Запросы с групповыми операциями

Лабораторная работа 10

Работа 10. SQL выполняется на учебной базе "sakila"

БД "sakila" создается вручную (пустая).

А все наполнение – при помощи SQL-инструкций из файлов 1.sql и 2.sql

Соединение с БД

1. Соединение с сервером БД

```
   $lnk= mysql_connect("localhost", "username",
"password")
      or die("Couldnotconnect: " . mysql_error());
```

2. Выбор БД

```
   mysql_select_db("dbname", $lnk)
      or die('Can\'t usedbname: '. mysql_error());
```

Часто выносится в отдельный файл, подключаемый require_once("db_connect.php");

Проверка текущей ситуации на странице

Читаем шаблон формы в строковую переменную $page;

Проверяем – мы пришли по кнопке Submit?

```
// МЫ ВПЕРВЫЕ НА СТРАНИЦЕ?
    if(isset($_POST['submit'])) // нет, не впервые
```

Заполнение переменных из глобальных массивов

Переносим массивы $_POST и $_GET в переменные

так:

```
$chcat = isset($_POST['chcat']) ? true:false;
$chrat = isset($_POST['chrat']) ? true:false;
$chlong = isset($_POST['chlong']) ? true:false;
$chprice = isset($_POST['chprice']) ? true:false;
```

или так:

```
extract($_POST);
```

Изменения в БД

- Если есть массив $_POST – выполняем действия для кнопки Submit (например, добавляем запись)
- Если есть $_GET – выполняем действия для гиперссылки (например, удаляем запись)
- В обоих случаях нам придется сформировать запрос и отправить его на выполнение.

Отображение данных на странице

Формируем запрос SELECT для выборки

(можно изначально сделать фиксированный запрос, например

```
 $sql = "SELECT
`film`.`film_id`,`title`,`rental_rate`,`length`,`rating`,`category
_id` ";
    $sql .= "FROM `film`,`film_category` ";
    $sql .= "WHERE `rental_rate` > 2 AND `rental_rate` < 3 ";
    $sql .= "AND `length` > 100 AND `length` < 200 ";
    $sql .= "AND `rating` = 'G' ";
    $sql .= "AND `category_id` = 1 ";
    $sql .= "AND `film`.`film_id` = `film_category`.`film_id` ";
    $sql .= "LIMIT 0, 30";
```

чтобы убедиться, что все работает. А затем постепенно подставлять значения полей.

Отображение данных на странице

Выполняем запрос

```
$result = mysql_query($sql);
```

В цикле каждую строку запроса переноси в массив (здесь – ассоциативный)

```
while($date = mysql_fetch_assoc($result))
```

Формируем строку для вывода на страницу:

```
$str = $date['id']." ".$date['name']."<br>";
```

Заменяем плейсхолдер на странице на полученную строку:

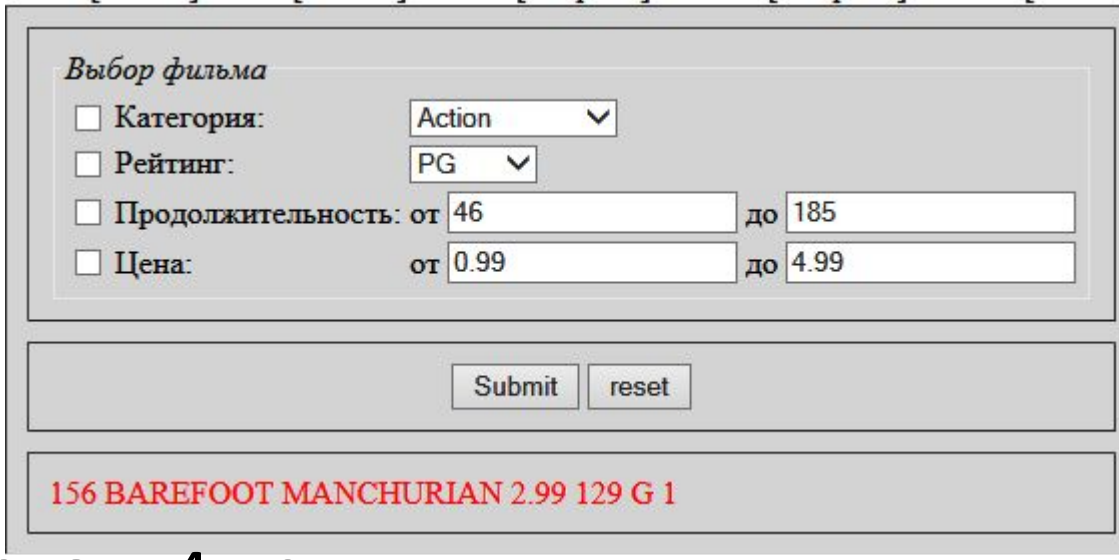
```
$page = str_replace('{max}', $str, $page);
```

Выводим страницу

```
echo $page;
```

ВСЕ!

Задание 2

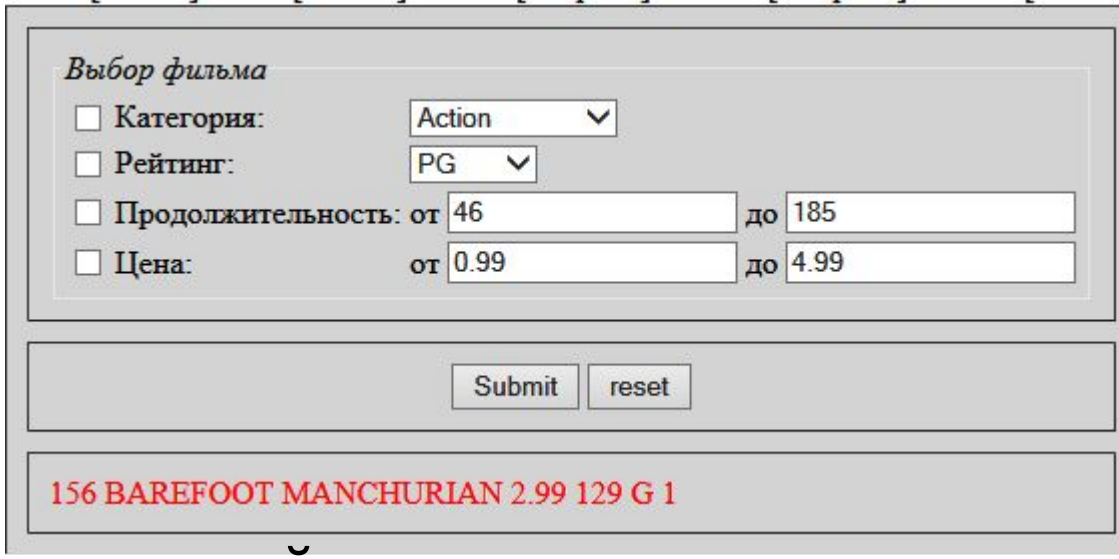


Требуется сразу же заполнить 2 списка и 4 поля:

Для получения диапазона продолжительности и цены используйте операции MIN и MAX

```
$sql = "SELECT MIN(`len`) minlen FROM `film`;";
```

Задание 2



Для получения списка постройте запрос, который вернет все категории или все варианты рейтинга:

```
$sql="SELECT DISTINCT `r` FROM `film`;";
```

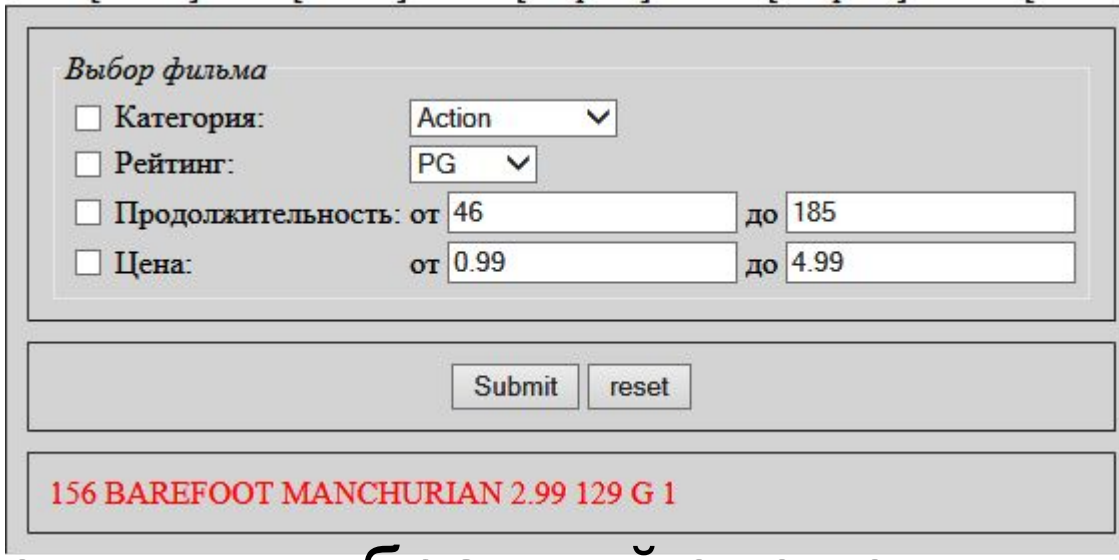
И заполните соответствующий плейсхолдер, добавляя теги списка:

```
$optionrat .= "<option ".$date[rating]."</option>";//в цикле

...

$page = str_replace('{optionrat}', $optionrat, $page);
```

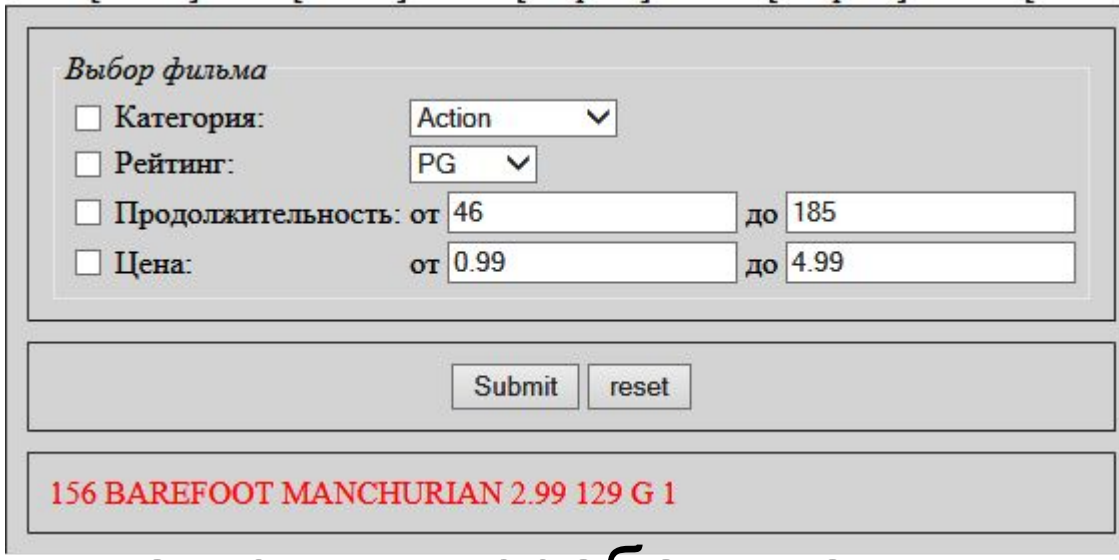
Задание 2.
Для профессионалов



При повторном вызове формы хорошо помнить выбранный элемент списка:

```
$optionrat .= "<option ".(($rat ==
$date[rating])?"selected ":"").value =
'$date[rating]'> $date[rating] </option>";
```

Задание 2



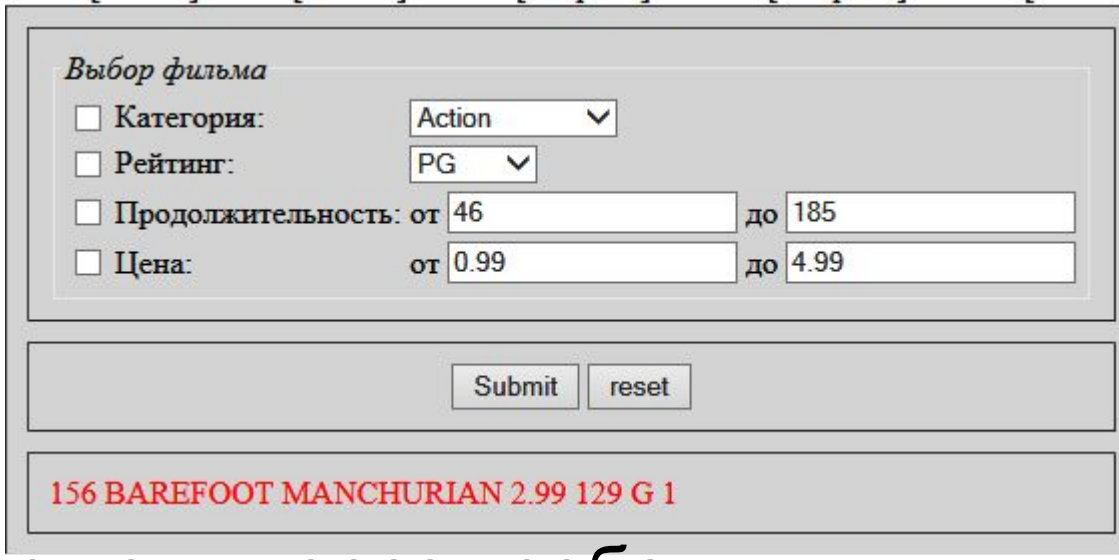
При построении запроса для выбора фильма должны срабатывать только те фильтры, возле которых помечены checkbox-ы. Поэтому сначала формируем начало инструкции SQL:

```
$sql = "SELECT `film`.`film_id`,`title`,`rental_rate`,".
"`length`,`rating`,`category_id` FROM `film`,`film_category` ".
"WHERE `film`.`film_id` = `film_category`.`film_id` ";
```

А затем добавляем фильтры:

```
if ($chcat) $sql .= " AND `category_id` = '$cat'";
```

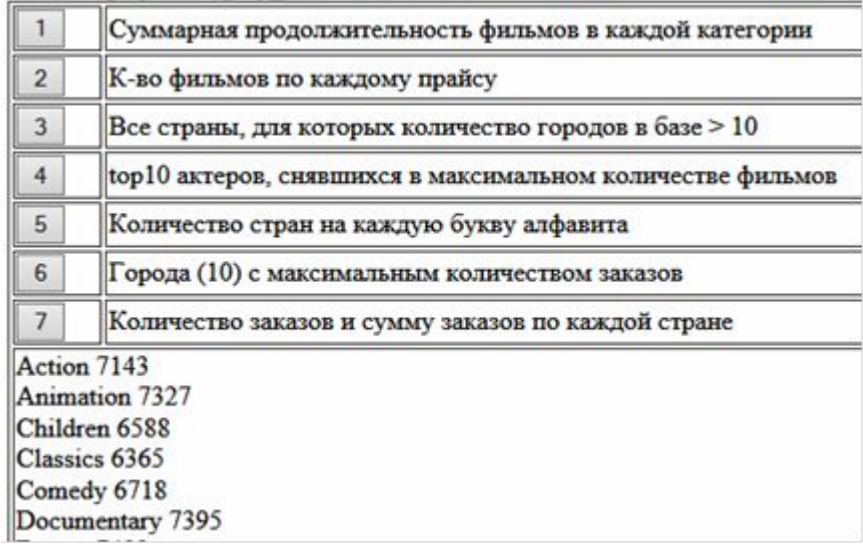
Задание 2



При разборе запроса просто выводим все поля через пробел и добавляем переход на новую строку:

```
$date = array_values($date);
for($i=0; $i<count($date); $i++)
{
    $str_info .= $date[$i]." ";
}
$str_info .="<br>";
```

Задание 3



Здесь все кнопки имеют одно имя, но разные значения.
Выбор можно построить через switch

```
switch($m)
{
    case 1: // Суммарная продолжительность фильмов в каждой категории
                                        break;
    case 2: // К-во фильмов по каждому прайсу
                                        break;
    case 3: // Все страны, для которых количество городов в базе > 10
                                        break;
    case 4: // top10 актеров, снявшихся в максимальном количестве
фильмов
```

Задание 3

1	Суммарная продолжительность фильмов в каждой категории
2	К-во фильмов по каждому прайсу
3	Все страны, для которых количество городов в базе > 10
4	top10 актеров, снявшихся в максимальном количестве фильмов
5	Количество стран на каждую букву алфавита
6	Города (10) с максимальным количеством заказов
7	Количество заказов и сумму заказов по каждой стране

Action 7143
Animation 7327
Children 6588
Classics 6365
Comedy 6718
Documentary 7395

Получение первой буквы в имени:

```
LEFT(`country`,1)
```

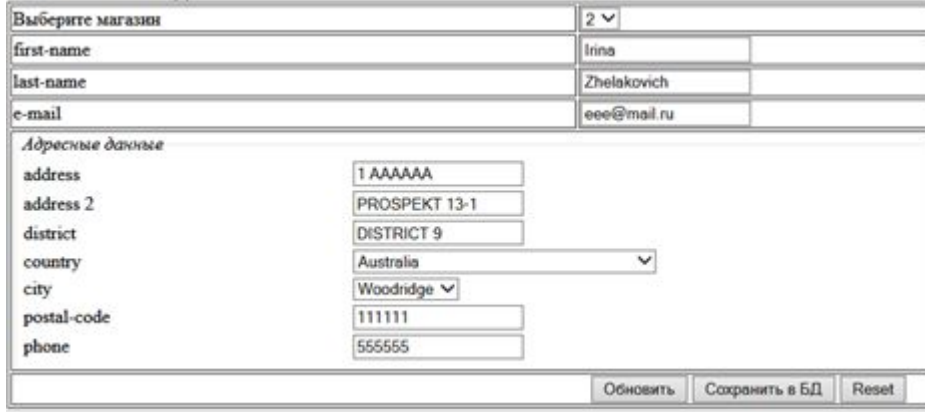
Для любого вычисляемого поля можно добавить альтернативное имя и затем использовать его в выражениях:

```
$sql = "SELECT `country`, COUNT(`city_id`) cc ";

...

$sql .= "HAVING cc > 10;";
```

Задание 4



Поскольку списки здесь взаимосвязаны (перечень городов зависит от страны), введена кнопка ОБНОВИТЬ.

Для добавления записи в таблицу `customer` нужно иметь последний ID при добавлении в таблицу `address`:

```
$sql = "SELECT LAST_INSERT_ID();";
```