



SQL запит:

Users:

CREATE TABLE users (

id INT AUTO\_INCREMENT PRIMARY KEY,

login VARCHAR(255) NOT NULL,

full\_name VARCHAR(255) NOT NULL,

email VARCHAR(255) NOT NULL,

password VARCHAR(255) NOT NULL

);

Books:

CREATE TABLE books (

id INT AUTO\_INCREMENT PRIMARY KEY,

title VARCHAR(255) NOT NULL,

author VARCHAR(255) NOT NULL,

published\_year INT NOT NULL

);

login.php:

<?php

session\_start();

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

require\_once 'config.php';

$dsn = "mysql:host=localhost;dbname=books\_database";

$username = "nick";

$password = "giena";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC

];

try {

$conn = new PDO($dsn, $username, $password, $options);

} catch (PDOException $e) {

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

}

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

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

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

$stmt->execute([$login]);

$user = $stmt->fetch();

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

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

header("Location: list.php");

} else {

echo "Неправильний логін або пароль";

}

}

?>

Add\_book.php:

<?php

require\_once 'config.php';

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

$dsn = "mysql:host=localhost;dbname=books";

$username = "root";

$password = "";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC

];

try {

$conn = new PDO($dsn, $username, $password, $options);

} catch (PDOException $e) {

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

}

if(isset($\_POST['title'], $\_POST['author'], $\_POST['published\_year'])){

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

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

$published\_year = $\_POST['published\_year'];

$sql = "INSERT INTO books (title, author, published\_year) VALUES (?, ?, ?)";

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

$stmt->execute([$title, $author, $published\_year]);

echo "Книга занесена";

} else {

echo "Помилка";

}

}

?>

List.php:

<?php

session\_start();

require\_once 'config.php';

if (!isset($\_SESSION['login'])) {

header("Location: login.php");

exit;

}

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

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

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

$published\_year = $\_POST['published\_year'];

$sql = "INSERT INTO books (title, author, published\_year) VALUES (?, ?, ?)";

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

$stmt->execute([$title, $author, $published\_year]);

echo "Книга успішно додана";

}

?>

Add.html:

<!DOCTYPE html>

<html lang="ru">

<head>

<title>Add a book</title>

<meta charset="UTF-8">

</head>

<body>

<h2>Додати книгу</h2>

<form action="add\_book.php" method="post">

Назва книги: <input type="text" name="title"><br>

Автор: <input type="text" name="author"><br>

Рік видання: <input type="number" name="published\_year"><br>

<input type="submit" value="Додати книгу">

</form>

</body>

</html>

Config.php:

<?php

$host = 'localhost';

$db\_name = 'books';

$username = 'root';

$password = '';

try {

$conn = new PDO("mysql:host=$host;dbname=$db\_name", $username, $password);

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

} catch(PDOException $e) {

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

}

?>

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, зо-крема 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, уникаючи 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, :зна-чення2)');

$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);