

КАТЕДРА: КОМПЮТЪРНИ СИСТЕМИ И ТЕХНОЛОГИИ
ДИСЦИПЛИНА: ИНФОРМАЦИОННИ СИСТЕМИ

ЛАБОРАТОРНО УПРАЖНЕНИЕ № 5

ТЕМА: Извличане на данни от MySQL база данни чрез PHP скрипт

ЦЕЛ:

Целта на упражнението е студентите да получат практически знания и умения за извличане на информация от MySQL база данни чрез PHP скрипт. След упражнението студентите би следвало и да могат да реализират елементарни заявки към базата данни чрез PHP.

I. ТЕОРЕТИЧНА ЧАСТ

Извличането на данни от MySQL база данни става след като се изпращат определени SQL заявки към базата. За целта трябва да се използва вече познатата ни функция **mysqli_query()**, която бе разгледана в предното упражнение.

За да може да извличаме информация от базата данни ще трябва да си припомним SQL оператора SELECT. Той извличаше определени записи от таблица в зависимост от определени условия. Синтаксисът му е следния:

```
SELECT [DISTINCT] [поле1 AS псевдоним, .... , полеN AS псевдонимN]  
FROM Име_на_таблица1 [, ... , Име_на_таблицаN]  
[WHERE условие]  
[GROUP BY списък с полета]  
[ORDER BY списък с полета]  
[LIMIT количество]
```

Как се работеше с този оператор е изучавано по дисциплината Базии Данни и не е обект на настоящото упражнение.

Функцията **mysqli_query()**, приемаше 2 параметъра. Първият е ресурса сочещ към базата от данни, който се получава от функцията **mysqli_connect()**, а вторият е низа със SQL заявката. При успешно изпълнение на **mysqli_query()** се връща ресурсен указател към всички записи които отговарят на SQL оператора който сме подали като втори параметър. Резултатът може да бъде нула, един или много записи. За да можем да покажем всички записи трябва да използваме функцията **mysqli_fetch_array()**:

Масив = **mysqli_fetch_array** (връзкаКъмДанни);

Функцията приема само един параметър – ресурса, получен от функцията **mysqli_query()** и връща масив, чиито ключове са полетата на запис от базата. Така, вече информацията може да се извлече, както се достъпват елементите на масив в PHP. Ако използваме функцията **mysqli_fetch_array()** в цикъл то ще можем да покажем всички извлечени записи.

```
While (Масив = mysqli_fetch_array ( връзкаКъмДанни ) )  
{ echo Масив["поле1"].“ “;  
.....  
echo Масив["полеN"].“ “;}
```

В горния примерен код извикваме функцията `mysqli_fetch_array()` от цикъла `“while”`. Функцията връща единичен запис от базата данни и го записва в масива. Когато бъде получен и последния запис `mysqli_fetch_array()` връща `„false“` и цикълът прекратява повторенията си. В тялото на цикъла използваме няколко оператора `„echo“` за да покажем стойностите на полетата. Те се съхраняват в масива и са достъпни като отделни елементи с ключове отговарящи на полетата от таблицата в базата данни.

II. ПРАКТИЧЕСКА ЧАСТ

В практическата част е показан пример чрез който се осъществява достъп до определена MySQL база данни и извлича информация от нея. За целта е решена следните задача:

ЗАДАЧА: Да се осъществи достъп от PHP скрипт към MySQL база от данни „University“, която бе създадена в практическата част на лабораторно упражнение номер 4.

- 1) Да се извлекат всички записи от таблица *Students*.
- 2) От таблица *Students* да се извлекат всички студенти които имат успех по голям от добър 4.00.
- 3) Да се създаде html форма с падащо меню, която предоставя възможност на потребителя да избере факултетния номер на студент. След което с натискане на бутон OK да се покаже името на студента и какъв му е средния успех.

Табл.1. Students

<i>fnum</i>	<i>name</i>	<i>age</i>	<i>grade</i>
2344	<i>Dian</i>	21	3.56
2234567	<i>Ivan</i>	21	4.59

!ВАЖНО: За да можете да тествате упражнението ви е необходим следния софтуер. Web server, Php интерпретатор, MYSQL база данни. Всеки един от тези софтуери ги има в пакетите WAMP или XAMP. Настоящото упражнение е тествано върху WAMP Version 3.2.0.

1. Решение на задача 1).

КОД:

```
<?php
```

```
//1. Осъществяване връзка към база данни University
```

```
$con = mysqli_connect('localhost', 'root', '', 'university') or die(mysqli_connect_error());
```

```
//2. Задаване на необходимата SQL заявка.
```

```
$querySt="SELECT * FROM Students";
```

```
//3.Изпращане на необходимата SQL заявка към сървъра
```

```
$dbStRecords=mysqli_query($con,$querySt) or  
die('Неуспешно запитване. Получи следната грешка'.mysqli_error($con));
```

//4. Чрез цикъл прочитаме всички върнати записи от заявката

```
while($StRecords=mysqli_fetch_array($dbStRecords))  
{  
    echo $StRecords["fnom"]." | ";  
    echo $StRecords['name']. " | ";  
    echo $StRecords['age']. " | ";  
    echo $StRecords['grade']. "<br>";  
}
```

//5. Затваряне на връзката с базата данни

```
mysqli_close($con);
```

?>

ПОЯСНЕНИЕ

На първия етап от кода се осъществява достъп до база данни “University” на MySQL сървър, инсталиран на локалната машина като се използва потребител root без парола. Ако поради някаква причина не бъде осъществена връзката се връща съобщението „Неуспешна връзка“ и се показва точната причина за грешката.

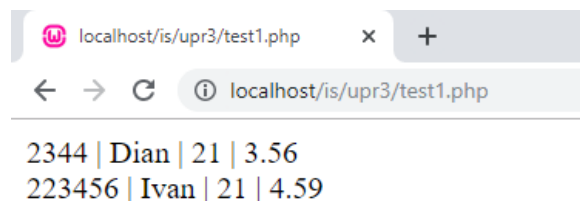
Във втория етап се подготвя SQL заявката за изпращане към сървъра. Тя се записва в променливата `$querySt`. За извличането на всичките записи от таблицата се използва SQL оператор **SELECT** със звездичка след което се посочва от коя таблица се извлича информацията.

На 3-тия етап се изпраща подготвената заявка чрез функцията `mysqli_query()`, като заявката се подава чрез променливата `$querySt`.

В четвъртия етап използваме функцията `mysqli_fetch_array()` която извлича един запис от ресурса `$dbStRecords` върнат от функцията `mysqli_query()`. Този запис се записва в масива `$StRecords`. От който чрез echo команди отпечатваме извлечената информация. Чрез цикъл While се извличат всички записи с функцията `mysqli_fetch_array()` от ресурса `$dbStRecords`. Когато няма повече записи функцията ще върне стойност „false“ и цикъла ще приключи своята работа.

На последния етап се затваря осъществената връзка с базата данни.

Резултат от изпълнението на задачата е показан на фиг.1.



Фиг.1. Резултат от изпълнението на зад.1)

2. Решение на задача 2).

Решението на задачата е абсолютно същото като предходното като единствено трябва да се промени SQL заявката във вторият етап от кода на следната:

```
$querySt="SELECT * FROM Students Where grade > 4.00";
```

По този начин чрез „Where“ ограничаваме да се извлекат само тези които студенти имат по голям успех от Добър 4.00. Така за резултат тази задача би трябвало да съдържа само 2-рият ред от фиг.1.

3. Решение на задача 3).

КОД

forma.php

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title></title>
  </head>
  <body>
    <?php
      $con = mysqli_connect('localhost', 'root', '', 'university')
      or die(mysqli_connect_error());
      $querySt= "SELECT DISTINCT fnom FROM Students";
      $dbStRecords=mysqli_query($con,$querySt) or
      die('Неуспешно запитване. Получи следната грешка'.mysqli_error($con));
    ?>
    <form method="post" action="test1.php">
      Моля посочете факултетния номер на студента<br>
      <select name="nomer">
        <?php
          While ($StRecords=mysqli_fetch_array($dbStRecords))
          {
            ?>
            <option value="<?php echo $StRecords['fnom']; ?>"><?php echo
$StRecords['fnom'];?></option>
            <?php
              }
            ?>
          </select>
          <input type="submit" name="submit" value="OK">
        </form>
      </body>
    </html>
```

test1.php

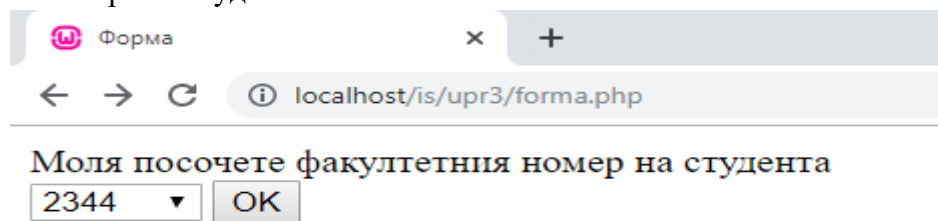
```
<?php
$nom=$_POST['nomer'];
$querySt="SELECT name, grade FROM studenti Where fnom=$nom";
$dbStRecords=mysqli_query($con,$querySt) or
die ('Неуспешно запитване. Получи се следната грешка'.mysqli_error($con));
$StRecords=mysqli_fetch_array($dbStRecords)
echo $StRecords['name']." | ";
echo $StRecords['grade']."<br>";
```

?>

ПОЯСНЕНИЕ

Чрез този код се предоставя възможност на потребителя да избере факултетен номер на студент. И с натискането на бутон ОК в нова страница се визуализира името на този студент за който отговаря факултетния номер и неговия успех.

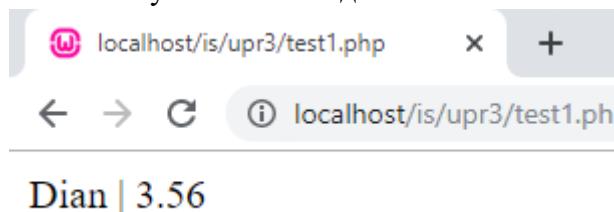
Във файла forma.php се намират html таговете които създават страницата показва на фиг.2. В него е вграден и php скрипт, чрез който се пълни падащия списък с всички факултетни номера на студентите.



Фиг.2. Визуализацията потребителската форма.

Във втория файл се намира php скрипт който обработва получената информация от първия файл. Подава се заявка към сървъра и се отпечатва съдържанието на получената заявка. Обърнете внимание че тук не е необходим цикъл за функцията `mysqli_fetch_array()` понеже резултата от получената заявка няма как да бъде повече от един ред.

Резултата след натискането на бутон ОК е следния:



Фиг.3. Визуализация на формата

III. Задача за самостоятелна работа.

1. Осъществете връзка до базата данни “Firma”, която би трябвало да сте създали в задачите за самостоятелна работа от лабораторно упражнение № 4.
2. Създайте PHP скрипт който да извлече съдържанието на таблица “Products”.
3. Създайте PHP скрипт който да изкара само тези продукти които имат цена по голяма от 5лв. (Цената може да я поставите така че да е някаква междинна цена между продуктите за да не изкара всички продукти).
4. Създайте потребителска форма със падащ списък който да предоставя възможност на потребителя да избере някой от продуктите в таблицата и при натискане на бутон „ОК“ да се изведе името на продукта описанието и неговата цена.