

Laboratorium PHP a baza danych MySQL

Baza danych jest pewnym zbiorem informacji. MySQL jest systemem zarządzania bazą danych (SZBD), który umożliwia w łatwy sposób dostęp do informacji i ich modyfikację.

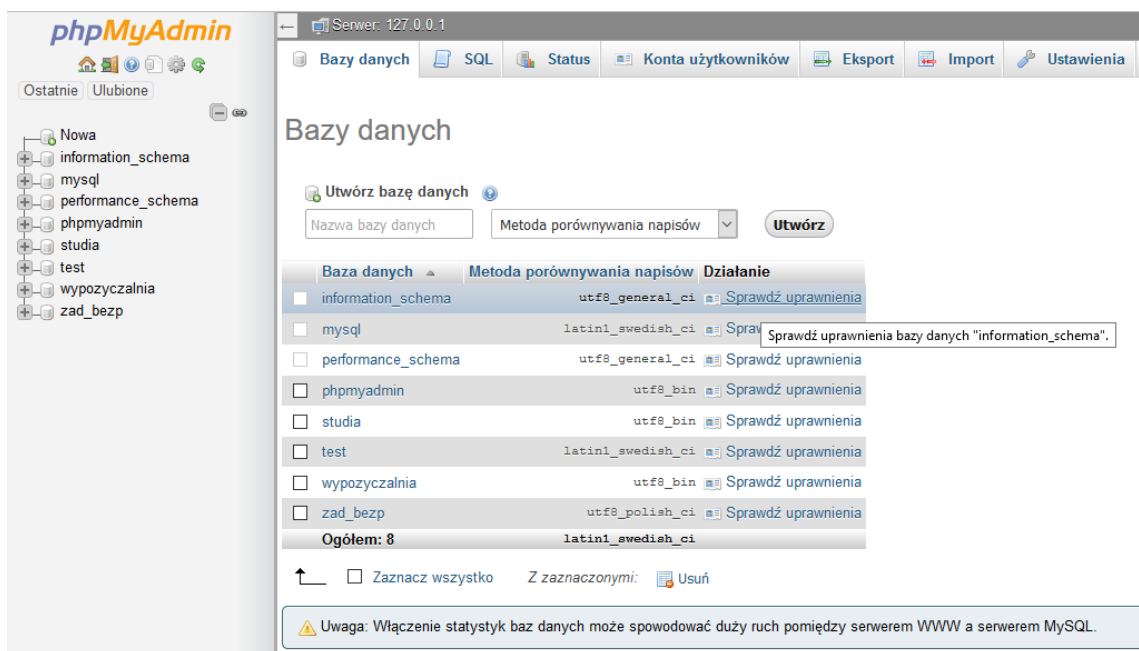
Podstawową jednostką organizacyjną w bazie danych jest tabela. Można ją sobie wyobrazić jako tablicę, w której wierszach są umieszczane rekordy, opisujące pojedyncze obiekty. Poszczególne kolumny tabeli opisują pola danego rekordu, które zawierają odpowiednie dotyczące go informacje.

Administracja bazą danych jest łatwa dzięki programowi **PHPMyAdmin** – napisanemu w PHP systemowi administracji bazą danych.

Ćwiczenie 1

Uruchom program PHPMyAdmin. Utwórz nową bazę danych o nazwie **czwicz**, a w niej tabelę **ksiazkatelefoniczna**.

Aby uruchomić program PHPMyAdmin, połącz się z adresem <http://localhost/phpmyadmin> (powinieneś mieć już uruchomiony serwer Apache i MySQL). Ukaże się program okienka administracyjnego.



Rys. 8.1. Panel administracyjny PHPMYAdmin.

Ćwiczenie 2

Korzystając z programu PHPMYAdmin, sprawdź zawartość tabeli **książkatelefoniczna**.

Ćwiczenie 3

Za pomocą PHP połącz się z bazą danych MySQL.

W celu połączenia się z bazą danych za pomocą PHP należy utworzyć skrypt. Łączyć możemy się za pomocą polecenia:

```
mysqli_connect('localhost', 'login', 'hasło', 'nazwa_bazy');
```

albo

```
$DBEngine = 'mysql';
$DBServer = 'localhost:3306';
$DBUser = 'root';
$DBPass = '';
$DBName = 'nazwa_bazy';
```

```
$pdo = new PDO("$DBEngine:host=$DBServer;dbname=$DBName", $DBUser, $DBPass);
```

UWAGA: Zalecane jest stosowanie tego drugiego sposobu.

Przykład skryptu z poleceniem mysqli_connect

```
<?php
/* Łączymy się z serwerem */
$link = mysqli_connect('localhost', 'login', 'haslo', 'nazwa_bazy');
if (!$link)
{
    echo ("Brak połączenia z serwerem MySQL. ");
    exit;
}
else
{
    Echo "jest OK";
}
?>
```

Sposób drugi:

```
<?php

$mysql_host = 'localhost'; //tutaj jest localhost ale może być adres do bazy danych
$port = '3306'; // domyślny numer portu
$username = 'root'; //w bazach rzeczywistych nazwa może być inna
$password = 'haslo'; /hasło w XAMPP jest puste
$database = 'nazwa_bazy';

try
{
    $pdo = new PDO('mysql:host=' . $mysql_host . ';dbname=' . $database . ';port=' .
$port . ";charset=utf8", $username, $password );
    echo('<p>Houston mamy polaczenie!</p>');
}
catch(PDOException $e)
{
    echo('<p>Houston mamy problem! Nie mozemy sie polaczyc!</p>');
    die(); // Nie połączyłeś się? To nie ma co robić nic więcej!
```

```
}  
?>
```

Konstruktor klasy **PDO** przyjmuje następujące parametry:

- pierwszy argument opisuje rodzaj bazy danych (np. **mysql**) następnie nazwa hosta (np. **host=localhost**); nazwa bazy danych (np. po średniku **dbname=nazwa_bazy**) i numer portu (np. domyślny **port=3306**). Do tego można (opcjonalnie) dodać inne elementy związane z ustawieniami połączenia, np. ustawienie kodowania znaków **charset=utf8**;
- nazwa użytkownika;
- twoje hasło

Poniżej sposób 2 w praktyce

http://pgmys.edu.pl/paiklasa4/lekcje/listing22_1.php

Ćwiczenie 4

Za pomocą PHP połącz się z bazą danych MySQL. Doprowadź do tego, by dane z tabeli *ksiazkatelefoniczna* zostały wyświetlone na stronie WWW

Do wykonywania zapytań służą dwie metody - `exec()` i `query()`.

Metoda `exec()` zwraca liczbę rekordów zmodyfikowanych, usuniętych lub dodanych w wyniku wykonania zapytania. Jak więc nietrudno się domyślić, wykorzystuje się ją w przypadku zapytań UPDATE, DELETE i INSERT.

Metoda `query()` zwraca obiekt PDOStatement, z którego łatwo wyciągniemy sobie rekordy, jakie otrzymaliśmy od bazy danych. Obsługuje ona również podpinanie parametrów (bind).

Przykład wywołania metody `exec()`:

```
$usuniete_rekordy = $db->exec('DELETE FROM tabela WHERE id = 5');  
echo('Usuniętych rekordów: '.$usuniete_rekordy);
```

Przykład wywołania metody query() z wyświetleniem wszystkich zwróconych przez zapytanie rekordów:

```
$statement = $db->query('SELECT imie, nazwisko FROM tabela');
foreach($statement as $wiersz)
{
    echo($wiersz['imie']." ".$wiersz['nazwisko']."<br />");
}
$statement->closeCursor();
```

Przykładowe rozwiązanie ćwiczenia 4:

```
<?php

$mysql_host = 'localhost'; //tutaj jest localhost ale może być adres do bazy danych
$port = '3306'; // domyślny numer portu
$username = 'root'; //w bazach rzeczywistych nazwa może być inna
$password = 'haslo'; //haslo w XAMPP jest puste
$databse = 'nazwa_bazy';

try
{
    $pdo = new PDO('mysql:host=' . $mysql_host . ';dbname=' . $databse . ';port=' . $port .
";charset=utf8", $username, $password );

    $pytanie = $pdo->query('SELECT *FROM `ksiazka_telefoniczna`');
    echo "Lista z bazy: <br />";
    echo '<ul>';
    foreach($pytanie as $row)
    {
        echo "<li>Imie i Nazwisko:". $row['imie']. " ". $row['nazwisko']. " telefon:
". $row['telefon']. "</li>";
    }
    $pytanie->closeCursor();
    echo '</ul>';

}
catch(PDOException $e)
{
    echo('<p>Houston mamy problem! Nie mozemy sie polaczyc!</p>');
    die(); // Nie połączyłeś się? To nie ma co robić nic więcej!
}
?>
```

Przykładowe rozwiązanie:

http://pgmys.edu.pl/paiklasa4/lekcje/listing22_2.php

Ćwiczenie 4

Napisz skrypt/skrypty wyszukujące informacje w naszej bazie książka_telefoniczna. Projekt może się składać z dwóch plików:

- szukaj.php - ma zawierać formularz umożliwiający określenie warunków wyszukiwania,
- wyniki.php - prezentujący wyniki wyszukiwania.

Przykładowe rozwiązanie:

http://pgmys.edu.pl/paiklasa4/lekcje/listing22_3a.php

```
// Przykład formularza
<!doctype html>
<html>
  <head>
    <title>Wyszukiwanie kontaktów w bazie książka_telefoniczna</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h1>Wyszukiwanie kontaktów w bazie książka_telefoniczna</h1>
    <form action="listing22_3b.php" method="post">
      Wyszukaj według:
      <select name="metoda">
        <option value="imie" />imie
        <option value="nazwisko" />nazwisko
        <option value="telefon" />telefon
      </select>
      <br /><br />
      Szukane wyrażenie:
      <input type="text" name="wyrazenie" />
      <input type="submit" name="wyszukaj" />
    </form>
  </body>
</html>
```

Przykład zapytania:

```
<!doctype html>
<html>
<head>
  <title>Wyniki wyszukiwania kontaktów</title>
  <meta charset="utf-8" />
</head>
<body>
  <h1>Wyniki wyszukiwania kontaktó w bazie ksiazka_telefoniczna</h1>
  <?php
    $metoda = $_POST['metoda'];
    $wyrazenie = $_POST['wyrazenie'];
    $wyrazenie = trim($wyrazenie);
    if (!$metoda || !$wyrazenie)
    {
      echo 'Brak parametrów wyszukiwania, wróć do poprzedniej strony i spóbuć ponownie!';
      exit;
    }
    if (!get_magic_quotes_gpc())
    {
      $metoda = addslashes($metoda);
      $wyrazenie = addslashes($wyrazenie);
    }

    $mysql_host = 'localhost'; //tutaj jest localhost ale może być adres do bazy danych
    $port = '3306'; // domyślny numer portu
    $username = 'pgmysedu_user';
    $password = 'ZAQ12wsxcde3';
    $database = 'pgmysedu_baza';
  try
  {
    $pdo = new PDO('mysql:host=' . $mysql_host . ';dbname=' . $database . ';port=' . $port .
";charset=utf8", $username, $password );
    $zapytanie = "select * from ksiazka_telefoniczna where ".$metoda. " like
'%" . $wyrazenie . "%'";
    $wynik = $pdo->query($zapytanie);
    $ile_znaleziono = $wynik->fetchAll();
    //echo count($ile_znaleziono);
    echo '<p> Liczba znalezionych kontaktów: ' . count($ile_znaleziono) . '</p>';

    $wynik = $pdo->query($zapytanie);
    echo "Lista z bazy: <br />";
    echo '<ul>';
    foreach($wynik as $row)
    {
      echo "<li>Imie i Nazwisko: " . $row['imie'] . " " . $row['nazwisko'] . " telefon:
" . $row['telefon'] . "</li>";
    }
    $wynik->closeCursor();
    echo '</ul>';
```

```

}

catch(PDOException $e)
{
    echo('<p>Houston mamy problem! Nie mozemy sie polaczyc!</p>');
    die(); // Nie połączyłeś się? To nie ma co robić nic więcej!
}

?>
</body>
</html>

```

Ćwiczenie 5

Napisz skrypt/skrypty które umożliwiają dopisywanie danych do bazy książka_telefoniczna. Projekt może się składać z dwóch plików:

- dodaj.php - ma zawierać formularz umożliwiający określenie warunków wyszukiwania,
- wpisz.php - prezentujący wyniki wyszukiwania.

Przykładowe rozwiązanie:

http://pgmys.edu.pl/paiklasa4/lekcje/listing22_4a.php

Przykładowy plik formularza:

```

<!doctype html>
<html>
<head>
    <title>Dodawanie kontaktów w bazie książka_telefoniczna</title>
    <meta charset="utf-8" />
</head>
<body>
    <h1>Wyszukiwanie kontaktów w bazie książka_telefoniczna</h1>
    <form action="listing22_4b.php" method="post">
        Wpisz do bazy:
        <TABLE>
            <TR><TD>Imie: </TD> <TD>    <input type="text" name="im" /> </TD> </TR>
            <TR><TD>Nazwisko: </TD> <TD> <input type="text" name="naz" /> </TD> </TR>
            <TR><TD>Telefon: </TD> <TD>    <input type="text" name="tel" /> </TD> </TR>
        </TABLE><br />
        <input type="submit" name="dopisz" value="Dopisz do bazy" />
    </form>
</body>
</html>

```

Przykładowy plik dopisujący dane

```
<!doctype html>
<html>
<head>
  <title>Dodawanie kontaktów kontaktów</title>
  <meta charset="utf-8" />
</head>
<body>
  <h1>Dodawanie kontaktów w bazie ksiazka_telefoniczna</h1>
  <?php
    $imie = trim($_POST['im']);
    $nazwisko = trim($_POST['naz']);
    $telefon = trim($_POST['tel']);

    if (!$imie || !$nazwisko || !$telefon)
    {
      echo "rak wszystkich danych, wróć do poprzedniej strony i spóbuć ponownie!";
      exit;
    }

    $imie = addslashes($imie);
    $nazwisko = addslashes($nazwisko);
    $telefon = addslashes($telefon);

    $mysql_host = 'localhost'; //tutaj jest localhost ale może być adres do bazy danych
    $port = '3306'; // domyślny numer portu
    $username = 'pgmysedu_user';
    $password = 'ZAQ12wsxcde3';
    $database = 'pgmysedu_baza';

  try
  {
    $pdo = new PDO('mysql:host=' . $mysql_host . ';dbname=' . $database . ';port=' . $port .
";charset=utf8", $username, $password );
    $zapytanie = "insert into ksiazka_telefoniczna values ('', ''.$imie.'', ''.$nazwisko.'',
''.$telefon.'')";
    $wynik = $pdo->query($zapytanie);
    if ($wynik) echo "Dodano rekord do bazy";
  }
  catch(PDOException $e)
  {
    echo('<p>Houston mamy problem! Nie mozemy sie polaczyc!</p>');
    die(); // Nie połączyłeś się? To nie ma co robić nic więcej!
  }
  ?>
</body>
</html>
```

Ćwiczenie 6

Napisz skrypt/skrypty które umożliwiają usunięcie wskazanego przez użytkownika rekordu. Projekt może się składać z dwóch plików:

- wybierz.php – wypisuje wszystkie dane z bazy oraz zawiera formularz w którym można wpisać numer usuwanego rekordu.
- usun.php – skrypt usuwający rekord z bazy.

Zadania:

1. Korzystając z programu PHPMyAdmin, wpisz dane do nowo utworzonej tabeli biblioteka (tabela powinna zawierać przynajmniej cztery różne pola danych).
2. Napisz program / skrypt, który umożliwia dodawanie, usuwanie i również kasowanie nowych rekordów w tabeli biblioteka.
3. Zadanie egzaminacyjne

ZADANIA

1. W MySQL utwórz bazę danych *Moja_szkola* lub wykorzystaj bazę danych *Moja_szkola* tworzoną w części II podręcznika. Baza danych powinna zawierać tabelę *Uczen* z danymi uczniów oraz tabelę *Rejestracja* zawierającą dwa pola: *login* i *hasło*. Zostanie ona wykorzystana do rejestracji loginów i haseł użytkowników strony.
2. Utwórz stronę internetową, która za pomocą skryptów PHP będzie komunikowała się z tą bazą danych. Strona główna ma zawierać formularz logowania oraz formularz rejestracji nowych uczniów.
3. Utwórz skrypt, który:
 - dane z formularza rejestracyjnego zapisze w tabeli *Uczen*,
 - login i hasło zapisze w tabeli *Rejestracja*,
 - po zalogowaniu przekieruje użytkownika do strony umożliwiającej mu zamieszczanie komentarza,
 - wyświetli na tej stronie dotychczasowe komentarze użytkowników,
 - wyświetli nowy komentarz poniżej wcześniejszych komentarzy,
 - wyświetli na stronie przycisk wylogowania, a po jego kliknięciu wyloguje użytkownika ze strony.