



ZSE- BAZY DANYCH

ŁĄCZENIE Z BAZĄ DANYCH W PHP - BIBLIOTEKA PDO



CELE LEKCJI:

- Poznanie biblioteki PDO w PHP.
- Nauczenie się, jak nawiązać połączenie z bazą danych przy użyciu PDO.
- Zrozumienie różnicy między proceduralnym a obiektowym podejściem do pracy z bazami danych w PHP.
- Bezpieczeństwo aplikacji przy użyciu przygotowanych zapytań (prepared statements).



WPROWADZENIE DO PDO

PDO (PHP Data Objects) to rozbudowana biblioteka PHP, która umożliwia interakcję z bazami danych w sposób bardziej nowoczesny i bezpieczny niż starsze funkcje, takie jak `mysqli_connect` (proceduralne). PDO obsługuje różne bazy danych (np. MySQL, PostgreSQL, SQLite), dzięki czemu kod aplikacji jest bardziej przenośny i łatwiejszy do utrzymania.

Zalety PDO:

- Obsługa wielu baz danych.
- Bezpieczeństwo (np. ochrona przed SQL Injection).
- Możliwość przygotowywania zapytań (Prepared Statements).

JAK POŁĄCZYĆ SIĘ Z BAZĄ DANYCH PRZY UŻYCIU PDO?

```
<?php
// Parametry połączenia z bazą danych
$host = 'localhost';
$db = 'nazwa_bazy';
$user = 'nazwa_uzytkownika';
$pass = 'haslo';

// Tworzenie DSN (Data Source Name)
$dsn = "mysql:host=$host;dbname=$db;charset=utf8mb4";

try {
    // Tworzenie obiektu PDO
    $pdo = new PDO($dsn, $user, $pass);
    // Ustawienie trybu błędów PDO na wyjątki
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Połączenie z bazą danych powiodło się!";
} catch (PDOException $e) {
    echo "Błąd połączenia z bazą danych ↓. $e->getMessage();
}
```

```
<?php
// Parametry połączenia z bazą danych
$host = 'localhost';
$db = 'nazwa_bazy';
$user = 'nazwa_uzytkownika';
$pass = 'haslo';

// Tworzenie DSN (Data Source Name)
$dsn = "mysql:host=$host;dbname=$db;charset=utf8mb4";

try {
    // Tworzenie obiektu PDO
    $pdo = new PDO($dsn, $user, $pass);
    // Ustawienie trybu błędów PDO na wyjątki
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Połączenie z bazą danych powiodło się!";
} catch (PDOException $e) {
    echo "Błąd połączenia z bazą danych: " . $e->getMessage();
}
?>
```

Wyjaśnienie:

- Tworzymy DSN (Data Source Name), który zawiera informacje o bazie danych.
- PDO tworzy nowy obiekt połączenia. Jeśli połączenie jest prawidłowe, możemy z niego korzystać. Jeśli wystąpi błąd, łapiemy go w bloku catch.

RÓŻNICE MIĘDZY PODEJŚCIEM PROCEDURALNYM A OBIEKTOWYM

PROCEDURALNE PODEJŚCIE (MYSQLI)

```
<?php
$host = 'localhost';
$db = 'nazwa_bazy';
$user = 'nazwa_uzytkownika';
$pass = 'haslo';

// Połączenie z bazą danych
$conn = mysqli_connect($host, $user, $pass, $db);

if (!$conn) {
    die("Połączenie nie powiodło się: " . mysqli_connect_error());
}
echo "Połączenie z bazą danych powiodło się!";
?>
```

Zalety:

- Prostota i łatwość w nauce.

Wady:

- Brak wsparcia dla wielu typów baz danych.
- Większe ryzyko podatności na ataki SQL Injection (chyba że użyjemy funkcji `mysqli_prepare()`).

SQL INJECTION

SQL Injection (wstrzykiwanie SQL) to technika ataku na aplikacje internetowe, która polega na wstrzyknięciu złośliwego kodu SQL do zapytania wysyłanego do bazy danych. Atak ten jest możliwy, gdy aplikacja nieprawidłowo przetwarza dane wejściowe pochodzące od użytkownika, takie jak formularze, pola wyszukiwania, czy adresy URL, i wprowadza je bez odpowiedniego oczyszczenia lub walidacji bezpośrednio do zapytania SQL.

Jak działa SQL Injection?

Atakujący modyfikuje dane wejściowe, aby manipulować zapytaniem SQL w taki sposób, że baza danych wykonuje niezamierzone polecenia.

Może to prowadzić do:

Kradzieży danych –

Atakujący może uzyskać dostęp do poufnych informacji, takich jak dane użytkowników.

Modyfikacji danych –

Zmiana lub usunięcie danych w bazie.

Nieautoryzowanego

dostępu – Zalogowanie się do konta bez posiadania odpowiednich uprawnień.

Usunięcia bazy danych –

Atakujący może wykonać zapytanie, które spowoduje usunięcie całej tabeli lub bazy danych.

OBIEKTOWE PODEJŚCIE (PDO)

Jak wspomniano wcześniej, PDO jest bardziej elastyczne i bezpieczne, dzięki obsłudze przygotowanych zapytań oraz lepszemu zarządzaniu błędami.

Bezpieczeństwo – przygotowane zapytania (Prepared Statements)

Przy użyciu PDO możemy stosować przygotowane zapytania, które chronią przed atakami SQL Injection.

```
<?php
$stmt = $pdo->prepare('SELECT * FROM users WHERE email = :email');
$stmt->execute(['email' => $email]);
$users = $stmt->fetchAll();

foreach ($users as $user) {
    echo $user['name'];
}
?>
```

Wyjaśnienie:

- **prepare()**.
przygotowuje
zapytanie z
parametrami.
- **execute()**. wykonuje
zapytanie z
podanymi danymi.

PODSUMOWANIE

- PDO oferuje elastyczność i bezpieczeństwo, obsługując różne typy baz danych.
- Proceduralne podejście z mysqli jest prostsze, ale mniej elastyczne i mniej bezpieczne.
- Zawsze warto korzystać z przygotowanych zapytań (Prepared Statements) dla bezpieczeństwa.

Zadanie dla uczniów:

- Napisz skrypt PHP łączący się z bazą danych MySQL przy użyciu PDO.
- Utwórz tabelę uczen i napisz zapytanie, które wstawia dane do tabeli z użyciem przygotowanych zapytań.



ZADANIE- PRZYKŁAD- METODA PROCEDURALNA

imie:

nazwisko:

Płeć: ☐ Kobieta ☐ Mężczyzna

miasto:

Dodałeś: Miś

```
<?php
$c=mysqli_connect("localhost", "root","", "klasa4");
if (isset($_POST["dodaj"])){
$imie=$_POST['imie'];
$nazwisko=$_POST['nazwisko'];
$miasto=$_POST['miasto'];
$plec=$_POST['plec'];
$z1=" INSERT INTO `osoby` ( `nazwisko`, `imie`, `miasto`, `plec`)
VALUES ('$nazwisko','$imie','$miasto','$plec');";
$rez=mysqli_query($c,$z1);

if ($rez) {
    echo "Dodałeś: " . $_POST['nazwisko'];
} else {
    echo "Nie udało się wstawić danych";
}
}
mysqli_close($c);
?>
```

```
<form method="post" action="wstaw2.php">
imie: <input type="text" name="imie" required><br>
nazwisko: <input type="text" name="nazwisko" required><br>
<label for="plec">Płeć:</label>
    <input type="radio" id="kobieta" name="plec" value="k">
    <label for="kobieta">Kobieta</label>

    <input type="radio" id="meczczyzna" name="plec" value="m">
    <label for="meczczyzna">Mężczyzna</label><br>
miasto: <input type="text" name="miasto" required><br>
```

```
<input type="submit" value="dodaj" name="dodaj">
<input type="reset" value="wyczysc">
</form>
```

Formularz, który wstawia rekordy:

```
<form method="post" action="wstaw2.php">
imie: <input type="text" name="imie" required><br>
nazwisko: <input type="text" name="nazwisko" required><br>
    <label for="plec">Płeć:</label>
    <input type="radio" id="kobieta" name="plec" value="k">
    <label for="kobieta">Kobieta</label>

    <input type="radio" id="meczczyzna" name="plec" value="m">
    <label for="meczczyzna">Mężczyzna</label><br>
miasto: <input type="text" name="miasto" required><br>

    <input type="submit" value="dodaj" name="dodaj">
    <input type="reset" value="wyczysc">
</form>
```

```
<?php
$c=mysqli_connect("localhost", "root","", "klasa4");
if (isset($_POST["dodaj"])){
    $imie=$_POST['imie'];
    $nazwisko=$_POST['nazwisko'];
    $miasto=$_POST['miasto'];
    $plec=$_POST['plec'];
    $z1=" INSERT INTO `osoby` ( `nazwisko`, `imie`, `miasto`, `plec`)
    VALUES ('$nazwisko','$imie','$miasto','$plec');";
    $rez=mysqli_query($c,$z1);

    if ($rez) {
        echo "Dodałeś: " . $_POST['nazwisko'];
    } else {
        echo "Nie udało się wstawić danych";
    }
}
mysqli_close($c);
?>
```

PRZYKŁAD- METODA PDO

imie:
nazwisko:
Płeć: ☐ Kobieta ☐ Mężczyzna
miasto:

Dodane: Miś

```
<?php
```

```
$host = 'localhost';  
$db = 'klasa4';  
$user = 'root';  
$pass = '';  
$pdo = new PDO("mysql:host=$host;dbname=$db;charset=utf8", $user, $pass);  
// Sprawdzenie, czy formularz został wysłany  
if (isset($_POST["dodaj"])) {  
    // Pobieranie danych z formularza  
    $imie = $_POST['imie'];  
    $nazwisko = $_POST['nazwisko'];  
    $miasto = $_POST['miasto'];  
    $plec = $_POST['plec'];  
  
    $sql = "INSERT INTO osoby (nazwisko, imie, miasto, plec) VALUES ('$nazwisko', '$imie', '$miasto', '$plec')";  
    // Wykonanie zapytania  
    $pdo->exec($sql);  
    echo "Dodane: " . $nazwisko;  
}  
// Zamknięcie połączenia (PDO automatycznie zamyka połączenie po zakończeniu skryptu)  
?>
```

```
<?php
```

```
$host = 'localhost';  
$db = 'klasa4';  
$user = 'root';  
$pass = '';  
$pdo = new  
PDO("mysql:host=$host;dbname=  
$db;charset=utf8", $user, $pass);  
// Sprawdzenie, czy formularz  
został wysłany  
if (isset($_POST["dodaj"])) {  
    // Pobieranie danych z  
    formularza  
    $imie = $_POST['imie'];  
    $nazwisko = $_POST['nazwisko'];  
    $miasto = $_POST['miasto'];  
    $plec = $_POST['plec'];  
  
    $sql = "INSERT INTO osoby  
(nazwisko, imie, miasto, plec)  
VALUES ('$nazwisko', '$imie',  
    '$miasto', '$plec')";  
    // Wykonanie zapytania  
    $pdo->exec($sql);  
    echo "Dodane: " . $nazwisko;  
    }  
// Zamknięcie połączenia (PDO  
automatycznie zamyka połączenie  
po zakończeniu skryptu)  
?>
```

PRZYKŁAD- METODA PDO - WYŚWIETLENIE DANYCH

Lista osób

ID	Imię	Nazwisko	Miasto	Płeć
1	Jan	Kowalski	Warszawa	Kobieta
2	Anna	Nowak	Kraków	Kobieta

```
<!DOCTYPE html>
<html lang="pl">
<head>
<meta charset="UTF-8">
<title>Wyświetlanie danych</title>
<style>
table {
width: 50%;
border-collapse: collapse;
}
table, th, td {
border: 1px solid black;
}
th, td {
padding: 8px;
text-align: left;
}
th {
background-color: #f2f2f2;
}
</style>
</head>
<body>
<h2>Lista osób</h2>
<table>
<tr>
<th>ID</th>
<th>Imię</th>
<th>Nazwisko</th>
<th>Miasto</th>
<th>Płeć</th>
</tr>

<?php
// Ustawienia połączenia z bazą danych
$host = 'localhost';
$db = 'klasa4';
$user = 'root';
$pass = '';

$pdo = new PDO("mysql:host=$host;dbname=$db;charset=utf8", $user, $pass);

$sql = "SELECT id_osoba, imie, nazwisko, miasto, plec FROM osoby";
$stmt = $pdo->query($sql);

// Sprawdzanie, czy są dane do wyświetlenia
if ($stmt->rowCount() > 0) {
// Pętla po wszystkich rekordach
while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
echo "<tr>";
echo "<td>" . $row['id_osoba'] . "</td>";
echo "<td>" . $row['imie'] . "</td>";
echo "<td>" . $row['nazwisko'] . "</td>";
echo "<td>" . $row['miasto'] . "</td>";
echo "<td>" . ($row['plec'] == 'm' ? 'Mężczyzna' : 'Kobieta') . "</td>";
echo "</tr>";
}
} else {
// Jeśli brak danych
echo "<tr><td colspan='5'>Brak danych do wyświetlenia</td></tr>";
}
?>
</table>
</body>
</html>
```

```
<?php
// Ustawienia połączenia z bazą danych
$host = 'localhost';
$db = 'klasa4';
$user = 'root';
$pass = '';

$pdo = new PDO("mysql:host=$host;dbname=$db;charset=utf8", $user, $pass);

$sql = "SELECT id_osoba, imie, nazwisko, miasto, plec FROM osoby";
$stmt = $pdo->query($sql);

// Sprawdzanie, czy są dane do wyświetlenia
if ($stmt->rowCount() > 0) {
// Pętla po wszystkich rekordach
while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
echo "<tr>";
echo "<td>" . $row['id_osoba'] . "</td>";
echo "<td>" . $row['imie'] . "</td>";
echo "<td>" . $row['nazwisko'] . "</td>";
echo "<td>" . $row['miasto'] . "</td>";
echo "<td>" . ($row['plec'] == 'm' ? 'Mężczyzna' : 'Kobieta') . "</td>";
echo "</tr>";
}
} else {
// Jeśli brak danych
echo "<tr><td colspan='5'>Brak danych do wyświetlenia</td></tr>";
}
?>
```

METODA PROCEDURALNA

WYŚWIETLENIE DANYCH

Lista osób

ID	Imię	Nazwisko	Miasto	Płeć
1	Jan	Kowalski	Warszawa	Kobieta
2	Anna	Nowak	Kraków	Kobieta

```
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <title>Wyświetlanie danych</title>
  <style>
    table {
      width: 50%;
      border-collapse: collapse;
    }
    table, th, td {
      border: 1px solid black;
    }
    th, td {
      padding: 8px;
      text-align: left;
    }
    th {
      background-color: #f2f2f2;
    }
  </style>
</head>
<body>
  <h2>Lista osób</h2>
  <table>
    <tr>
      <th>ID</th>
      <th>Imię</th>
      <th>Nazwisko</th>
      <th>Miasto</th>
      <th>Płeć</th>
    </tr>

    <?php
    $conn = mysqli_connect('localhost','root','','klasa4');
    if (!$conn) {
      die("Błąd połączenia: " . mysqli_connect_error());
    }
    $sql = "SELECT id_osoba, imie, nazwisko, miasto, plec FROM osoby";
    $result = mysqli_query($conn, $sql);

    if (mysqli_num_rows($result) > 0) {

      while ($row = mysqli_fetch_assoc($result)) {
        echo "<tr>";
        echo "<td>". $row['id_osoba'] . "</td>";
        echo "<td>". $row['imie'] . "</td>";
        echo "<td>". $row['nazwisko'] . "</td>";
        echo "<td>". $row['miasto'] . "</td>";
        echo "<td>". ($row['plec'] == 'm' ? 'Mężczyzna' : 'Kobieta') . "</td>";
        echo "</tr>";
      }
    } else {
      echo "<tr><td colspan='5'>Brak danych do wyświetlenia</td></tr>";
    }
    mysqli_close($conn);
  ?>
  </table>
</body>
</html>
```

```
<?php
$conn = mysqli_connect('localhost','root','','klasa4');
if (!$conn) {
  die("Błąd połączenia: " . mysqli_connect_error());
}

$sql = "SELECT id_osoba, imie, nazwisko, miasto, plec FROM osoby";
$result = mysqli_query($conn, $sql);

if (mysqli_num_rows($result) > 0) {

  while ($row = mysqli_fetch_assoc($result)) {
    echo "<tr>";
    echo "<td>" . $row['id_osoba'] . "</td>";
    echo "<td>" . $row['imie'] . "</td>";
    echo "<td>" . $row['nazwisko'] . "</td>";
    echo "<td>" . $row['miasto'] . "</td>";
    echo "<td>" . ($row['plec'] == 'm' ? 'Mężczyzna' : 'Kobieta') . "</td>";
    echo "</tr>";
  }
} else {
  echo "<tr><td colspan='5'>Brak danych do wyświetlenia</td></tr>";
}

mysqli_close($conn);
?>
```



ZSE RZESZÓW

KONIEC