

# Команды ORDER BY, LIMIT, COUNT, LIKE в SQL

## ORDER BY - сортировка

С помощью команды **ORDER BY** можно сортировать строки результата.

Выберем из нашей таблицы **workers** всех работников и отсортируем их по возрасту:

```
<?php
```

```
// В $data строки будут отсортированы по возрасту от меньшего к большему:  
$query = "SELECT * FROM workers WHERE id>0 ORDER BY age";  
?>
```

Если мы хотим **обратный порядок сортировки**, то следует написать:

```
<?php
```

```
// В $data строки будут отсортированы по возрасту от большего к меньшего:  
$query = "SELECT * FROM workers WHERE id>0 ORDER BY age DESC";  
?>
```

## LIMIT — ограничение количества

С помощью команды **LIMIT** мы можем ограничить количество строк в результате.

В следующем примере ограничим количество строк до двух:

```
<?php
```

```
// В $data будет только две первых строки:  
$query = "SELECT * FROM workers WHERE id>0 LIMIT 2";  
?>
```

Однако, это еще не все!

С помощью **LIMIT** можно выбрать несколько строк из середины результата!

В примере ниже мы выберем со второй строки, 5 штук:

```
<?php
```

```
// В $data будут строки со второй, пять штук:  
$query = "SELECT * FROM workers WHERE id>0 LIMIT 2,5";  
?>
```

**LIMIT** и **ORDER BY** вместе:

```
<?php
```

```
// В $data будут строки со второй, 5 штук, отсортированные по убыванию id  
$query = "SELECT * FROM workers WHERE id>0 ORDER BY id DESC LIMIT 2,5";
```

?>

## Команда COUNT — считаем количество

С помощью команды **COUNT** можно подсчитать **количество строк** в выборке.

Синтаксис:

<?php

```
// В $result будет лежать количество строк:  
$query = "SELECT COUNT(*) as count FROM workers WHERE id>0";  
$result = mysqli_query($link, $query) or die( mysqli_error($link) );  
$count = mysqli_fetch_[$result];  
  
// В $count будет лежать массив 'count'=>кол-во:  
var_dump($count);
```

?>

Обратите внимание на распространенную ошибку с **COUNT**: конструкцию **COUNT(\*)** следует писать слитно, без всяких пробелов, иначе не будет работать.

## Команда LIKE - реализуем поиск

С помощью команды **LIKE** (англ. *подобный*) можно реализовать поиск. Посмотрите пример использования с комментарием:

<?php

```
// ВЫБРАТЬ все ИЗ таблицы ГДЕ имя ПОДОБНО любой_строке_заканчивающейся_на_я  
$query = "SELECT * FROM workers WHERE name LIKE '%я"';  
  
/*  
Результатом будет Петя, Вася и Коля  
(если они еще уцелели после урока с DELETE).  
*/
```

?>

Подробнее о возможностях этой команды смотрите здесь: [LIKE](#).

## Кавычки ``

Обратите внимание на такой нюанс: следующий запрос работать не будет, так как имя таблицы **from** совпадает с командой **FROM**:

<?php

```
$query = "SELECT * FROM from";  
?>
```

В таких случаях проблемные слова следует брать в косые кавычки, вот так:

```
<?php  
    $query = "SELECT * FROM `from`";  
?>
```

# Задачи на базы данных SQL в PHP

## Таблица для задач

Все задачи будут по данной таблице **workers** (если не сказано иное):

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

## На LIMIT

Для решения задач данного блока вам понадобятся следующие SQL команды: [LIMIT](#).

1. Из таблицы **workers** достаньте **первые 6 записей**.
2. Из таблицы **workers** достаньте записи **со второй, 3 штуки**.

## На ORDER BY

Для решения задач данного блока вам понадобятся следующие SQL команды: [ORDER BY](#), [LIMIT](#).

3. Из таблицы **workers** достаньте всех работников и **отсортируйте их по возрастанию** зарплаты.
4. Из таблицы **workers** достаньте всех работников и **отсортируйте их по убыванию** зарплаты.
5. Из таблицы **workers** достаньте работников **со второго шестого** и **отсортируйте их по возрастанию** возраста.

## На COUNT

Для решения задач данного блока вам понадобятся следующие функции SQL: [COUNT](#).

6. В таблице **workers** подсчитайте всех работников.
7. В таблице **workers** подсчитайте всех работников с зарплатой 300\$.

## На LIKE

Для решения задач данного блока вам понадобятся следующие SQL команды: [LIKE](#).

Создайте таблицу **pages**:

| <b>id</b> | <b>author (автор)</b> | <b>article (статья)</b>                                   |
|-----------|-----------------------|---|
| 1         | Петров                | В своей статье рассказывает о машинах.                    |
| 2         | Иванов                | Написал статью об инфляции.                               |
| 3         | Сидоров               | Придумал новый химический элемент.                        |
| 4         | Осокина               | Также писала о машинах.                                   |
| 5         | Ветров                | Написал статью о том, как разрабатывать элементы дизайна. |

8. В таблице **pages** найти строки, в которых **фамилия** автора заканчивается на "ов".
9. В таблице **pages** найти строки, в которых есть слово "элемент" (искать только по колонке **article**).
10. В таблице **workers** найти строки, в которых возраст работника **начинается с числа 3**, а далее идет **только одна цифра**.
11. В таблице **workers** найти строки, в которых **имя** работника заканчивается на "я".