

# Работа с базами данных SQL в PHP (beginner)

Создать тестовую таблицу в базе данных, заполненная некоторыми данными. Назовите ее **"workers"**.

В ней создайте 4 столбца (столбцы в базе данных называются **поля**):

- **id** – тип integer, не забудьте поставить ему галочку AUTO\_INCREMENT (чтобы в этом столбце номера проставлялись автоматически).
- **name** (англ. имя) – тип varchar, размером в 256 знаков.
- **age** (англ. возраст) - тип integer.
- **salary** (англ. зарплата) - тип integer.

Ее заполните тестовыми данными, как показано в таблице ниже (этот шаг обязателен, так как дальше все задачи будут по этой таблице):

id	name	age	salary
1	Дима	23	400
2	Петя	25	500
3	Вася	23	500
4	Коля	30	1000
5	Иван	27	500
6	Кирилл	28	1000

Итак, есть таблица с работниками фирмы, в которой указаны их имена, возрасты и зарплаты (в \$).

## Тестируем работоспособность

Прежде чем начать работу, следует протестировать работоспособность - попробуем сделать какой-нибудь запрос к нашей базе.

Просто скопируйте этот код и запустите его у себя:

```
<?php
<?php
//Устанавливаем доступы к базе данных:
$host = 'localhost';
$user = '';
$password = '';
$db_name = '';
//Соединяемся с базой данных используя наши доступы:
$db = new mysqli($host, $user, $password, $db_name);
//Устанавливаем кодировку (не обязательно, но поможет избежать проблем):
//mysqli_query($db, "SET NAMES 'utf8'");
//Формируем тестовый запрос:
$query = "SELECT * FROM workers WHERE id > 0";
//Делаем запрос к БД, результат запроса пишем в $result:
$result = mysqli_query($db, $query) or die(mysqli_error($db));
//Проверяем что же нам отдала база данных, если null – то какие-то проблемы:
var_dump($result);
$db->close();
?>
```

Если `var_dump($result)` вернет `resource`, то все работает, если же `null` — то возникли какие-то проблемы. в новых версиях PHP в `$result` будет лежать объект с данными (всегда будет непустой).

## Как достать результат

После того, как сделан 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.

Достать результат в нормальном виде (в массиве) можно с помощью следующего кода:

```
<?php
```

```
//Делаем запрос к БД, результат запроса пишем в $res:  
//Преобразуем то, что отдала нам база в нормальный массив PHP $data:  
if($res = $db->query("SELECT name, age FROM workers")){  
    while($row = $res->fetch_assoc()){  
        echo "Name: {$row['name']}; Возраст: {$row['age']}<br>";  
    }  
}  
$db->close();  
?>
```

Как работает последняя строка?

Функция `fetch_assoc()` считывает последовательно каждую строку результата, который прислала нам база.

В цикле `while` считываем построчно результат из базы.

Когда цикл дойдет до последней строки - `fetch_assoc()` вернет `false` и цикл `while` закончит свою работу.

А результат из БД будет лежать в нормальном виде в массиве `$res`.

## Примеры решения задач

### Задача 1

**Задача.** Выбрать работника с `id=10`.

**Решение:**

```
<?php
```

```
$query = "SELECT * FROM workers WHERE id=10";
```

```
?>
```

### Задача 2

**Задача.** Выбрать работников с зарплатой 500\$.

**Решение:**

```
<?php
```

```
$query = "SELECT * FROM workers WHERE salary=500";
```

```
?>
```

### Задача 3

**Задача.** Выбрать работников с зарплатой 500\$ и id больше 3.

**Решение:**

```
<?php
```

```
$query = "SELECT * FROM workers WHERE salary=500 AND id>3";  
?>
```

## Задача 4

**Задача.** Добавьте нового работника **Джона, 20 лет, зарплата 700\$.**

**Решение:**

Воспользуемся первым синтаксисом:

```
<?php  
$query = "INSERT INTO workers SET name='Джон', age=20, salary=700";  
?>
```

Воспользуемся вторым синтаксисом:

```
<?php  
$query = "INSERT INTO workers (name, age, salary) VALUES ('Джон', 20, 700)";  
?>
```

## Задача 5

**Задача.** Добавьте одним запросом трех новых работников: **Катю, 20 лет, зарплата 500\$, Юлю, 25 лет, зарплата 600\$, Женю, 30 лет, зарплата 900\$.**

**Решение:** запрос должен выглядеть так:

```
INSERT INTO workers (name, age, salary)  
VALUES ('Катя', 20, 500), ('Юля', 25, 600), ('Женя', 30, 900)
```

## Задача 6

**Задача.** Удалите работника **Джона.**

**Решение:**

```
<?php  
$query = "DELETE FROM workers WHERE name='Джон'";  
?>
```

## Задача 7

**Задача.** Поставьте Диме **зарплату в 1000\$.**

**Решение:**

```
<?php  
$query = "UPDATE workers SET salary=1000 WHERE name='Дима'";  
?>
```

## Задача 8

**Задача.** Поставьте Диме **зарплату в 1000\$ и возраст 20 лет.**

**Решение:**

```
<?php  
$query = "UPDATE workers SET salary=1000, age=20 WHERE name='Дима'";  
?>
```

## Задачи для решения

### На SELECT

1. Выбрать работника с **id = 3**.
2. Выбрать работников с зарплатой **1000\$**.
3. Выбрать работников в возрасте **23** года.
4. Выбрать работников с зарплатой более **400\$**.
5. Выбрать работников с зарплатой равной или большей **500\$**.
6. Выбрать работников с зарплатой **НЕ** равной **500\$**.
7. Выбрать работников с зарплатой равной или меньшей **900\$**.
8. Узнайте зарплату и возраст **Васи**.

### На OR и AND

9. Выбрать работников в возрасте от **25** (не включительно) до **28** лет (включительно).
10. Выбрать работника **Петю**.
11. Выбрать работников **Петю и Васю**.
12. Выбрать всех, кроме работника **Петя**.
13. Выбрать всех работников в возрасте **27** лет или с зарплатой **1000\$**.
14. Выбрать всех работников в возрасте от **23** лет (включительно) до **27** лет (не включительно) или с зарплатой **1000\$**.
15. Выбрать всех работников в возрасте от **23** лет до **27** лет или с зарплатой от **400\$** до **1000\$**.
16. Выбрать всех работников в возрасте **27** лет или с зарплатой не равной **400\$**.

### На INSERT

17. Добавьте нового работника **Никиту**, **26** лет, зарплата **300\$**.  
Воспользуйтесь **первым** синтаксисом.
18. Добавьте нового работника **Светлану** с зарплатой **1200\$**.  
Воспользуйтесь **вторым** синтаксисом.
19. Добавьте двух новых работников одним запросом: **Ярослава** с зарплатой **1200\$** и возрастом **30**, **Петра** с зарплатой **1000\$** и возрастом **31**.

### На DELETE

Удалите работника с **id=7**.

Удалите **Колю**.

Удалите всех работников, у которых **возраст 23 года**.

**Верните таблицу workers в исходное состояние.**

На UPDATE

20. Поставьте **Васе** зарплату в **200\$**.

21. Работнику с **id=4** поставьте возраст **35** лет.

22. Всем, у кого **зарплата 500\$** сделайте ее **700\$**.

23. Работникам с **id больше 2 и меньше 5 включительно** поставьте возраст **23**.

24. Поменяйте Васю на Женю и прибавьте ему зарплату до **900\$**.