

Лабораторна робота №6

Тема: Асинхронні запити на сервер та їх обробка

Мета роботи: ознайомитися з асинхронними запитами на сервер та їх обробкою

Завдання на лабораторну роботу

Завдання 1: Система керування користувачами

Опис завдання: Створіть веб-додаток для системи керування користувачами, який використовує асинхронні запити для взаємодії між клієнтом та сервером. У цьому завданні передбачено дві основні функціональності: реєстрація нового користувача та перегляд списку користувачів.

1. Реєстрація користувача (фронтенд):

Створіть HTML-форму для реєстрації нового користувача з полями для ім'я, електронної адреси та пароля.

Використовуйте JavaScript для збору даних з форми та відправки їх на сервер за допомогою fetch API.

Перевірте, чи всі поля заповнені перед відправкою на сервер.

2. Реєстрація користувача (бекенд):

Створіть PHP-скрипт для обробки запиту на реєстрацію нового користувача.

Перевірте правильність введених даних, такі як унікальність електронної адреси та мінімальна довжина пароля.

Збережіть дані нового користувача в базі даних MySQL

3. Отримання списку користувачів (фронтенд):

Створіть кнопку або посилання для відображення списку користувачів.

Використовуйте JavaScript для відправки запиту на сервер для отримання списку користувачів.

Виведіть отриманий список користувачів на веб-сторінці.

4. Отримання списку користувачів (бекенд):

Створіть PHP-скрипт, який вибирає всіх користувачів з бази даних.

Поверніть список користувачів у форматі JSON, щоб його можна було легко обробити на клієнті.

					ДУ «Житомирська політехніка».24.121.05.000 – Лр6			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Кравченко К.М.			Звіт з лабораторної роботи		Літ.	Арк.
Перевір.		Ковтун В.В.						1
Керівник								15
Н. контр.							ФІКТ Гр. ІПЗ-22-2[2]	
Зав. каф.								

Додайте можливість входу користувача за допомогою електронної адреси та пароля.
Реалізуйте можливість видалення користувача.
Додайте можливість редагування даних користувача.

Результат:

<input type="checkbox"/>	1	id	int	Нет	Нет	AUTO_INCREMENT		Изменить		Удалить		Ещё
<input type="checkbox"/>	2	name	varchar(30)	utf8mb4_general_ci	Да	NULL		Изменить		Удалить		Ещё
<input type="checkbox"/>	3	email	varchar(50)	utf8mb4_general_ci	Нет	Нет		Изменить		Удалить		Ещё
<input type="checkbox"/>	4	password	varchar(255)	utf8mb4_general_ci	Нет	Нет		Изменить		Удалить		Ещё

Реєстрація нового користувача

Ім'я:

Електронна адреса:

Пароль:

Зареєструватися

Показати список користувачів

Список користувачів порожній

Електронна адреса:

Пароль:

Увійти

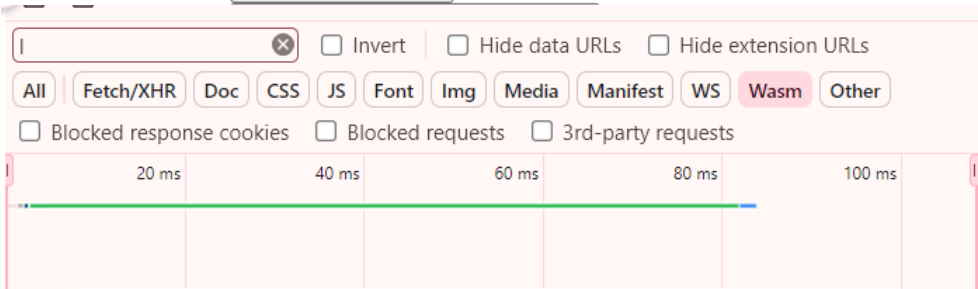
Реєструємо нового користувача:

Ім'я:

Електронна адреса:

Пароль:

Зареєструватися



Повідомлення з backendlab6

Користувач успішно зареєстрований



Список користувачів:

Показати список користувачів

Ім'я Електронна адреса

Катерина ipz222@gmail.com

Дії

Видалити

Редагувати

<div><div>←</div><div>→</div></div>			▼	id	name	email	password
<input type="checkbox"/>	<div><div></div><div>Изменить</div></div>	<div><div></div><div>Копировать</div></div>	<div><div></div><div>Удалить</div></div>	8	Катерина	ipz222@gmail.com	\$2y\$10\$1.X241N79E2uHmLXguksROLZkcCB.wnbCXCDJf3V3.r...

Змінимо:

Повідомлення з backendlab6

Введіть нове ім'я для користувача:

Катя

ОК

Скасувати

Повідомлення з backendlab6

Введіть нову електронну адресу для користувача:

ipz222_kkm@gmail.com

ОК

Скасувати

Повідомлення з backendlab6

Введіть новий пароль для користувача:

1212121212

ОК

Скасувати

Повідомлення з backendlab6

Дані користувача успішно оновлено.

ОК

Ім'я Електронна адреса

Катя ipz222_kkm@gmail.com

Дії

Видалити

Редагувати

<div>← T →</div>						
<div><div><div><div></div><div>id</div></div><div><div></div><div>name</div></div><div><div></div><div>email</div></div><div><div></div><div>password</div></div></div></div>						
<div><div><div><div></div><div>Изменить</div></div><div><div></div><div>Копировать</div></div><div><div></div><div>Удалить</div></div></div></div>	8	Катя	ipz222_kkm@gmail.com	\$2y\$10\$gKyTKn9f/7Y/NOUIg7aReO9IzK3yBIL6T8COaKEVGyZ...		

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

Тепер видалим користувача:

Повідомлення з backendlab6

Ви впевнені, що хочете видалити цього користувача?

ОК

Скасувати

Повідомлення з backendlab6

Користувача успішно видалено.

ОК

Список користувачів порожній

✓ MySQL вернула пустой результат (т.е. ноль строк). (Запрос занял 0,0002 сек.)

`SELECT * FROM `users``

☐ Профилирование ☐ Построчное редактирование ☐ Изменить ☐ Анализ SQL запроса ☐ Создать PHP-код ☐ Обновить

id	name	email	password
----	------	-------	----------

Использование результатов запроса

Лістинг: Index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Реєстрація нового користувача</title>
</head>
<body>
<h2>Реєстрація нового користувача</h2>
<form id="registrationForm" action="register.php" method="post">
  <label for="name">Ім'я:</label><br>
  <input type="text" id="name" name="name" required><br>
  <label for="email">Електронна адреса:</label><br>
  <input type="email" id="email" name="email" required><br>
  <label for="password">Пароль:</label><br>
  <input type="password" id="password" name="password" required><br><br>
  <button type="submit">Зареєструватися</button>
</form>

<button onclick="getUsers()">Показати список користувачів</button>
<ul id="userList"></ul>

<form id="loginForm" method="post">
  <label for="loginEmail">Електронна адреса:</label><br>
  <input type="email" id="loginEmail" name="emailLog" required><br>
  <label for="loginPassword">Пароль:</label><br>
  <input type="password" id="loginPassword" name="passwordLog" required><br><br>
  <button type="button" onclick="loginUser()">Увійти</button>
</form>

<script src="script.js"></script>
</body>
</html>
```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

Script.js:

```
document.getElementById("registrationForm").addEventListener("submit", function(event) {
    event.preventDefault();

    const formData = new FormData(event.target);

    if (!formData.get('name') || !formData.get('email') || !formData.get('password')) {
        alert('Будь ласка, заповніть всі поля');
        return;
    }

    fetch('register.php', {
        method: 'POST',
        body: formData
    })
        .then(response => {
            if (!response.ok) {
                throw new Error('Network response was not ok');
            }
            return response.json();
        })
        .then(data => {
            console.log(data);
            alert('Користувач успішно зареєстрований');
            event.target.reset();
            getUsers();
        })
        .catch(error => {
            console.error('There was a problem with your fetch operation:', error);
            alert('Помилка реєстрації користувача');
        });
});

function getUsers() {
    fetch('get_users.php')
        .then(response => response.json())
        .then(users => {
            const userList = document.getElementById("userList");
            userList.innerHTML = '';

            if (users.length > 0) {
                const table = document.createElement('table');
                table.innerHTML = `
                    <tr>
                        <th>Ім'я</th>
                        <th>Електронна адреса</th>
                        <th>Дії</th>
                    </tr>
                `;

                users.forEach(user => {
                    const row = document.createElement('tr');
                    row.innerHTML = `
                        <td>${user.name}</td>
                        <td>${user.email}</td>
                        <td>
                            <button on-
click="deleteUser(${user.id})">Видалити</button>
                            <button onclick="editUser(${user.id}, '${user.name}',
'${user.email}', '${user.password}')">Редагувати</button>
                        </td>
                    `;
                });
            }
        });
}
```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        `;
        table.appendChild(row);
    });

    userList.appendChild(table);
} else {
    userList.textContent = "Список користувачів порожній";
}
})
.catch(error => {
    console.error('Помилка при отриманні списку користувачів:', error);
});
}

function deleteUser(userId) {
    if (confirm("Ви впевнені, що хочете видалити цього користувача?")) {
        fetch('delete_user.php', {
            method: 'POST',
            body: JSON.stringify({ userId }),
            headers: {
                'Content-Type': 'application/json'
            }
        })
        .then(response => response.json())
        .then(data => {
            console.log(data);
            alert(data.message);
            getUsers();
        })
        .catch(error => {
            console.error('Помилка видалення користувача:', error);
            alert('Помилка видалення користувача');
        });
    }
}

function editUser(userId) {
    if (!userId) {
        alert("Неможливо відредагувати користувача: невірний ідентифікатор користувача");
        return;
    }

    const newName = prompt("Введіть нове ім'я для користувача:");
    if (newName !== null) {
        const newEmail = prompt("Введіть нову електронну адресу для користувача:");
        const newPassword = prompt("Введіть новий пароль для користувача:");
        fetch('edit_user.php', {
            method: 'POST',
            headers: {
                'Content-Type': 'application/json'
            },
            body: JSON.stringify({ userId, newName, newPassword, newEmail })
        })
        .then(response => response.json())
        .then(data => {
            console.log(data);
            alert(data.message);
            getUsers();
        })
        .catch(error => {
            console.error('Помилка редагування користувача:', error);
            alert('Помилка редагування користувача');
        });
    }
}

```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				6
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    });
}
}
function loginUser() {
    const loginEmail = document.getElementById("loginEmail").value;
    const loginPassword = document.getElementById("loginPassword").value;

    const userData = {
        email: loginEmail,
        password: loginPassword
    };

    fetch('login.php', {
        method: 'POST',
        headers: {
            'Content-Type': 'application/json'
        },
        body: JSON.stringify(userData)
    })
        .then(response => response.json())
        .then(data => {
            if (data.success) {
                alert('Ви успішно увійшли!');
                getUsers();
            } else {
                alert(data.error);
            }
        })
        .catch(error => {
            console.error('Помилка авторизації:', error);
            alert('Виникла помилка при авторизації. Будь ласка, спробуйте ще раз.');
```

раз.');

```

        });
    }

    getUsers();
}

```

login.php:

```

<?php
header('Content-Type: application/json');

try {
    $conn = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $json_str = file_get_contents('php://input');
    $data = json_decode($json_str, true);

    $email = $data['email'];
    $password = $data['password'];

    $stmt = $conn->prepare("SELECT * FROM users WHERE email = ?");
    $stmt->execute([$email]);
    $user = $stmt->fetch();

    if (!$user) {
        throw new Exception('Невірна електронна адреса');
    }

    if (!password_verify($password, $user['password'])) {
        throw new Exception('Невірна електронна адреса або пароль.');
```

```

    }

    echo json_encode(['success' => true]);
}

```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

} catch (Exception $e) {
    echo json_encode(['error' => $e->getMessage()]);
}
?>

```

Register.php:

```

<?php
$host = 'localhost';
$dbname = 'lab6';
$username = 'root';
$password = '';

try {
    $conn = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e) {
    echo "Connection failed: " . $e->getMessage();
    exit;
}

$name = $_POST['name'];
$email = $_POST['email'];
$password = $_POST['password'];

if(empty($name) || empty($email) || empty($password)) {
    echo json_encode(array("message" => "All fields are required"));
    exit;
}

if(strlen($password) < 6) {
    echo json_encode(array("message" => "Password should be at least 6 characters long"));
    exit;
}

$stmt = $conn->prepare("SELECT * FROM users WHERE email = :email");
$stmt->execute(array(':email' => $email));
if($stmt->rowCount() > 0) {
    echo json_encode(array("message" => "Email already exists"));
    exit;
}

$hashedPassword = password_hash($password, PASSWORD_DEFAULT);

$stmt = $conn->prepare("INSERT INTO users (name, email, password) VALUES (:name, :email, :password)");
$stmt->execute(array(':name' => $name, ':email' => $email, ':password' => $hashedPassword));

if($stmt) {
    echo json_encode(array("message" => "New record created successfully"));
} else {
    echo json_encode(array("message" => "Error: Unable to insert record"));
}

$conn = null;
?>

```

Get_users.php:

```

<?php
$host = 'localhost';
$dbname = 'lab6';
$username = 'root';
$password = '';

try {

```

К.


```

    $conn = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e) {
    echo "Connection failed: " . $e->getMessage();
    exit;
}

$stmt = $conn->query("SELECT id, name, email FROM users");
$users = $stmt->fetchAll(PDO::FETCH_ASSOC);

header('Content-Type: application/json');
echo json_encode($users);
?>

```

Edit_user.php:

```

<?php
header('Content-Type: application/json');

try {
    $conn = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');

    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $json_data = file_get_contents('php://input');
    $data = json_decode($json_data, true);

    $hashed_password = password_hash($data['newPassword'], PASSWORD_DEFAULT);

    $stmt = $conn->prepare("UPDATE users SET name = ?, email = ?, password = ?
WHERE id = ?");
    $stmt->execute([ $data['newName'], $data['newEmail'], $hashed_password, $data['userId']]);

    echo json_encode(['success' => true, 'message' => 'Дані користувача успішно оновлено.']);
} catch (Exception $e) {
    echo json_encode(['error' => $e->getMessage()]);
}

```

Delete_user.php:

```

<?php
header('Content-Type: application/json');

try {
    $conn = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $json_data = file_get_contents('php://input');
    $data = json_decode($json_data, true);

    $stmt = $conn->prepare("DELETE FROM users WHERE id = ?");
    $stmt->execute([$data['userId']]);

    echo json_encode(['success' => true, 'message' => 'Користувача успішно видалено.']);
} catch (Exception $e) {
    echo json_encode(['error' => $e->getMessage()]);
}

```

Завдання 2: Реалізація заміток (Notes App) за допомогою FETCH та PHP

Опис завдання: Створіть веб-додаток, який дозволяє користувачам додавати, переглядати, оновлювати та видаляти замітки. Користувачі можуть створювати замітки, вказуючи заголовок та текст. Всі замітки повинні зберігатися на сервері.

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

Взаємодія між клієнтом та сервером повинна здійснюватися за допомогою асинхронних запитів та PHP на серверній стороні.

Ключові вимоги:

Додавання заміток: Користувач повинен мати можливість додавати нові замітки, вказуючи заголовок та текст, та надсилати їх на сервер за допомогою асинхронних запитів.

Перегляд заміток: Усі замітки, збережені на сервері, повинні відображатися на головній сторінці.

Оновлення заміток: Користувач повинен мати можливість оновлювати замітки, змінюючи їх заголовок або текст. Зміни повинні відображатися на сервері після натискання відповідної кнопки або події.

Видалення заміток: Користувач повинен мати можливість видаляти замітки за допомогою відповідної опції або кнопки. Видалення заміток повинно бути асинхронним, без перезавантаження сторінки.

Валідація даних на клієнті та сервері: Перед надсиланням даних на сервер, вони повинні бути перевірені на клієнті (заголовок та текст мають бути заповнені).

Відправка та отримання даних за допомогою fetch API: Взаємодія з сервером повинна здійснюватися за допомогою fetch API на клієнтській стороні для відправки та отримання даних.

Зберігання даних на сервері з використанням PHP: Замітки повинні зберігатися на сервері за допомогою PHP. Використовуйте MySQL або іншу базу даних для зберігання інформації про замітки.

Асинхронність та безперервність додатку: Усі операції (додавання, оновлення, видалення заміток) повинні відбуватися без перезавантаження сторінки, з використанням асинхронних запитів.

Результат:

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	int		Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
<input type="checkbox"/>	2	title	varchar(255)	utf8mb4_general_ci	Нет	Нет			Изменить Удалить Ещё
<input type="checkbox"/>	3	text	text	utf8mb4_general_ci	Нет	Нет			Изменить Удалить Ещё

Notes App

Note Title

Note Content

Add Note

Нотатка 1

Здати 6 лабу

Update

Delete

Змінимо:

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				10
Змн.	Арк.	№ докум.	Підпис	Дата		

Повідомлення з backendlab6

Enter the new title:

Нотатка 2

OK

Скасувати

Нотатка 1

Здати 6 лабу

Add Note

Нотатка 1

Здати 6 лабу

Update

Delete

Повідомлення з backendlab6

Enter the new text:

Розпочати лаб 7

OK

Скасувати

Нотатка 2

Розпочати лаб 7

Update

Delete

←T→


▼

id


title

text


☐



Изменить



Копировать



Удалить

4

Нотатка 2

Розпочати лаб 7

*Лістинг:
Index.html:*

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Notes App</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
    }

    header {
      background-color: #646547;
      color: #fff;
      text-align: center;
      padding: 20px 0;
    }

    main {
      max-width: 800px;
      margin: 20px auto;
```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        padding: 0 20px;
    }

    form {
        margin-bottom: 20px;
    }

    input[type="text"],
    textarea {
        width: 100%;
        padding: 10px;
        margin-bottom: 10px;
    }

    button {
        background-color: #646547;
        color: #fff;
        border: none;
        padding: 10px 20px;
        cursor: pointer;
    }

    button:hover {
        background-color: #3e3f23;
    }

    .note {
        border: 1px solid #ccc;
        border-radius: 5px;
        padding: 10px;
        margin-bottom: 10px;
    }

    .note h2 {
        margin-top: 0;
    }

    .note p {
        margin-bottom: 0;
    }

    .note button {
        margin-left: 10px;
    }
</style>
</head>
<body>
<header>
    <h1>Notes App</h1>
</header>
<main>
    <form id="noteForm">
        <input type="text" id="noteTitle" placeholder="Note Title" required>
        <textarea id="noteContent" placeholder="Note Content" required></textarea>
        <button type="submit">Add Note</button>
    </form>
    <div id="notesList"></div>
</main>

<script src="script.js"></script>
</body>
</html>

```

Script.js:

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

```

document.addEventListener("DOMContentLoaded", () => {
    const noteForm = document.getElementById("noteForm");
    const notesList = document.getElementById("notesList");
    function fetchAndDisplayNotes() {
        fetch("fetch_notes.php")
            .then(response => response.json())
            .then(notes => {
                notesList.innerHTML = "";
                notes.forEach(note => {
                    const noteElement = document.createElement("div");
                    noteElement.innerHTML = `
                        <h2>${note.title}</h2>
                        <p>${note.text}</p>
                        <button onclick="updateNote(${note.id})">Update</button>
                        <button onclick="deleteNote(${note.id})">Delete</button>
                    `;
                    notesList.appendChild(noteElement);
                });
            });
    }
    fetchAndDisplayNotes();
    noteForm.addEventListener("submit", event => {
        event.preventDefault();
        const title = document.getElementById("noteTitle").value;
        const text = document.getElementById("noteContent").value;
        fetch("add_note.php", {
            method: "POST",
            headers: {
                "Content-Type": "application/json"
            },
            body: JSON.stringify({ title, text })
        })
            .then(response => {
                if (response.ok) {
                    fetchAndDisplayNotes();
                } else {
                    console.error("Failed to add note");
                }
            })
            .catch(() => {});
    });
    window.updateNote = id => {
        const newTitle = prompt("Enter the new title:");
        const newText = prompt("Enter the new text:");
        if (newTitle !== null && newText !== null) {
            fetch("update_note.php", {
                method: "POST",
                headers: {
                    "Content-Type": "application/json"
                },
                body: JSON.stringify({ id, title: newTitle, text: newText })
            })
                .then(response => {
                    if (response.ok) {
                        fetchAndDisplayNotes();
                    } else {
                        console.error("Failed to update note");
                    }
                })
                .catch(() => {});
        }
    };
    window.deleteNote = id => {
        if (confirm("Are you sure you want to delete this note?")) {
            fetch("delete_note.php", {
                method: "POST",
            })
                .then(response => {
                    if (response.ok) {
                        fetchAndDisplayNotes();
                    } else {
                        console.error("Failed to delete note");
                    }
                })
                .catch(() => {});
        }
    };
});

```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				13
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        headers: {
            "Content-Type": "application/json"
        },
        body: JSON.stringify({ id })
    })

    .then(response => {
        if (response.ok) {
            fetchAndDisplayNotes();
        } else {
            console.error("Failed to delete note");
        }
    });
}

};
});

```

fetch_notes.php:

```

<?php
try {
    $pdo = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');

    $stmt = $pdo->query("SELECT * FROM notes");
    $notes = $stmt->fetchAll(PDO::FETCH_ASSOC);

    header('Content-Type: application/json');
    echo json_encode($notes);
} catch (PDOException $e) {
    die("Error: " . $e->getMessage());
}
?>

```

Add_note.php:

```

<?php
header('Content-Type: application/json');
try {
    $pdo = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');

    $data = json_decode(file_get_contents("php://input"), true);
    $title = $data["title"];
    $text = $data["text"];

    $stmt = $pdo->prepare("INSERT INTO notes (title, text) VALUES (:title, :text)");
    $stmt->execute(['title' => $title, 'text' => $text]);

    http_response_code(200);
} catch (PDOException $e) {
    http_response_code(500);
    die("Error: " . $e->getMessage());
}
?>

```

Update_note.php:

```

<?php
try {
    $pdo = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');
    $data = json_decode(file_get_contents("php://input"), true);
    $id = $data["id"];
    $title = $data["title"];
    $text = $data["text"];
    $stmt = $pdo->prepare("UPDATE notes SET title = :title, text = :text WHERE id = :id");
    $stmt->execute(['title' => $title, 'text' => $text, 'id' => $id]);
    http_response_code(200);
} catch (PDOException $e) {
    http_response_code(500);
}

```

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        die("Error: " . $e->getMessage());
    }
?>

```

Delete_note.php:

```

<?php
try {
    $pdo = new PDO("mysql:host=localhost;dbname=lab6", 'root', '');
    $data = json_decode(file_get_contents("php://input"), true);
    $id = $data["id"];
    $stmt = $pdo->prepare("DELETE FROM notes WHERE id = :id");
    $stmt->execute(['id' => $id]);
    http_response_code(200);
} catch (PDOException $e) {
    http_response_code(500);
    die("Error: " . $e->getMessage());
}
?>

```

Висновок: ознайомилась з асинхронними запитами на сервер та їх обробкою.

		Кравченко К.М.			ДУ «Житомирська політехніка».24.121.05.000 – Лр6	Арк.
		Ковтун В.В.				15
Змн.	Арк.	№ докум.	Підпис	Дата		