
Разработка многостраничного сайта на PHP.

ЗАНЯТИЕ № 3
HTTP – ЗАГОЛОВКИ ОТВЕТА СЕРВЕРА

Тема занятия: HTTP – заголовки ответа сервера

- Цель занятия –

Продолжить изучать HTTP – заголовки ответа сервера. Сегодня мы продолжим говорить про параметры запроса, вспоминать некоторые понятия из прошлого урока.

Глоссарий :

- URI – это универсальный идентификатор ресурса.
- Header() – функция для отправки или смены заголовков
- Сервер - это специальное оборудование для выполнения сервисного программного обеспечения.
- Заголовок запроса - это метайнформация, отправленная браузером при запросе сценария

Актуализация

- 1) Как происходит добавление и изменение заголовков ответа?
- 2) Как происходит управление телом ответа?
- 3) Какие есть параметры запроса?
- 4) Из чего состоит URI?
- 5) Какие есть параметры запроса?
- 6) Как происходит формирование URI с параметром запроса?

Добавление/ изменение заголовков ответа

В PHP-сценарии можно управлять всеми заголовками ответа, которые попадут к пользователю вместе с контентом страницы. Это возможно, потому что PHP работает на стороне веб-сервера и имеет с ним очень тесную интеграцию. Вот примеры сценариев, когда пригодится управление заголовками ответа:

- Кэширование;
- Переадресация пользователя;
- Установка cookies;
- Отправка файлов;
- Передача дополнительной информации браузеру.

Добавление/ изменение заголовков ответа

Заголовки ответа нужны для выполнения множества важных задач.

В PHP есть функция для отправки или смены заголовков: header().

Она принимает имя и значение заголовка и добавляет его в список из всех заголовков, которые уйдут в браузер пользователя после окончания работы сценария.

```
header("Location: /index.php");
```

Пример того, как выполняется перенаправление пользователя на другую страницу.

За переадресацию отвечает заголовок с именем Location, а через двоеточие задаётся значение — адрес страницы для перехода.

ПРИМЕЧАНИЕ!!!

Есть одно ограничение: заголовки нельзя отправлять, если пользователю к этому моменту уже отправили любой контент. То есть если показать что-то на экране, например, через функцию `print()`, то после этого заголовки поменять уже не получится.

Управление телом ответа

Всё, что PHP выводит на экран, является содержимым ответа. Иными словами, вызовы функций `print`, `echo` или показ текста через шорт-теги являются телом ответа, которое попадает в браузер пользователю.

Параметры запроса

Обычно каждый PHP-сценарий отвечает за одну страницу. Посетитель сайта вводит в адресную строку путь, который состоит из имени домена и имени PHP-сценария.

Например,

[HTTP://WEATHER-DIARY.RU/DAY.PHP](http://weather-diary.ru/day.php)

В случае, если одна страница должна показывать разную информацию, на сайте дневника погоды надо сделать отдельную страницу, чтобы показывать на ней информацию о погоде из истории за один конкретный день. То есть страница одна, но показывает разные данные в зависимости от выбранного дня.

URI

URI – это уникальный идентификатор ресурса.

Ресурс (в нашем случае) – это полный путь до страницы сайта. И вот как может выглядеть ресурс для показа погоды за конкретный день:

[HTTP://WEATHER-DIARY.RY/DAY.PHP?DATE=2023-02-12](http://weather-diary.ru/day.php?date=2023-02-12)

Далее разберем подробнее, из чего состоит URI.



Из чего состоит URI

[HTTP://WEATHER-DIARY.RY/DAY.PHP?DATE=2023-02-12](http://weather-diary.ru/day.php?date=2023-02-12)

Здесь есть имя домена: weather-diary.ru.

Затем идёт имя сценария: day.php.

А всё, что идёт после — это параметры запроса.

Параметры запроса — это как бы дополнительные атрибуты адреса страницы. Они отделяются от имени страницы знаком запроса. В примере выше параметр запроса только один: date=2023-02-12.

Имя этого параметра: date, значение: 2023-02-12.

Параметров запроса может быть несколько, тогда они разделяются знаком амперсанда:

?date=2023-02-12&tscale=celsius.

В примере выше указывается два аргумента: дата и единица измерения температуры.

Параметры запроса – как внешние переменные

Теперь в адресе страницы используются параметры запроса. Нам это полезно тем, что если имя страницы вызывает соответствующий сценарий PHP, то параметры запроса превращаются в специальные внешние переменные в этом сценарии.

Таким образом, если в адресе присутствуют такие параметры, то их легко получить внутри кода сценария и выполнить с ними какие-нибудь действия. Например, показать погоду за конкретный день в выбранных единицах измерения.

Получение параметров запроса

Все параметры запроса находятся в специальном, ассоциативном массиве `$_GET`, а значит, сценарий, вызванный с таким адресом: `day.php?date=2023-02-12&tscale=celsius` будет иметь в этом массиве два значения с ключами `date` и `scale`.

Запрос на получение данных за выбранный день выглядит так:

```
<?php
$date = $_GET['date'] ?? date('Y-m-d');
$sql = sprintf('SELECT * FROM weather WHERE day = "%s"',
    mysqli_real_escape_string($con, $date));
?>
```

Получение параметров запроса

В первой строчке примера выше мы получаем значение параметра date, а если он отсутствует, то используем текущую дату в качестве выбранного дня.

Никогда не полагайтесь на существование параметра в массиве \$_GET и делайте проверку либо функцией isset(), либо как в этом примере.

```
<?php
$date = $_GET['date'] ?? date('Y-m-d');
$sql = sprintf('SELECT * FROM weather WHERE day = "%s"',
    mysqli_real_escape_string($con, $date));
?>
```

Формирование URI с параметрами запроса

Иногда нужно совершить обратную операцию: сформировать адрес страницы, включив туда нужные параметры запроса из массива.

Скажем, на странице погодного дневника надо поставить ссылку на следующий и предыдущий день. Нужно также сохранить выбранную единицу измерений. То есть необходимо сохранить текущие параметры запроса, поменять значение одного из них (день), и сформировать новую ссылку.

В следующем примере мы использовали две функции:

- `basename(FILE)` — получает имя текущего сценария;
- `http_build_query()` — преобразует ассоциативный массив в строку запроса.

Формирование URI с параметрами запроса

Вот как это можно сделать:

```
<?php
$params = $_GET;
$date = $params['date'] ?? date('Y-m-d');

$tomorrow = date('Y-m-d', strtotime('tomorrow', strtotime($date)));
$params['date'] = $tomorrow;

$url = basename(__FILE__) . '/?' . http_build_query($params);
print($url);
?>
```


Рефлексия

- Что мы прошли сегодня на занятии?
- Какие ваши впечатления от новой темы?
- Какие основные моменты вы можете выделить?
- Как происходит формирование URI?
- Из чего состоит URI?
- Какие вы узнали параметры запроса?
- Можно ли сказать, что параметры запроса - это внешние параметры?

A decorative border with floral and scrollwork motifs in the corners and along the sides of the text area.

**Спасибо
за
внимание**