Лабораторная работа №4

Получение данных из базы данных MySQL.

В предыдущем уроке мы сделали тестовый код. Напомню его существенную часть, выполняющую запрос к базе:

<?php

$query = 'SELECT \* FROM users';

$result = mysqli\_query($link, $query) or die(mysqli\_error($link));

var\_dump($result);

?>

Как вы видите, после того, как произойдет запрос к базе, в переменной $result будет лежать результат этого действия. Однако лежит он не в той форме, которая нам нужна в PHP, а в той форме, в которой его прислала нам база.

Для того, чтобы получить результат в привычной нам форме, необходимо воспользоваться функцией mysqli\_fetch\_assoc, извлекающей из результата одну строку.

Давайте попробуем:

<?php

$row = mysqli\_fetch\_assoc($result);

var\_dump($row);

?>

В результате var\_dump выведет массив с первым работником:

<?php

['id' => 1, 'name' => 'user1', 'age' => 23,   
 'salary' => 400]

?>

При этом из переменной $result первый работник исчезнет, и следующий вызов mysqli\_fetch\_assoc получит уже следующего работника.

И так можно можно вызывать нашу функцию до тех пор, пока работники не закончатся. Как только это произойдет, то следующий вызов функции вернет false.

Давайте попробуем:

<?php

$row1 = mysqli\_fetch\_assoc($result);

var\_dump($row1); // работник номер 1

$row2 = mysqli\_fetch\_assoc($result);

var\_dump($row2); // работник номер 2

$row3 = mysqli\_fetch\_assoc($result);

var\_dump($row3); // работник номер 3

$row4 = mysqli\_fetch\_assoc($result);

var\_dump($row4); // работник номер 4

$row5 = mysqli\_fetch\_assoc($result);

var\_dump($row5); // работник номер 5

$row6 = mysqli\_fetch\_assoc($result);

var\_dump($row6); // работник номер 6

$row7 = mysqli\_fetch\_assoc($result);

var\_dump($row7); // выведет NULL - работники кончились

?>

Получение в виде массива

При считывании по рядам можно не выводить каждого работника, а записывать их в какой-нибудь массив:

<?php

$row1 = mysqli\_fetch\_assoc($result);

$data[] = $row1;

$row2 = mysqli\_fetch\_assoc($result);

$data[] = $row2;

$row3 = mysqli\_fetch\_assoc($result);

$data[] = $row3;

$row4 = mysqli\_fetch\_assoc($result);

$data[] = $row4;

$row5 = mysqli\_fetch\_assoc($result);

$data[] = $row5;

$row6 = mysqli\_fetch\_assoc($result);

$data[] = $row6;

?>

В результате в переменной $data получится следующий двухмерный массив:

<?php

[

['id' => 1, 'name' => 'user1', 'age' => 23,   
 'salary' => 400],

['id' => 2, 'name' => 'user2', 'age' => 25,   
 'salary' => 500],

['id' => 3, 'name' => 'user3', 'age' => 23,   
 'salary' => 500],

['id' => 4, 'name' => 'user4', 'age' => 30,   
 'salary' => 900],

['id' => 5, 'name' => 'user5', 'age' => 27,   
 'salary' => 500],

['id' => 6, 'name' => 'user6', 'age' => 28,   
 'salary' => 900],

]

?>

Формируем массив в цикле

Конечно же, в ручную перебирать всех работников не очень удобно. Пусть лучше за нас это сделает цикл:

<?php

for ($data = []; $row = mysqli\_fetch\_assoc($result); $data[] = $row);

var\_dump($data); // здесь будет массив с результатом

?>

Давайте разберемся, как работает этот цикл.

В каждой итерации цикла функция mysqli\_fetch\_assoc последовательно считывает каждую строку результата, записывая его в массив $data.

Как только в $result закончатся строки, то mysqli\_fetch\_assoc вернет NULL и цикл закончит свою работу. А полученный результат будет лежать в двухмерном массиве $data.

**Задание №1**

С помощью описанного цикла получите, и выведите через var\_dump на экран массив всех пользователей.

**Задание №2**

Из полученного результата получите первого работника. Через echo выведите на экран его имя.

**Задание №3**

Из полученного результата получите второго работника. Через echo выведите на экран его имя и возраст.

**Задание №4**

Из полученного результата получите третьего работника. Через echo выведите на экран его имя, возраст и зарплату.