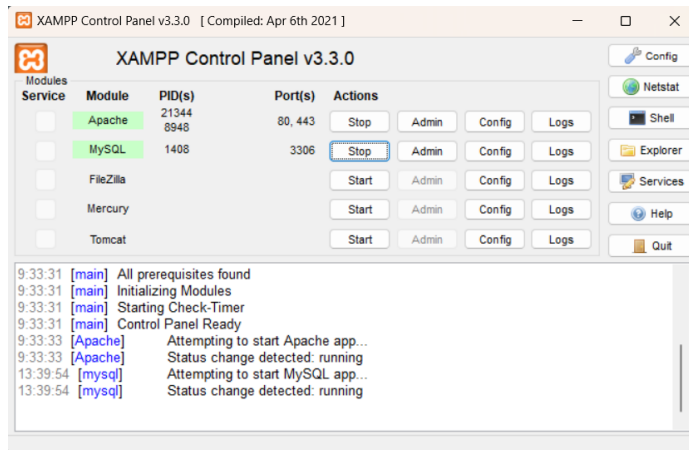


Лабораторна робота №12

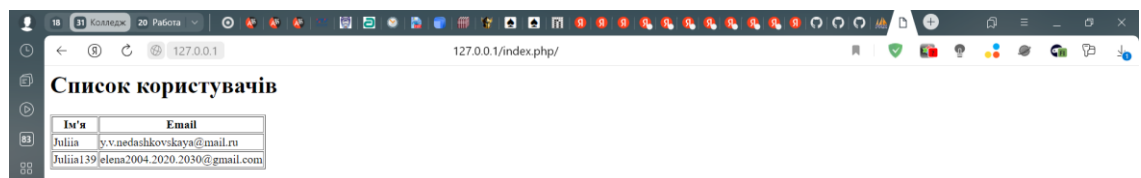
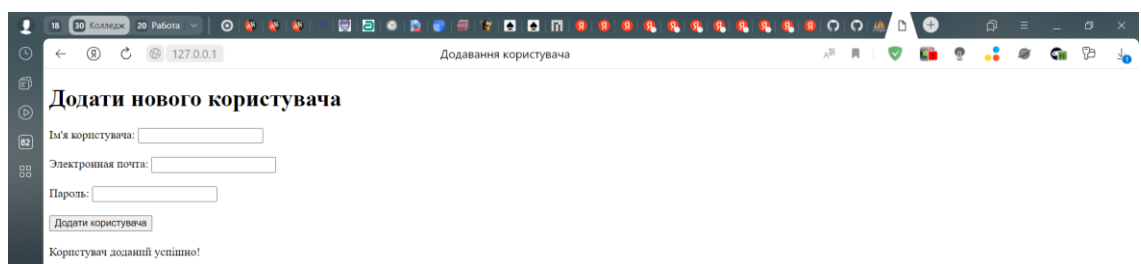
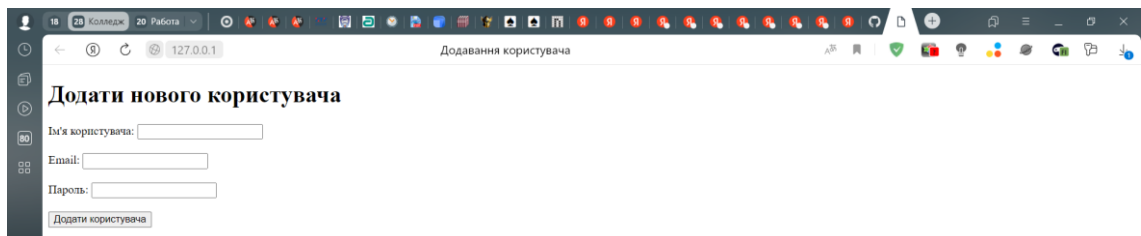
Тема: Використання PDO для обміну даними з СКБД.

Хід роботи

1. Запустіть web-сервер Apache та СУБД MySQL у вікні прикладки XAMPP Control Panel.

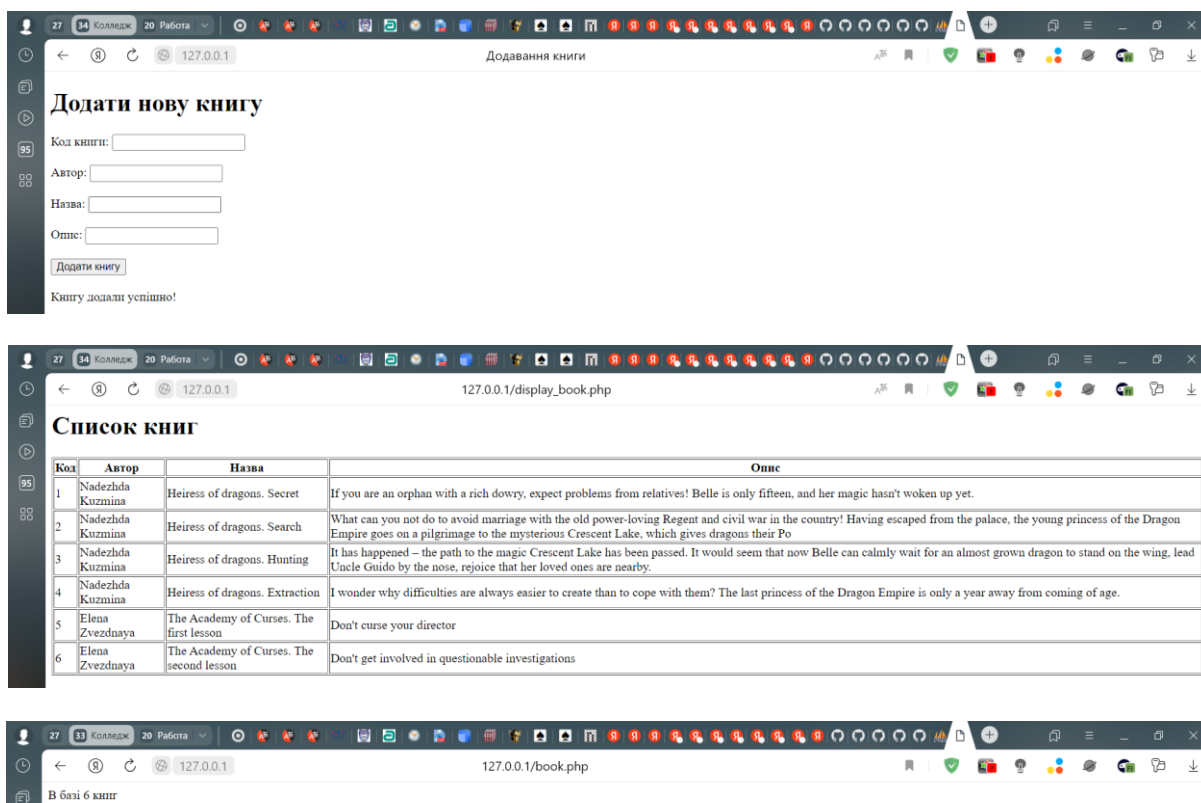


2. Перейдіть до каталогу C:\xampp\htdocs\ та очистіть його.
3. Виконайте приклади наведені нижче. Зверніть увагу на дані підключення до СКБД!



					Лабораторна робота №12			
Зм	Лист	№ докум	Підпис	Дата	Використання PDO для обміну даними з СКБД	Літ.	Лист.	Листів
Розробив	<i>Недашковська Ю.В.</i>		<i>Ю.В.</i>			У	1	6
Перевірив	<i>Левицький В.М.</i>					Група 451		
Оцінка								
Затв.								

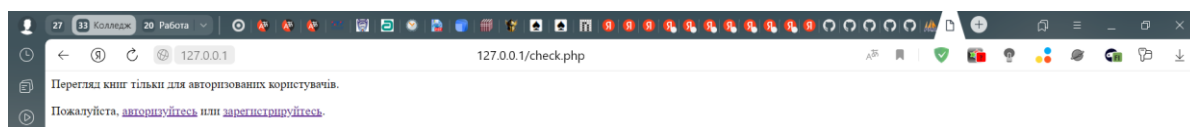
4. Опрацюйте лабораторні роботи № 8 та № 11.
5. У вже виконаних завданнях змініть драйвер бази даних з MySQLi на PDO.



6. Відредагуйте запити до СКБД так, щоб вони працювали з драйвером PDO.
7. Додайте до вашої роботи таблицю з користувачами, що містить наступні дані:
 - Ідентифікатор користувача (PK).
 - Ім'я входу - логін (унікальне).
 - Прізвище та ім'я.
 - Електронна пошта.
 - Пароль.

#	Ім'я	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
2	login	varchar(60)	latin1_swedish_ci		Нет	Нет			Изменить Удалить Ещё
3	surname	varchar(60)	latin1_swedish_ci		Нет	Нет			Изменить Удалить Ещё
4	email	tinytext	latin1_swedish_ci		Нет	Нет			Изменить Удалить Ещё
5	password	varchar(60)	latin1_swedish_ci		Нет	Нет			Изменить Удалить Ещё

8. Додайте до вашої роботи можливість перегляду списку книг тільки для авторизованих користувачів.



9. Додайте можливість виходу з авторизованої сесії.

Зм.	Арк.	№ докум.	Підпис	Дата

Лабораторна робота № 12

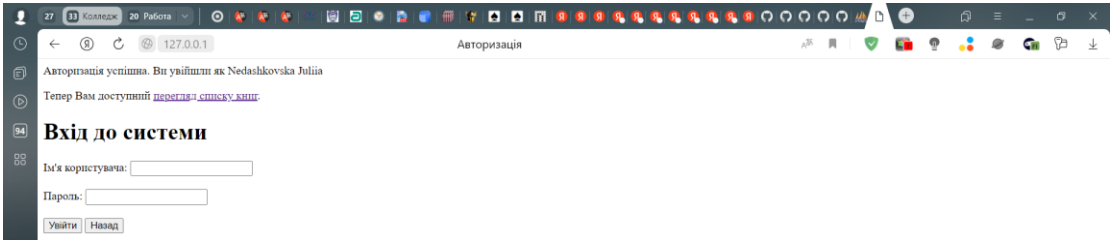
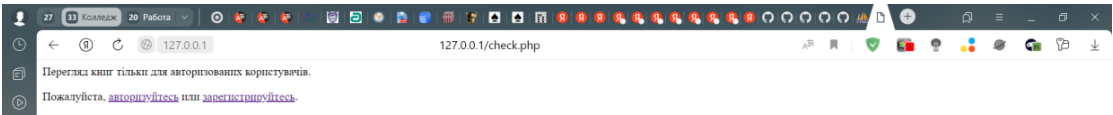
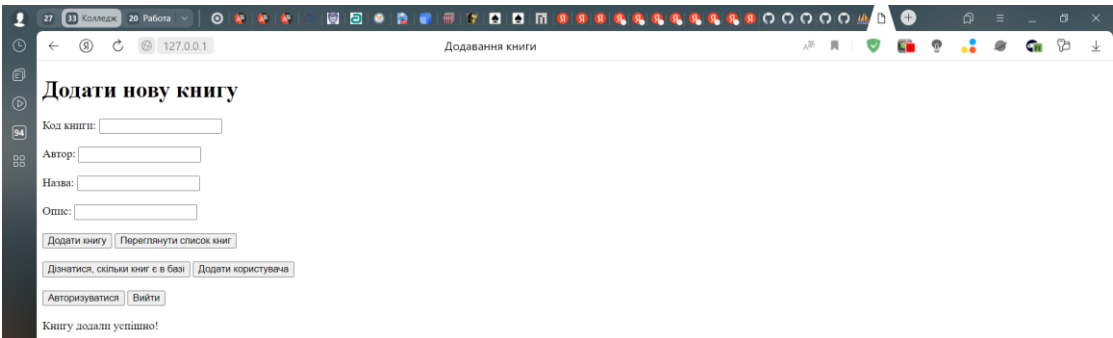
Ар
2

```
C:\xampp\htdocs\logout.php - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
index.php | logout.php | check.php | display_book.php | book.php | authorization.php | newuser.php
1 <?php
2 // Запускаем сессию
3 session_start();
4 // Удаляем все переменные сессии
5 session_unset();
6 // Уничтожаем сессию
7 session_destroy();
8 // Перенаправляем пользователя на главную страницу
9 header("Location: index.php");
10 ?>
```

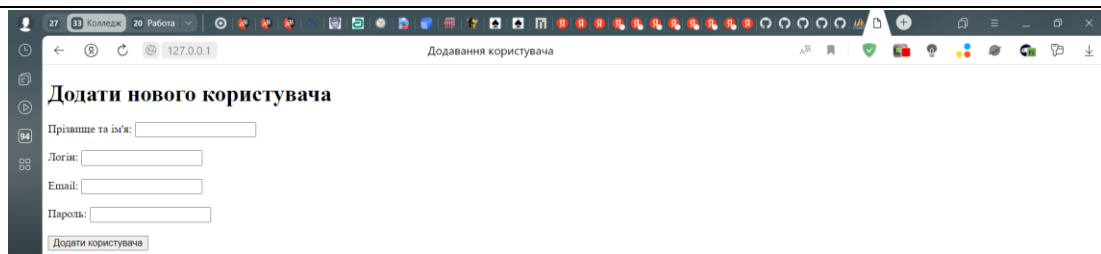
10. Додайте до вашої роботи елемент безпеки: зробіть так, щоб пароль у базі даних зберігався у вигляді хешу. Можна використати MD5, але зверніть увагу, що наразі дана функція не вважається криптостійкою, тому бажано використати іншу.

	id	login	surname	email	password
<input type="checkbox"/> Изменить Копировать Удалить	33	Julia139	Nedashkovska Julia	y.v.nedashkovskaya@mail.ru	\$2y\$10\$OtO3jg04mEYmbeUIAq7iBOSUZdEjXb9/Mssze71wMaD...

11. Для кожного етапу роботи зробити знімки екрану та додати їх у звіт з описом кожного скріншота.



Список книг			
Код	Автор	Назва	Опис
1	Nadezhda Kuzmina	Heiress of dragons. Secret	If you are an orphan with a rich dowry, expect problems from relatives! Belle is only fifteen, and her magic hasn't woken up yet.
2	Nadezhda Kuzmina	Heiress of dragons. Search	What can you not do to avoid marriage with the old power-loving Regent and civil war in the country! Having escaped from the palace, the young princess of the Dragon Empire goes on a pilgrimage to the mysterious Crescent Lake, which gives dragons their Po
3	Nadezhda Kuzmina	Heiress of dragons. Hunting	It has happened – the path to the magic Crescent Lake has been passed. It would seem that now Belle can calmly wait for an almost grown dragon to stand on the wing, lead Uncle Guido by the nose, rejoice that her loved ones are nearby.
4	Nadezhda Kuzmina	Heiress of dragons. Extraction	I wonder why difficulties are always easier to create than to cope with them? The last princess of the Dragon Empire is only a year away from coming of age.
5	Elena Zvezdnaya	The Academy of Curses. The first lesson	Don't curse your director
6	Elena Zvezdnaya	The Academy of Curses. The second lesson	Don't get involved in questionable investigations
7	Elena Zvezdnaya	The Academy of Curses. The third lesson	Secrets can be deadly



12. Додати програмний код завдання для самостійного виконання.

13. Дати відповіді на контрольні запитання.

14. Зберегти звіт у форматі PDF.

Контрольні питання

1. Що таке PDO і для чого воно використовується в PHP?

PDO (PHP Data Objects) – це розширення для PHP, що надає розробнику простий і універсальний інтерфейс для доступу до різних баз даних. Воно використовується для виконання SQL-запитів, обробки результатів, управління транзакціями та обробки помилок при роботі з базами даних.

2. Які переваги надає використання PDO в порівнянні з іншими методами взаємодії з базами даних?

Переваги використання PDO в порівнянні з іншими методами взаємодії з базами даних полягають в наступному:

- PDO пропонує єдині методи для роботи з різними базами даних, хоча текст запитів може трохи відрізнятися. Це дозволяє легко перейти на іншу СУБД, при цьому не змінюючи або частково змінюючи код програми.
- PDO не використовує абстрактних прошарків для підключення до БД, на зразок ODBC, а використовує для різних БД їх «рідні» драйвери, що дозволяє добитися високої продуктивності.
- PDO підтримує параметризовані запити, які забезпечують безпеку від SQL-ін'єкцій та підвищують продуктивність за рахунок повторного використання підготовлених запитів².
- PDO дозволяє виводити результати запиту у вигляді асоціативного масиву, нумерованого масиву, об'єктів або комбінації цих форматів².
- PDO дозволяє використовувати різні режими обробки помилок, включаючи винятки, які спрощують відлагодження та контроль за виконанням запитів².

3. Які типи баз даних підтримуються PDO?

Типи баз даних, які підтримуються PDO, залежать від наявності відповідних драйверів. В даний час для PDO існують драйвери практично для всіх загальновідомих СУБД та інтерфейсів. Ось деякі з них: FreeTDS, Microsoft SQL Server, Sybase, Firebird, Interbase 6, SQLite 3 і SQLite 2.

4. Як створити підключення до бази даних за допомогою PDO?

					Лабораторна робота № 12	Ар
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

Для створення підключення до бази даних за допомогою PDO потрібно створити екземпляр класу PDO, передавши йому параметри, які визначають тип драйвера, ім'я хоста, ім'я бази даних, ім'я користувача та пароль.

5. Як відбувається обробка помилок та винятків при використанні PDO?

Обробка помилок та винятків при використанні PDO залежить від режиму обробки помилок, який можна встановити за допомогою методу `setAttribute` класу PDO. Існують три режими обробки помилок:

- `PDO::ERRMODE_SILENT`.
- `PDO::ERRMODE_WARNING`.
- `PDO::ERRMODE_EXCEPTION`.

6. Що таке параметризовані запити і як їх використовувати за допомогою PDO?

Параметризовані запити – це запити, в яких значення параметрів вказуються окремо від тексту запиту. Це дозволяє запобігти SQL-ін'єкціям, оскільки значення параметрів не впливають на структуру запиту, а також підвищує продуктивність, оскільки один і той же запит може бути використаний багато разів з різними значеннями параметрів². Для використання параметризованих запитів за допомогою PDO можна використати два способи.

7. Як використовувати підготовлені запити для підвищення безпеки та продуктивності?

Підготовлені запити – це запити, які підготовлюються один раз і потім використовуються багато разів з різними значеннями параметрів. Вони дозволяють підвищити безпеку та продуктивність за рахунок наступного:

- Підготовлені запити запобігають SQL-ін'єкціям, оскільки значення параметрів не впливають на структуру запиту, а передаються окремо.
- Підготовлені запити зменшують навантаження на сервер, оскільки запит не потребує повторної компіляції та оптимізації при кожному виконанні.
- Підготовлені запити забезпечують кращу переносимість коду, оскільки вони автоматично адаптуються до різних типів даних та форматів баз даних.

8. Як вивести результати запиту у вигляді асоціативного масиву або об'єктів з PDO?

Для виведення результатів запиту у вигляді асоціативного масиву або об'єктів з PDO можна використати наступні способи:

- Вказати режим виведення результатів при підготовці запиту за допомогою другого параметра методу `prepare` класу PDO.
- Вказати режим виведення результатів при виконанні запиту за допомогою другого параметра методу `execute` класу `PDOStatement`.
- Вказати режим виведення результатів при отриманні результатів за допомогою другого параметра методів `fetch` або `fetchAll` класу `PDOStatement`.

					<i>Лабораторна робота № 12</i>	Ар
						2
Зм.	Арк.	№ докум.	Підпис	Дата		

- Встановити режим виведення результатів за замовчуванням для всіх запитів за допомогою методу `setAttribute` класу PDO.

					<i>Лабораторна робота № 12</i>	Ар
						2
Зм.	Арк.	№ докум.	Підпис	Дата		