

Урок №8

Мини-Фреймворк для создания сайтов

В этом уроке

- Вы узнаете, что такое ЧПУ и зачем они используются;
- как организовать переписывание URL-адресов;
- изучите финальную версию мини-Фреймворка для создания сайтов.

1. ЧПУ

ЧПУ (человекопонятный урл) – веб-адреса, удобные для восприятия человеком.

Просто сравните два варианта:

```
http://mysite.ru/index.php?c=articles&act=show&id=25
```

и

```
http://mysite.ru/articles/show/25
```

Думаю, разница очевидна. Конечно, обычные пользователи редко смотрят на URL-адреса, однако, владельцы сайтов в 99% случаев просят ввести ЧПУ, так как некрасивые URL-адреса очень не любят роботы поисковых систем. Иными словами, введение ЧПУ необходимо для SEO.

Общая идея создания ЧПУ заключается в том, что пользователь на экране видит один адрес, а PHP-скрипт на сервере - другой. Для того, чтобы включить переписывание URL-адресов, необходимо в корне сайта создать файл .htaccess со следующим содержимым:

```
# Включаем модуль переписывания
RewriteEngine On

# Путь до корня сайта
RewriteBase /

# Исключения - адреса, которые не нужно переписывать
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d

# Правила изменения URL-адреса
RewriteRule .* index.php?q=$0
```

Сначала разберёмся с самой системой изменения адреса. Строка

```
RewriteRule .* index.php?q=$0
```

означает, что любой адрес, введённый в браузере для PHP-скрипта, будет превращён в конструкции `index.php?q=«Введённый адрес»`.

Например, если человек видит на экране

```
http://mysite.ru/articles/show/25
```

, то после переписывания URL, PHP-скрипт работает с адресом

```
http://mysite.ru/index.php?q=articles/show/25
```

Т.е., в параметр «q» передаётся та часть адреса, которая не является доменным именем.

Для того, чтобы теперь в PHP-скрипте ориентироваться на полученные параметры, их нужно разбить по слешу, например:

```
$info = explode('/', $_GET['q']); // для index.php?q=$0
$params = array();

foreach ($info as $v)
{
    if ($v != '')
        $params[] = $v;
}
```

Помимо самого разбиения, мы запускаем цикл, который исключает из итогового массива с параметрами пустые значения. Они могли бы возникнуть в случае написания URL-адреса со слешем на конце:

```
http://mysite.ru/articles/show/25/
```

В итоге, после такой перегонки, у нас возникает массив \$params, который, по сути, является аналогом массива \$_GET. Единственное отличие заключается в том, что \$_GET – ассоциативный массив, а \$params – нет.

О том, что дальше делать с полученными параметрами, поговорим в следующем пункте, а пока что обратите внимание на строки с исключениями:

```
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
```

Они означают, что если в браузере введён прямой адрес до файла или папки на сервере, то его переписывать не нужно. Действительно, если мы обращаемся к сайту по адресу

```
http://mysite.ru/images/1.jpg
```

, то наверняка стоит просто показать картинку, а не запускать PHP-скрипт с адресом

```
http://mysite.ru/index.php?q=images/1.jpg
```

То же самое относится и к подключению css-стилей, скриптов и т.п. Без прописанных исключений сайт будет работать некорректно.

Последняя строка, которую мы не разобрали,

```
RewriteBase /
```

означает, что переписывание адреса ведётся от корня сайта. Например, если бы мы размещали сайт не в корне, а в некоторой папке, то запись выглядела бы так:

```
RewriteBase /somedirectory/
```

2. Мини-Фреймворк

А теперь рассмотрим то, как можно внедрить ЧПУ в систему, созданную на пятом уроке. Для этого нам придётся немного изменить метод Request класса C_Controller:

```
public function Request($action, $params)
{
    $this->params = $params;
    $this->before();
    $this->$action();
    $this->render();
}
```

Как мы видим, теперь он принимает дополнительный параметр \$params и делает его полем класса. Собственно говоря, \$params – это и есть массив, который будет получен после анализа URL-адреса. Точку входа также нужно изменить – настроить на работу с массивом \$params, а не \$_GET, как это было раньше:

```
$info = explode('/', $_GET['q']);
$params = array();

foreach ($info as $v)
{
    if ($v != '')
        $params[] = $v;
}

// .....
$this->Request($action, $params);
```

Описание мини-Фреймворка:

- **index.php** – точка входа
- **config.php** – файл с константами настроек и функцией __autoload
- **.htaccess** – файл с настройками для apache, нужен для создания ЧПУ
- **Контроллеры:**
 - **C_Controller** – ядро системы – не изменять!
 - **C_Base** – общий контроллер для всех страниц сайта
 - **C_Auth** – контроллер страниц авторизации и регистрации
 - **C_Page** – контроллер для примера
- **Модели:**
 - **M_Users** – работа с пользователями
 - **M_MSQL** – вспомогательный модуль для работы с базой
 - **M_Pages** – модель для примера
- **Вьюшки:**
 - **v_main.php** – базовый шаблон
 - **v_index.php** – главная страница
 - **v_edit.php** – редактирование контента
 - **v_login.php** – авторизация

Для расширения системы на каждую новую сущность создаётся отдельный контроллер и наследуется от `C_Base`. Для создания новой странички необходимо добавить метод соответствующему контроллеру и выюшку под эту страницу.

Внимательно изучите финальную версию мини-Фреймворка.

Самоконтроль

- ✓ Что такое ЧПУ
- ✓ Почему сайты используют ЧПУ
- ✓ Для чего нужен файл `.htaccess`
- ✓ Как происходит переписывание URL-адреса
- ✓ Зачем нужны исключения
- ✓ `RewriteBase`
- ✓ Обработка полученного адреса в PHP-скрипте

Домашнее задание

1. Добавить в блог работу с пользователями.
2. Добавить переписывание url-адресов.