) ?> грн</p>

<p>Кількість працівників за посадами:</p>

<ul>

<?php foreach ($positionCounts as $pos): ?>

    <li><?= htmlspecialchars($pos['position']) ?>: <?= $pos['count'] ?></li>

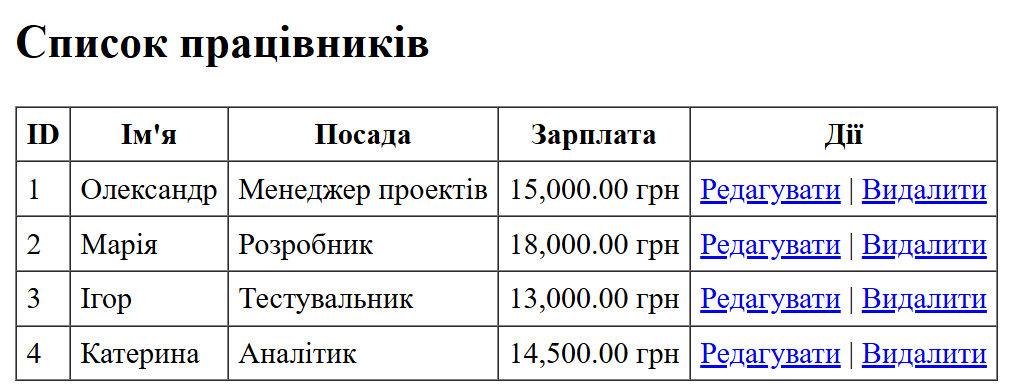
<?php endforeach; ?>

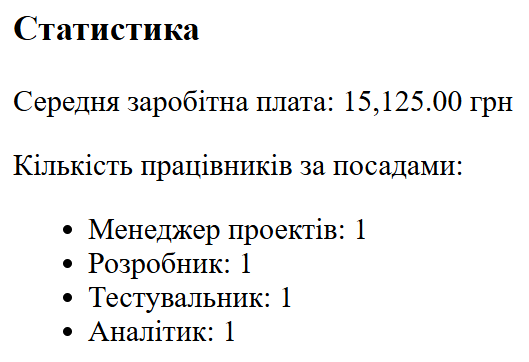
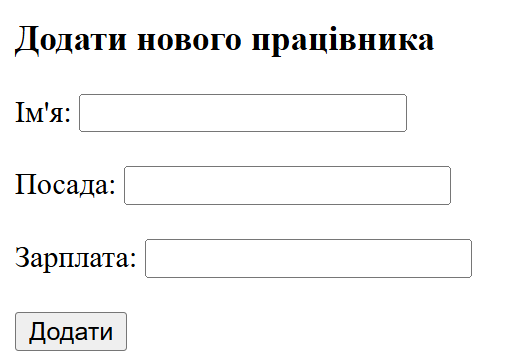
</ul>

</body>

</html>

Результат виконання:





**Висновок**

У ході виконання лабораторної роботи було створено базу даних company\_db з таблицею employees, що містить інформацію про працівників компанії: їхні імена, посади та заробітну плату. Було реалізовано веб-інтерфейс на PHP для виведення списку працівників, додавання нових записів, редагування та видалення існуючих.

Особливу увагу було приділено обробці виключень і коректній роботі з PDO для забезпечення безпеки та надійності взаємодії з базою даних. Також реалізовано статистичний модуль, який підраховує середню заробітну плату та кількість працівників на кожній посаді.

Дана лабораторна робота дозволила закріпити навички роботи з базами даних MySQL, використання PDO в PHP, створення форм для CRUD-операцій, а також обробку помилок та виведення статистичних даних.