

Хід роботи

Додати книгу

Назва книги:

Автор:

Рік видання:

127.0.0.1/add_book.php
Помилка підключення: SQLSTATE[HY000] [1045] Access denied for user 'hack'@'localhost' (using password: YES)Помилка підключення: SQLSTATE[HY000] [1045] Access denied for user 'hack'@'localhost' (using password: YES)

Контрольні питання:

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

PDO - це розширення PHP, яке надає уніфікований спосіб доступу до різних джерел даних, зокрема реляційних баз даних.

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

- Універсальність
- Безпека
- Читання результатів

3. Які типи баз даних підтримуються PDO?

PDO підтримує багато різних СУБД, такі як MySQL, PostgreSQL, SQLite, SQL Server, Oracle...

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

```
$dsn = 'mysql:host=хост;dbname=назва_бази';
```

```
$username = 'користувач';
```

```
$password = 'пароль';
```

```
try {
    $pdo = new PDO($dsn, $username, $password);
} catch (PDOException $e) {
    echo 'Помилка підключення: ' . $e->getMessage();
}
```

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

```
try {
    // ваші запити до бази даних
} catch (PDOException $e) {
    echo 'Помилка: ' . $e->getMessage();
}
```

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

Параметризовані запити дозволяють передавати дані до запиту безпосередньо,

					<i>Лабораторна робота № 12</i>		
Зм	Лист	№ докум	Підпис	Дата	Використання PDO для обміну даними з СКБД		
Розробив	Єдаменко						
Перевірив	Левицький						
Оцінка					Група 451		
Затв							

унікаючи можливості SQL-ін'єкцій

```
$stmt = $pdo->prepare('SELECT * FROM таблиця WHERE поле = :значення');
```

```
$stmt->execute([':значення' => $значення]);
```

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

```
$stmt = $pdo->prepare('INSERT INTO таблиця (поле1, поле2) VALUES (:значення1, :значення2)');
```

```
$stmt->execute([':значення1' => $знач1, ':значення2' => $знач2]);
```

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

```
// Виведення результату у вигляді асоціативного масиву
```

```
$result = $stmt->fetch(PDO::FETCH_ASSOC);
```

```
// Виведення результату у вигляді об'єкта
```

```
$result = $stmt->fetch(PDO::FETCH_OBJ);
```

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