

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**

С.В. Одиночкина

**Web-программирование: PHP
практикум**



Санкт-Петербург

УДК 004.655, 004.657, 004.62

С.В.Одиночкина

Web-программирование PHP - СПб: НИУ ИТМО, 2012. – 79 с.

В пособии излагаются методические рекомендации к выполнению лабораторных работ по дисциплине «Web-программирование PHP-технологии».

Предназначено для студентов, обучающихся по всем профилям подготовки бакалавров направления: 210700 Инфокоммуникационные технологии и системы связи.

Рекомендовано к печати Ученым советом факультета Инфокоммуникационных технологий, протокол №4 от 13 декабря 2011г.



В 2009 году Университет стал победителем многоэтапного конкурса, в результате которого определены 12 ведущих университетов России, которым присвоена категория «Национальный исследовательский университет». Министерством образования и науки Российской Федерации была утверждена программа его развития на 2009–2018 годы. В 2011 году Университет получил наименование «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

© Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики, 2012

© С.В.Одиночкина, 2012.

Лабораторная работа №7: Страница статистики inform.php

В ходе выполнения лабораторной работы будет организована работа и выведены на страницу статистики следующие данные web-сайта:

1. Сколько всего было сделано записей в блоге;
2. Сколько комментариев было добавлено;
3. Сколько записей было сделано за последний месяц;
4. Сколько комментариев было оставлено за последний месяц;
5. Какая заметка была сделана последней;
6. Какую заметку больше всего комментировали.

Упражнение 1: Общее количество заметок и общее количество комментариев

1. Установите подключение к серверу и выберите базу данных.
2. С помощью SQL-запроса необходимо вычислить общее количество заметок в блоге. Для этого используется SQL-функция COUNT().

Данная функция возвращает количество строк, которые соответствуют определенным критериям. Синтаксис функции:

SELECT COUNT (fieldName) FROM tblName

Данная функция является **агрегатной**, т.е. позволяет выполнять различные действия сразу над многими записями.

Вариант реализации кода

//Вычисление количества заметок

```
$query_allnotes = "SELECT COUNT(id) AS allnotes FROM notes";
```

```
$allnotes = mysqli_query ($link, $query_allnotes) or die (mysqli_error());
```

//mysqli_error()возвращает строку ошибки последней операции с MySQL

```
$row_allnotes = mysqli_fetch_assoc ($allnotes);
```

```
$allnotes_num = $row_allnotes['allnotes'];
```

```
mysqli_free_result ($allnotes);
```

//mysqli_free_result() освобождает память от результата запроса

3. Аналогичным образом реализуйте подсчет общего количества комментариев.

Упражнение 2: Подсчет количества заметок и комментариев за последний месяц

В ходе выполнения этого упражнения необходимо реализовать следующий алгоритм работы:

- Вычислить начальную и конечную даты текущего месяца;
- Подставить результаты этих вычислений в условие фильтрации SQL-запроса.

1. Работа с датой

//Функция getdate() возвращает массив, содержащий информацию о различных составляющих текущей даты (чтобы дальше работать с ними "по частям"). В массив помещаются: секунды, минуты, часы, порядковый номер дня, порядковый номер месяца, порядковый номер года, название дня недели, название месяца, количество секунд с начала эпохи Unix.

```
$date_array = getdate();
```

//Вычисление начальной даты текущего месяца

//Функция mktime() возвращает объединенное значение времени. Аргументы: кол-во часов, минут, секунд, № месяца, число, год.

```
$begin_date = date ("Y-m-d", mktime(0,0,0, $date_array['mon'],1,  
$date_array['year']));
```

//Т.к. время в данном случае не нужно - поставлены нули.

//Возвращенное функцией mktime() значение приведено к воспринимаемому MySQL параметру даты "Y-m-d".

//Вычисление конечной даты текущего месяца

```
$end_date = date ("Y-m-d", mktime(0,0,0, $date_array['mon'] + 1,0,  
$date_array['year']));
```

//Здесь все аналогично, кроме того, что мы вводим число месяца, равное нулю; на основании этого функция mktime(), встретив дату с нулевым числом, вернет последнее число предыдущего месяца (28, 29, 30 или 31).

2. Запрос на получение заметок за последний месяц

```
$query_lmnotes = "SELECT COUNT(id) AS lmnotes FROM notes  
                WHERE created>='$begin_date' AND created<='$end_date';"  
$lmnotes = mysqli_query ($link, $query_lmnotes) or die (mysqli_error());  
$row_lmnotes = mysqli_fetch_assoc ($lmnotes);  
$lmnotes_num = $row_lmnotes['lmnotes'];  
mysqli_free_result ($lmnotes);  
  
//$begin_date – первое число месяца,  
//$end_date – последнее число месяца.
```

3. Аналогичным образом вычислите количество комментариев за последний месяц.

Упражнение 3: Последняя добавленная заметка

1. Для вывода последней добавленной заметки необходимо использовать оператор LIMIT в конструкции SELECT.

⇒ Выражение LIMIT используется для ограничения количества строк, возвращенных командой SELECT. LIMIT принимает один или два

числовых аргумента. Эти аргументы должны быть целочисленными константами. Если заданы два аргумента, то первый указывает на начало первой возвращаемой строки, а второй задает максимальное количество возвращаемых строк. При этом смещение начальной строки равно 0, а не 1 (т.к. первый элемент массива строк имеет индекс 0).

Например:

```
SELECT * FROM table LIMIT 5,10; // возвращает строки 6-15
```

Вариант реализации кода

```
//Последняя добавленная заметка  
//Таблица notes сортируется по дате публикации заметки по убыванию, а  
затем из нее берется только самая первая запись (LIMIT 0,1) - "начиная с  
нулевой записи выбрать одну запись"
```

```

$query_last_note = "SELECT id, title FROM notes
                    ORDER BY created DESC LIMIT 0,1";
$lastnote = mysqli_query ($link, $query_last_note) or die (mysqli_error());
$row_lastnote = mysqli_fetch_assoc ($lastnote);
mysqli_free_result ($lastnote);

```

Упражнение 4: Самая комментируемая заметка

В ходе выполнения этого упражнения необходимо реализовать следующий алгоритм работы:

- Связать таблицы **notes** и **comments** по полям **id** и **art_id** соответственно, чтобы затем вычислить количество комментариев для каждой заметки;
- Выполнить группировку таблицы **comments** по идентификатору заметки.
- Вычислить количество комментариев для каждой заметки.
- Отсортировать результат по количеству комментариев для каждой заметки по убыванию.
- Вывести первую запись из получившегося набора записей.

1. Построение SQL-запроса

```

$query_mcnote = "SELECT notes.id, notes.title FROM comments, notes
                WHERE comments.art_id=notes.id

                GROUP BY notes.id
                ORDER BY COUNT(comments.id) DESC LIMIT 0,1";

```

В тексте запроса:

GROUP BY – оператор группировки. Группировка – это объединение записей в группы по какому-либо критерию (т.н. критерию группировки), который записывается сразу после оператора (в данном случае группировка по полю **id** таблицы **notes**).

ORDER BY COUNT (comments.id) DESC LIMIT 0,1 – осуществление сортировки по убыванию результатов выполнения агрегирующей функции **COUNT()** по **id** комментариев и вывод первой записи (т.е. записи с самым большим количеством комментариев).

Реализуйте данный запрос и поместите его результат в массив.

Упражнение 5: Размещение данных на странице

С помощью php-сценариев и оператора echo вывести результаты на страницу сайта. Ниже представлен вариант реализации кода вывода информации:

```
<html>
<body>
Сделано записей - <?php echo $allnotes_num; ?><br>
Оставлено комментариев - <?php echo $allcomments_num; ?><br>
За последний месяц я создал записей - <?php echo
                                $row_lmnotes['lmnotes'];?><br>
                                За последний месяц оставлено комментариев - <?php echo
                                $row_lmcomments['lmcomments'];?><br>
Моя последняя запись -
    <a href="comments.php?note=<?php echo $row_lastnote['id'];?>">
        <?php echo $row_lastnote['title'];?></a><br>
Самая обсуждаемая запись -
    <a href="comments.php?note=<?php echo $row_mcnote['id'];?>">
        <?php echo $row_mcnote['title'];?>
</a><br><br>

<p><a href="blog.php">Возврат на главную страницу сайта </a></p>
</body>
</html>
```

В представленном коде:

\$allnotes_num, *\$allcomments_num*, *\$row_imnotes*, *\$row_lmcomments*, *\$row_lastnote*, *\$row_mcnote* – массивы, в которые помещаются результаты выполнения функций *mysqli_fetch_array()*, вызываемых в ранее созданном коде для получения и хранения соответствующих данных.

Ниже представлен возможный вариант реализации всего кода страницы *inform.php* (без табличной html-структуры).

```
<?php require_once ("connections/MySiteDB.php");?>
```

```
<?php
```

```
mysqli_select_db ($link, $db);
```

```
//Вычисление количества заметок
```

```
$query_allnotes = "SELECT COUNT(id) AS allnotes FROM notes";
```

```
$allnotes = mysqli_query ($link, $query_allnotes) or die (mysqli_error());
```

```
$row_allnotes = mysqli_fetch_assoc ($allnotes);
```

```
$allnotes_num = $row_allnotes['allnotes'];
```

```
mysqli_free_result ($allnotes);
```

```
//Вычисление количества комментариев
```

```
$query_allcomments = "SELECT COUNT(id) AS allcomments FROM comments";
```

```
$allcomments = mysqli_query ($link, $query_allcomments) or die  
(mysqli_error());
```

```
$row_allcomments = mysqli_fetch_assoc ($allcomments);
```

```
$allcomments_num = $row_allcomments['allcomments'];
```

```
mysqli_free_result ($allcomments);
```

```
//Работа с датой
```

```
$date_array = getdate();
```

```
$begin_date = date ("Y-m-d", mktime(0,0,0, $date_array['mon'],1,  
                                     $date_array['year']));
```

```
$end_date = date ("Y-m-d", mktime(0,0,0, $date_array['mon'] + 1,0,  
                                     $date_array['year']));
```

```
//Заметки за последний месяц
```

```
$query_lmnotes = "SELECT COUNT(id) AS lmnotes FROM notes
```

```
WHERE created>='$begin_date' AND created<='$end_date'";
```

```
$lmnotes = mysqli_query ($link, $query_lmnotes) or die (mysqli_error());
```

```
$row_lmnotes = mysqli_fetch_assoc ($lmnotes);
```

```
$lmnotes_num = $row_lmnotes['lmnotes'];
```



```

mysqli_free_result ($lmnotes);

//Комментарии за последний месяц
$query_lmcomments = "SELECT COUNT(id) AS lmcomments FROM comments
                     WHERE created >= '$begin_date' AND created <=
'$end_date'";
$lmcomments = mysqli_query ($link, $query_lmcomments) or die (mysqli_error());
$row_lmcomments = mysqli_fetch_assoc ($lmcomments);
$lmcomments_num = $row_lmcomments['lmcomments'];
mysqli_free_result ($lmcomments);

//Последняя добавленная заметка
$query_last_note = "SELECT id, title FROM notes
                   ORDER BY created DESC LIMIT 0,1";
$lastnote = mysqli_query ($link, $query_last_note) or die (mysqli_error());
$row_lastnote = mysqli_fetch_assoc ($lastnote);
mysqli_free_result ($lastnote);

//Самая комментируемая заметка
$query_mcnote = "SELECT notes.id, notes.title FROM comments, notes
                WHERE comments.art_id=notes.id
                GROUP BY notes.id
                ORDER BY COUNT(comments.id) DESC LIMIT 0,1";
$mcnote = mysqli_query($link, $query_mcnote) or die (mysqli_error());
$row_mcnote = mysqli_fetch_assoc($mcnote);
mysqli_free_result ($mcnote);
?>

<html>
<body>
Сделано записей - <?php echo $allnotes_num; ?><br>
Оставлено комментариев - <?php echo $allcomments_num; ?><br>
За последний месяц я создал записей - <?php echo
                                     $row_lmnotes['lmnotes'];?><br>
За последний месяц оставлено комментариев - <?php echo
                                     $row_lmcomments['lmcomments'];?><br>
Моя последняя запись -
    <a href="comments.php?note=<?php echo $row_lastnote['id'];?>">
        <?php echo $row_lastnote['title'];?></a><br>
Самая обсуждаемая запись -
    <a href="comments.php?note=<?php echo $row_mcnote['id'];?>">

```

<?php echo \$row_mcnote['title'];?>

<p>Возврат на главную страницу сайта </p>

</body>

</html>