Back-end

Web.Dev.
Courses

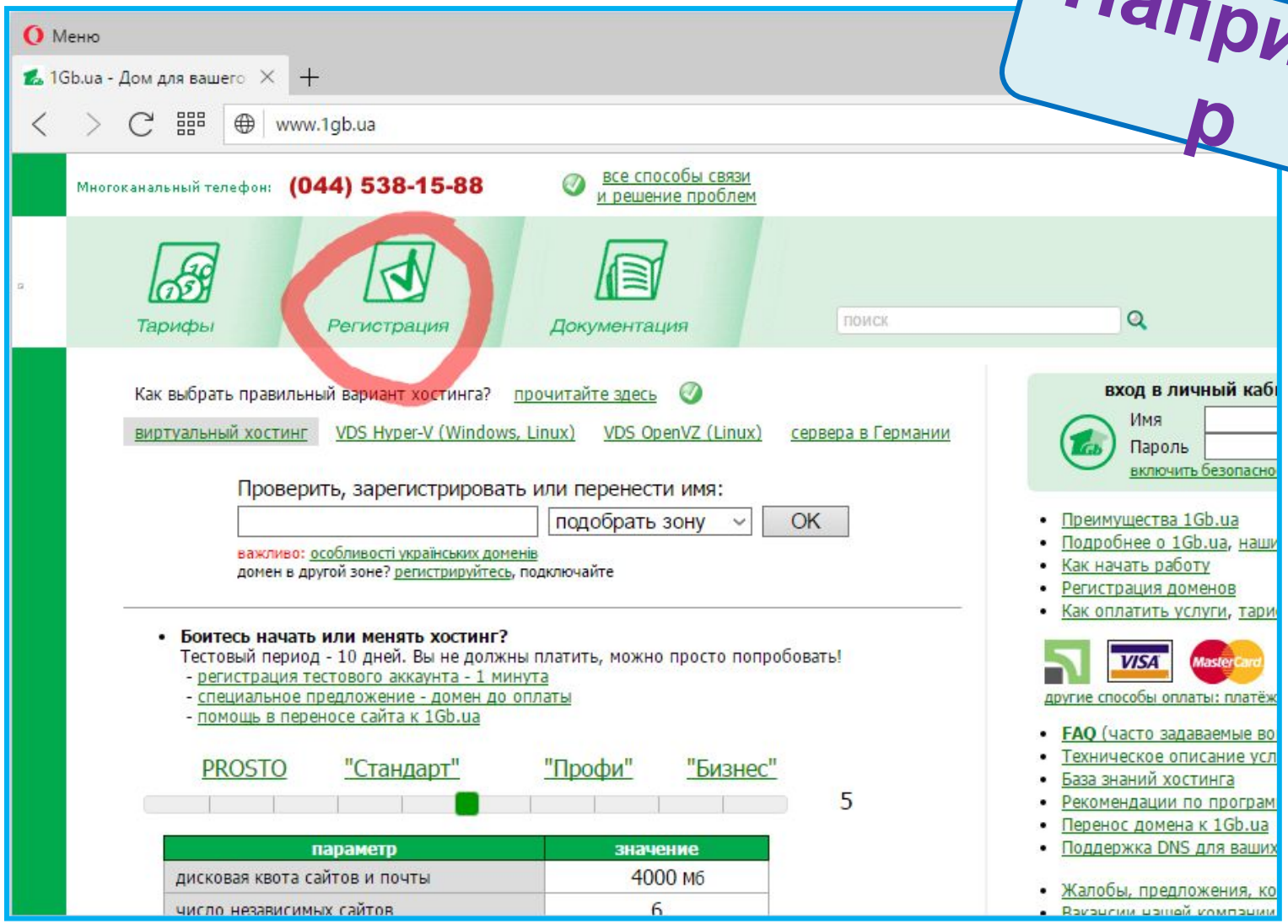
vk.com/**web.dev.courses**

web.dev.courses.dp.ua/files

Хостинг

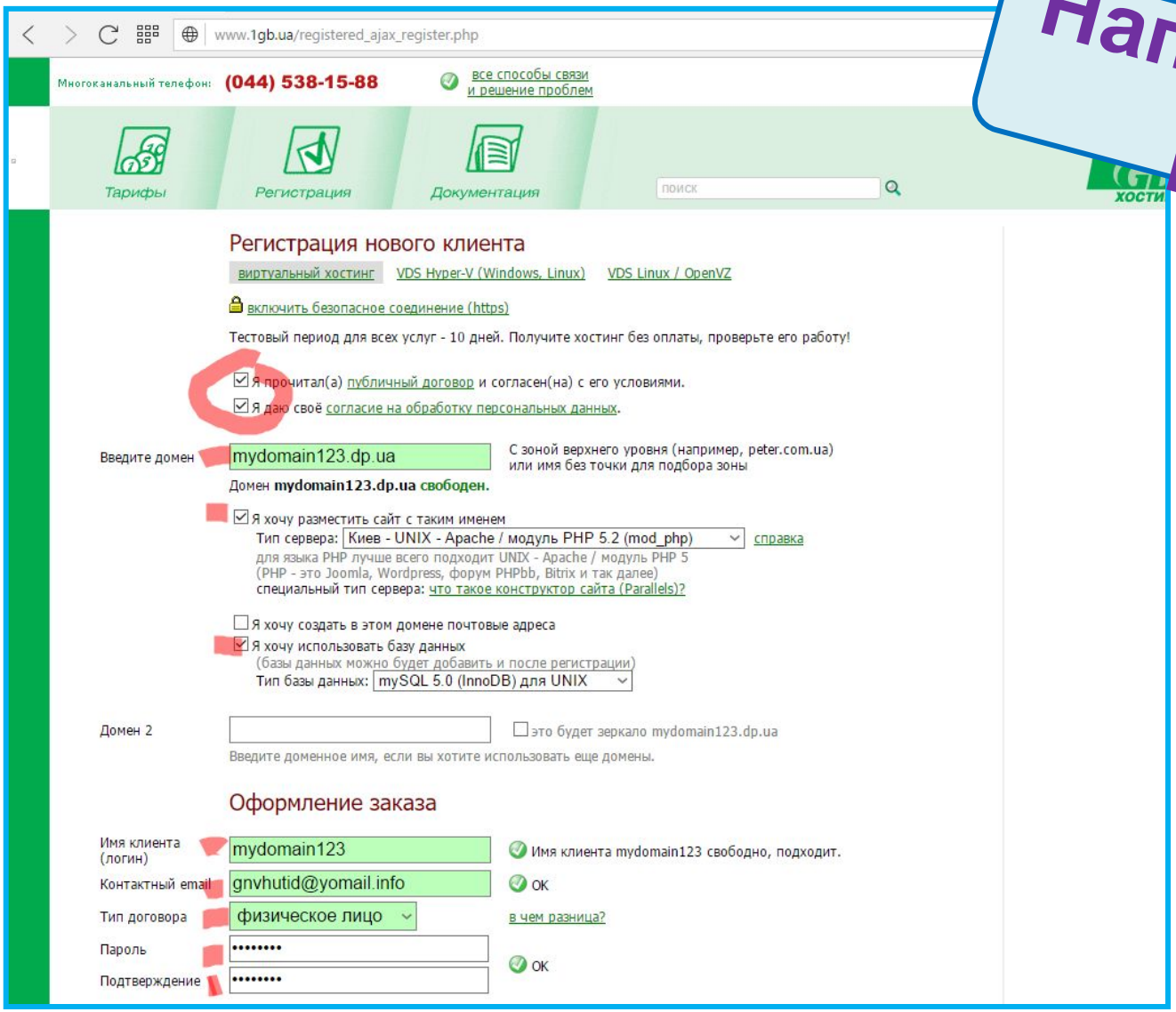
услуга по размещению информации на сервере, постоянно находящемся в Интернете.

Например



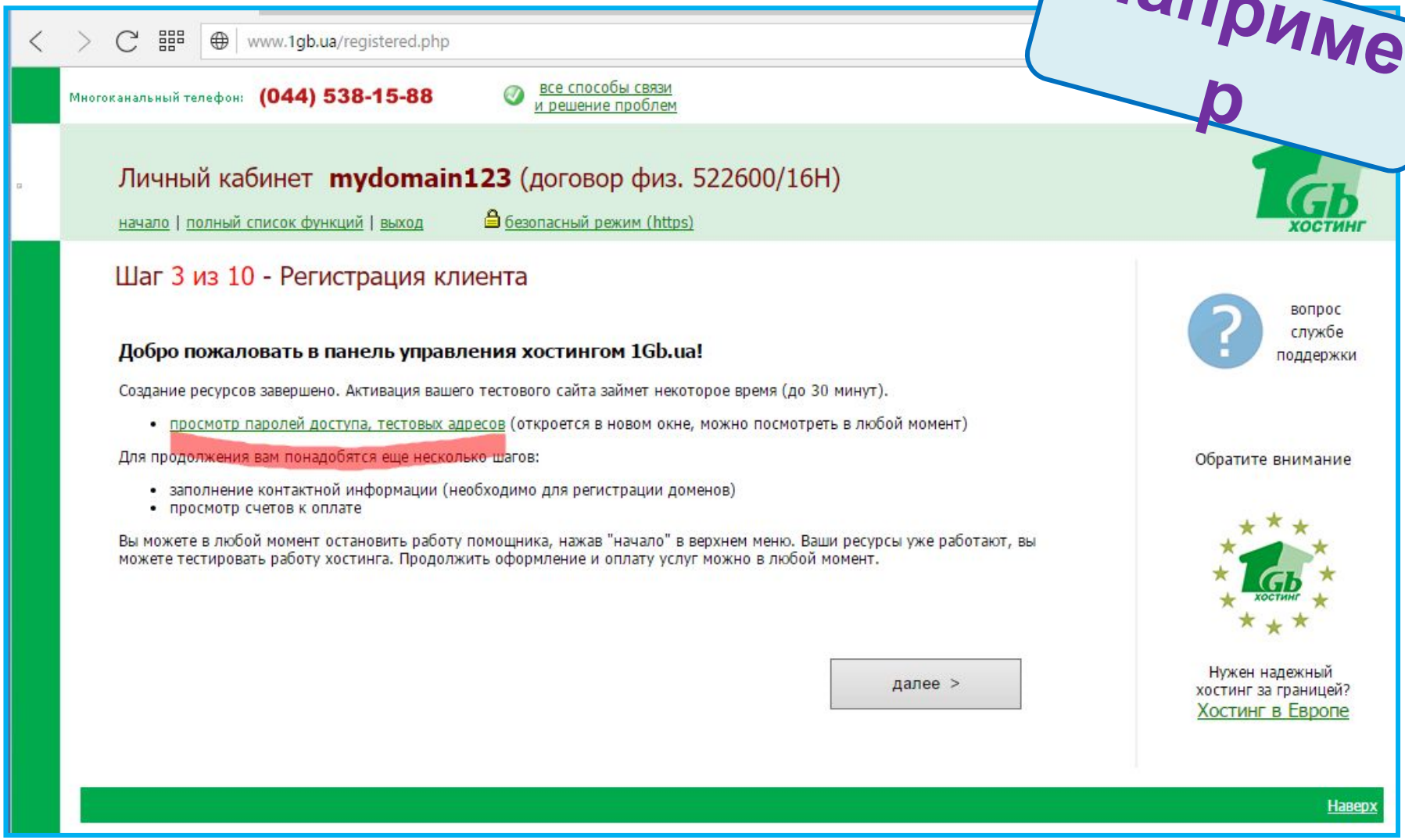
1gb.ua *– один из многочисленных хостинг-провайдеров, но у него есть пробный период!*

Например



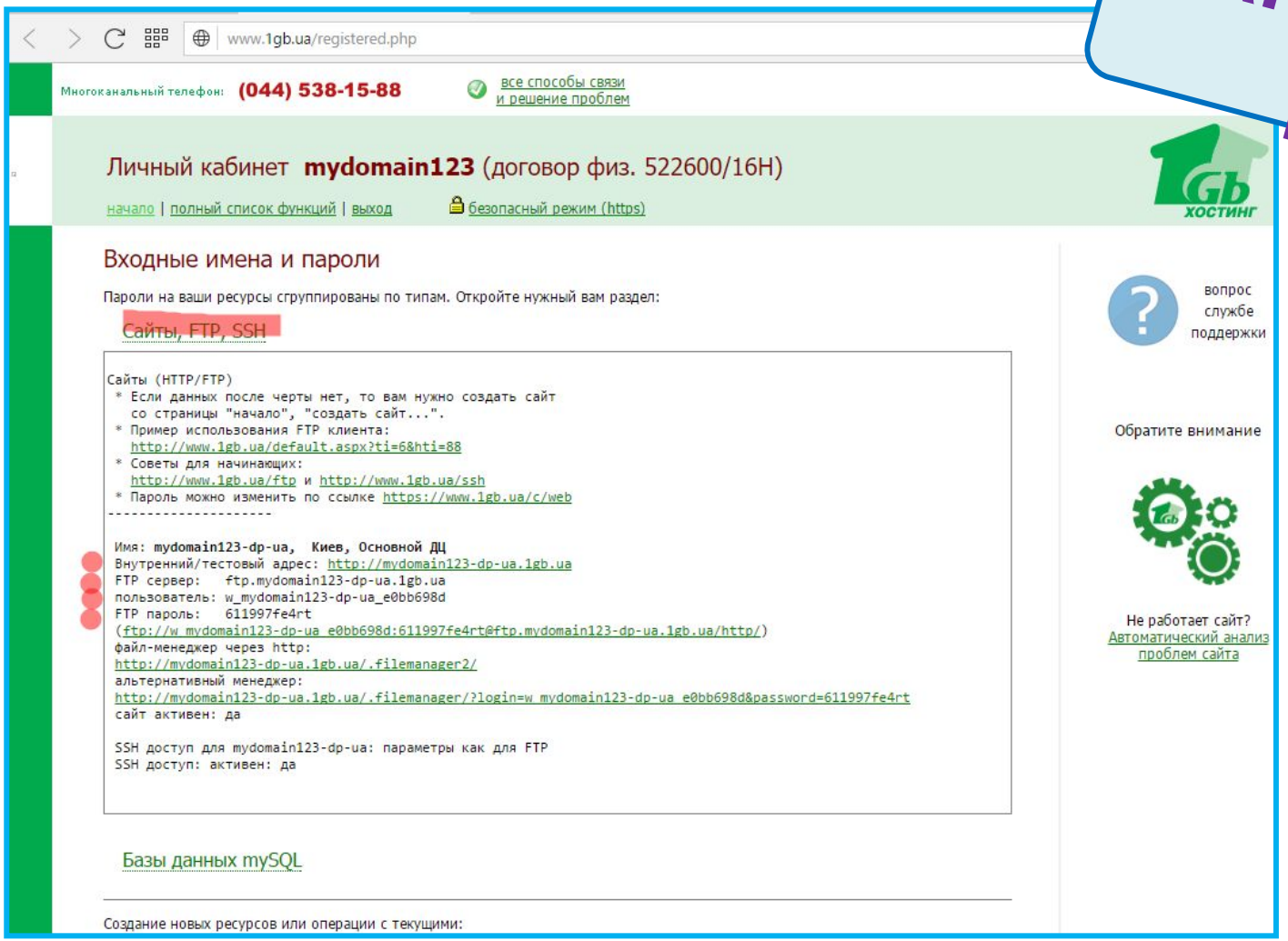
Регистрация не представляет особой сложности, необходимо ввести желаемое доменное имя, и стандартный набор данных о себе.

Например



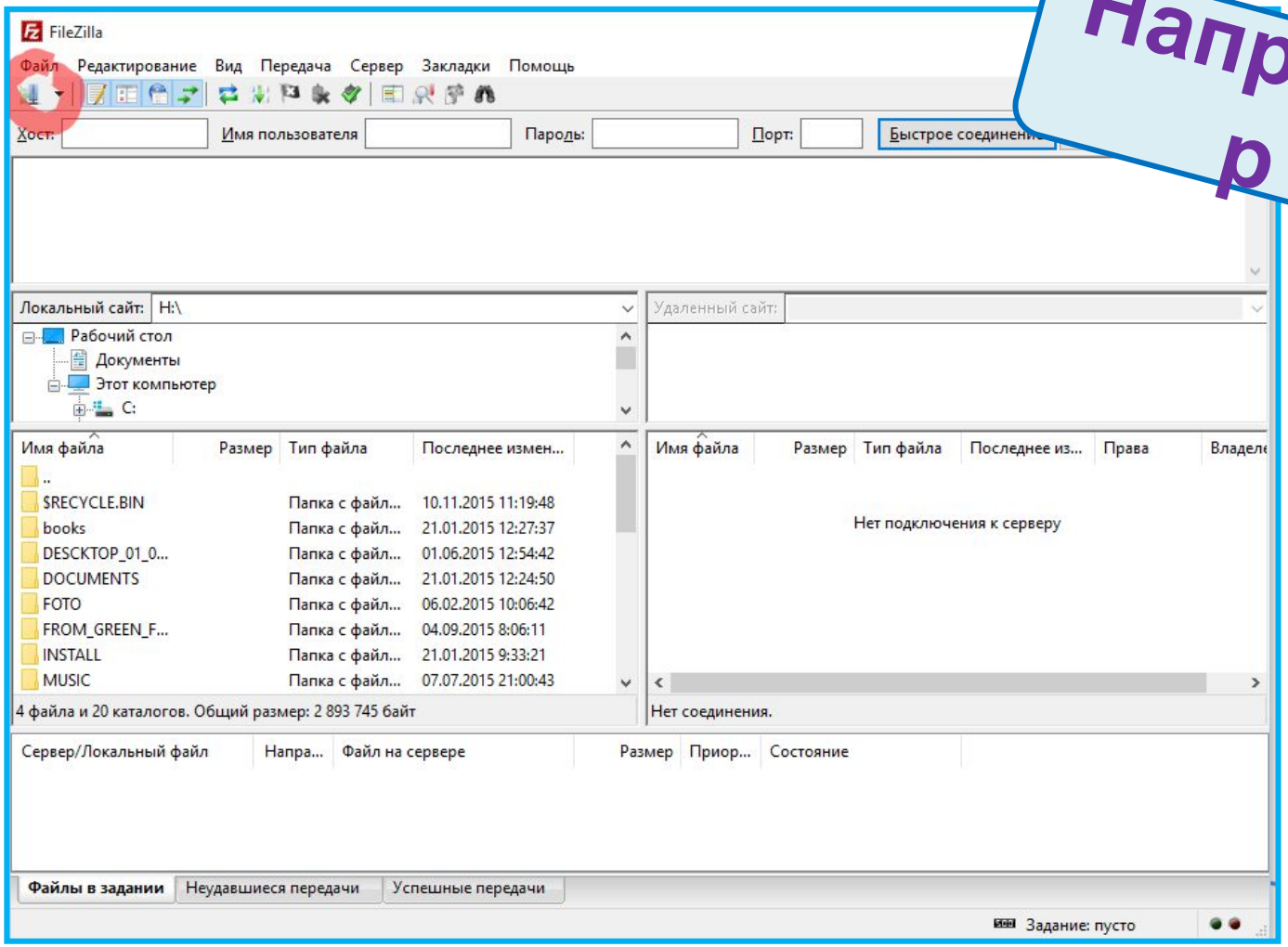
После регистрации нас будет интересовать данные о доступе к хостингу…

Например



…а именно адрес FTP-сервера, для закачки туда файлов нашего сайта, и логина пароля для доступа к нему (FTP-серверу).

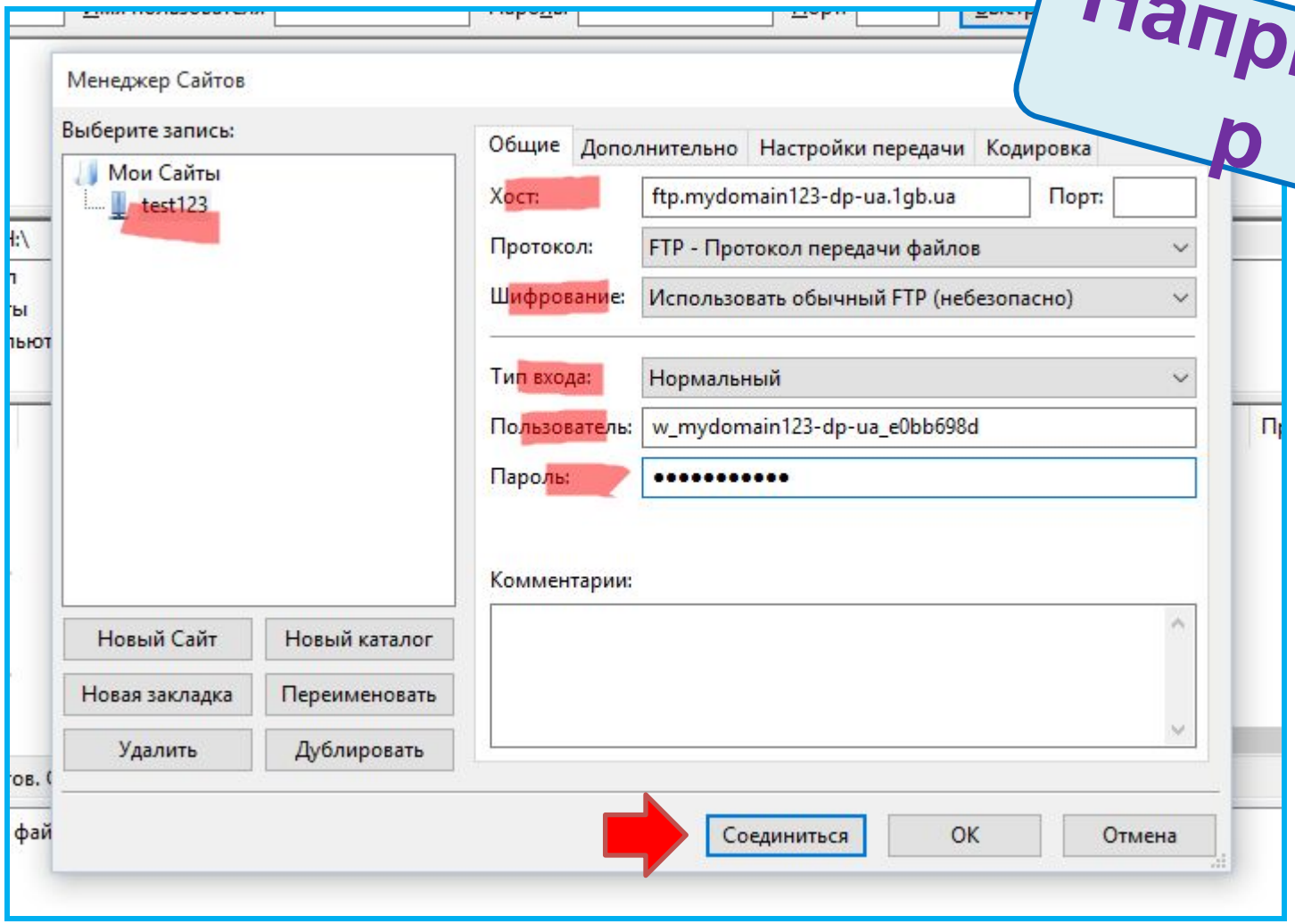
Например



FTP-клиент *– программа, которая по протоколу* ***FTP*** *сможет загружать файлы на хостинг (а также удалять их там при необходимости)* https://ru.wikipedia.org/wiki/FTP

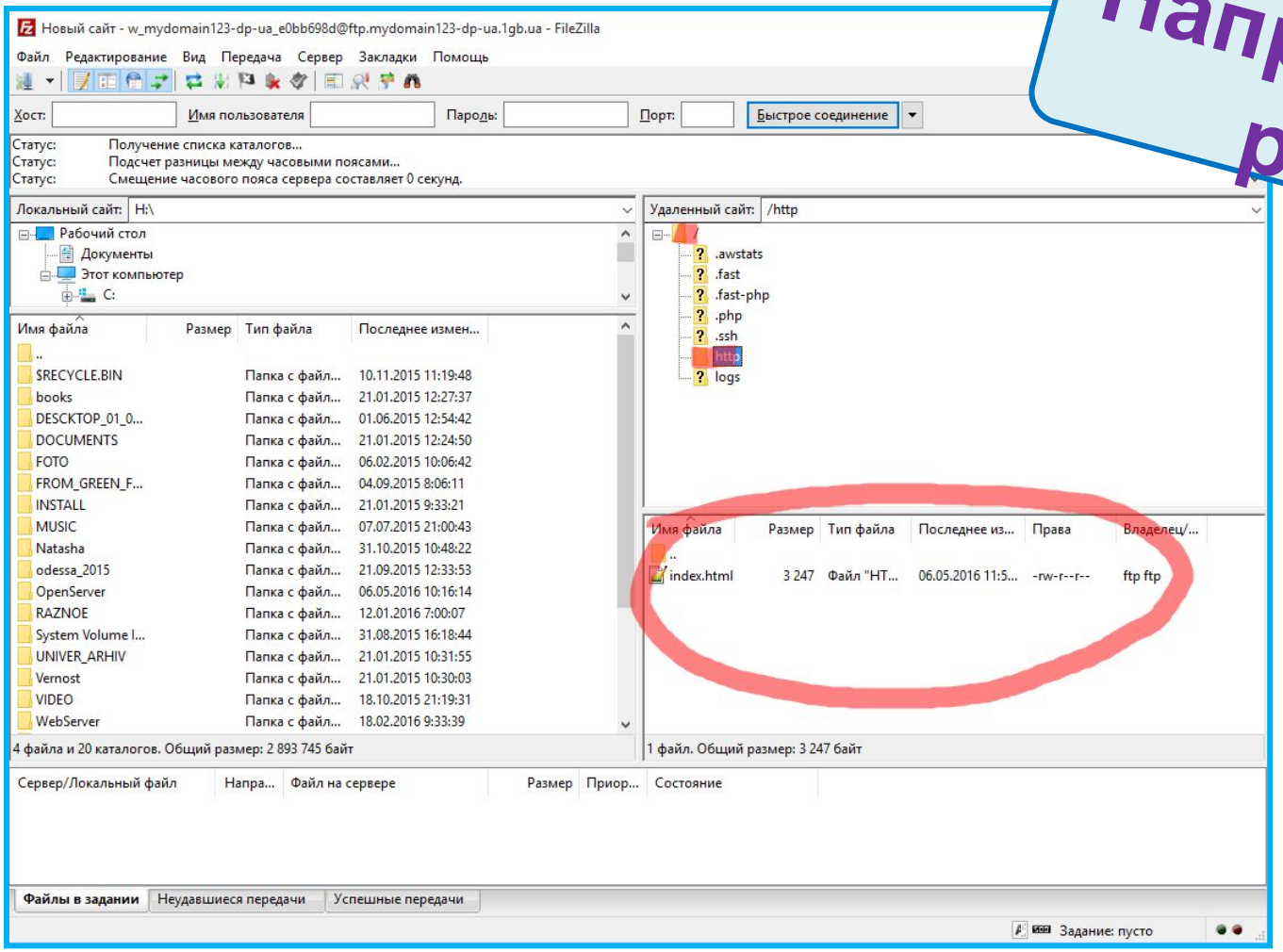
http://filezilla.ru/ *–* ***FileZilla*** *– один из популярных* ***FTP-клиентов***.

Например



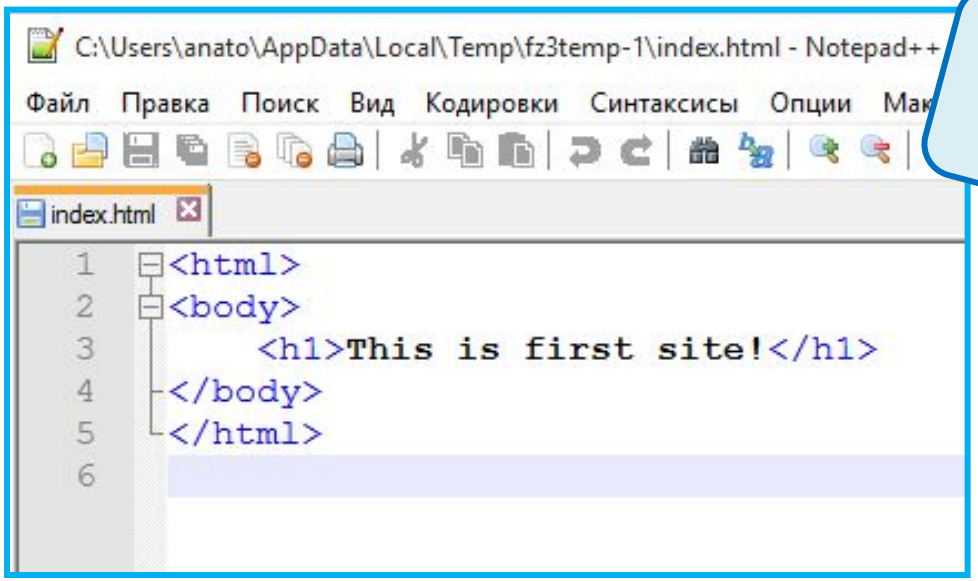
Создаём новое подключение и заполняем его данными которые мы получили на странице с адресами и пароля FTP-сервера (слайд №6).

Например

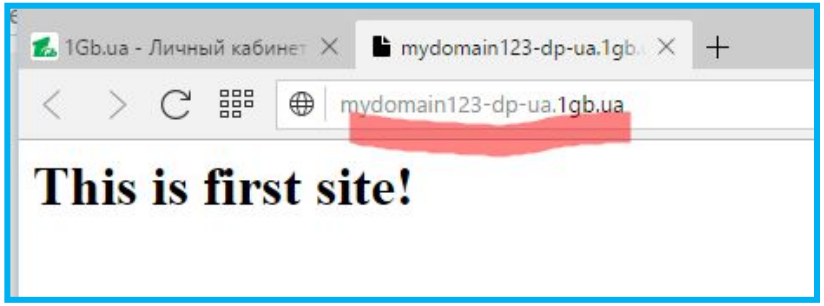


FileZilla *откроет соединение с сервером и в правой части окна будут расположены каталоги на сервере (как в проводинике), нас будет интересовать каталог* ***http****, в нём и будут лежать файлы вашего сайта.*

Например



*Поместите в этот каталог файлы из которых состоит ваш сайт. При этом один из файлов должен носить имя **index**.html – он станет файлом по умолчанию, т.е. если браузер не уточняет какой файл ему нужен, то ему будет возвращаться этот самый файл **index.**html.*



Зайдите в браузере на ваш сайт (его временный адрес есть на странице с логинами и паролями).

Имя первого файла (файла по умолчанию)
на сайта должно быть:

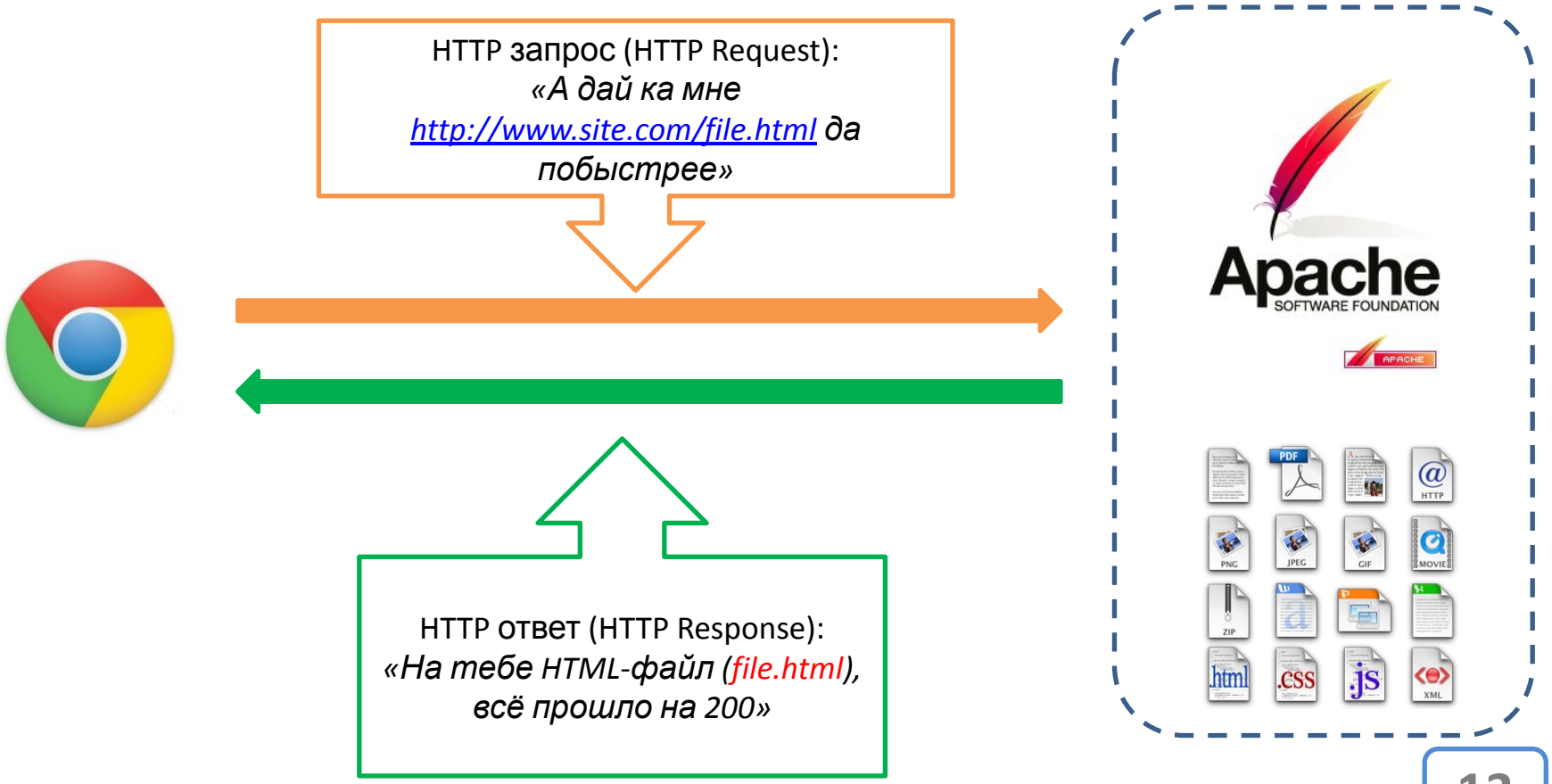
index.php
index.html
index.etc…

Например

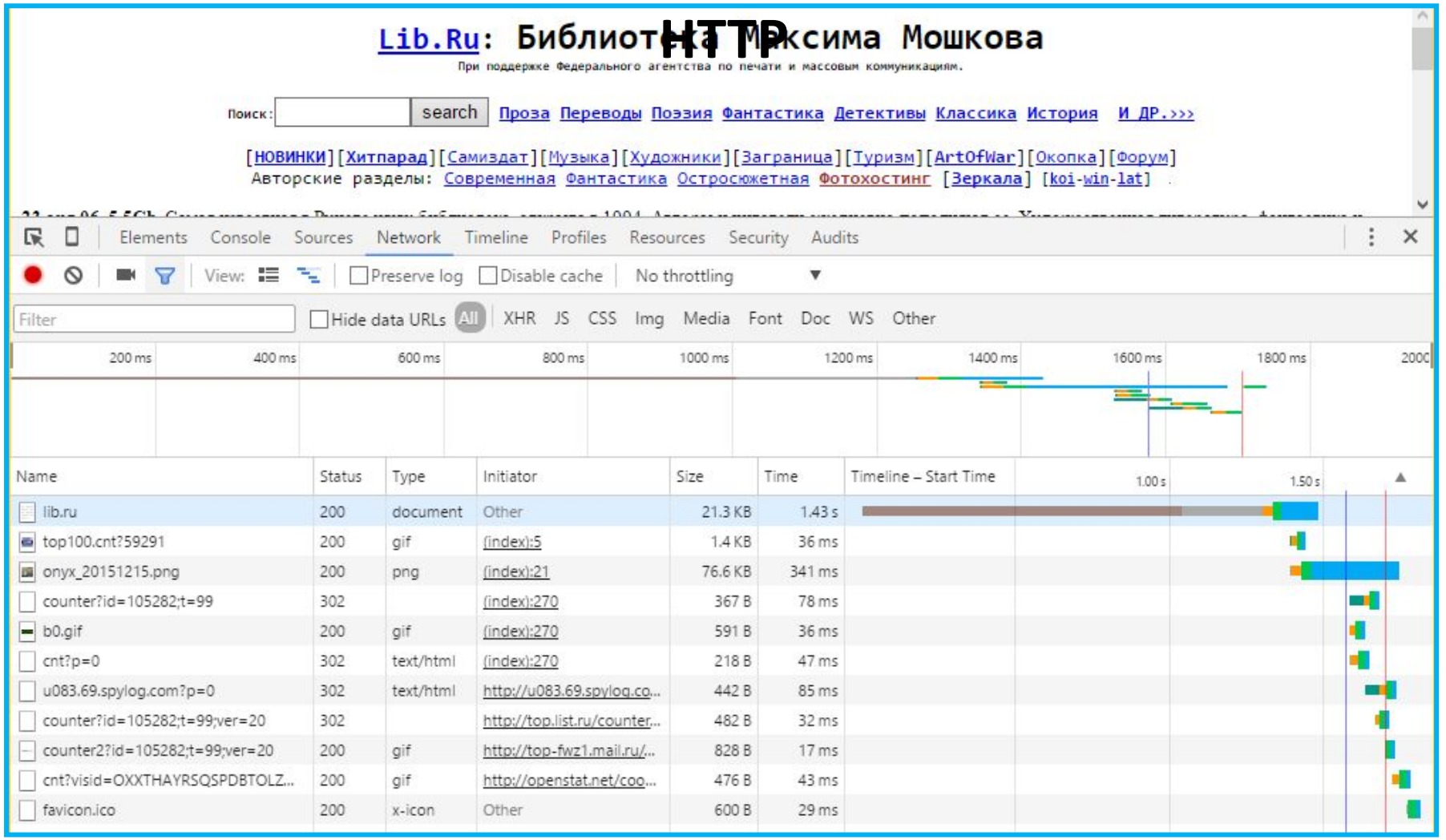
!!! Логины и пароли на слайдах в этом примере тестовые, т.е. не будут работать если вы их захотите использовать. Во время настоящей регистрации вы получить актуальные и работающие логины пароли которые и нужно использовать !!!

Как это всё работает?

Веб-сервер *— программа принимающий HTTP-запросы от клиентов (обычно* ***веб-браузеров****), и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, или другими данными.*

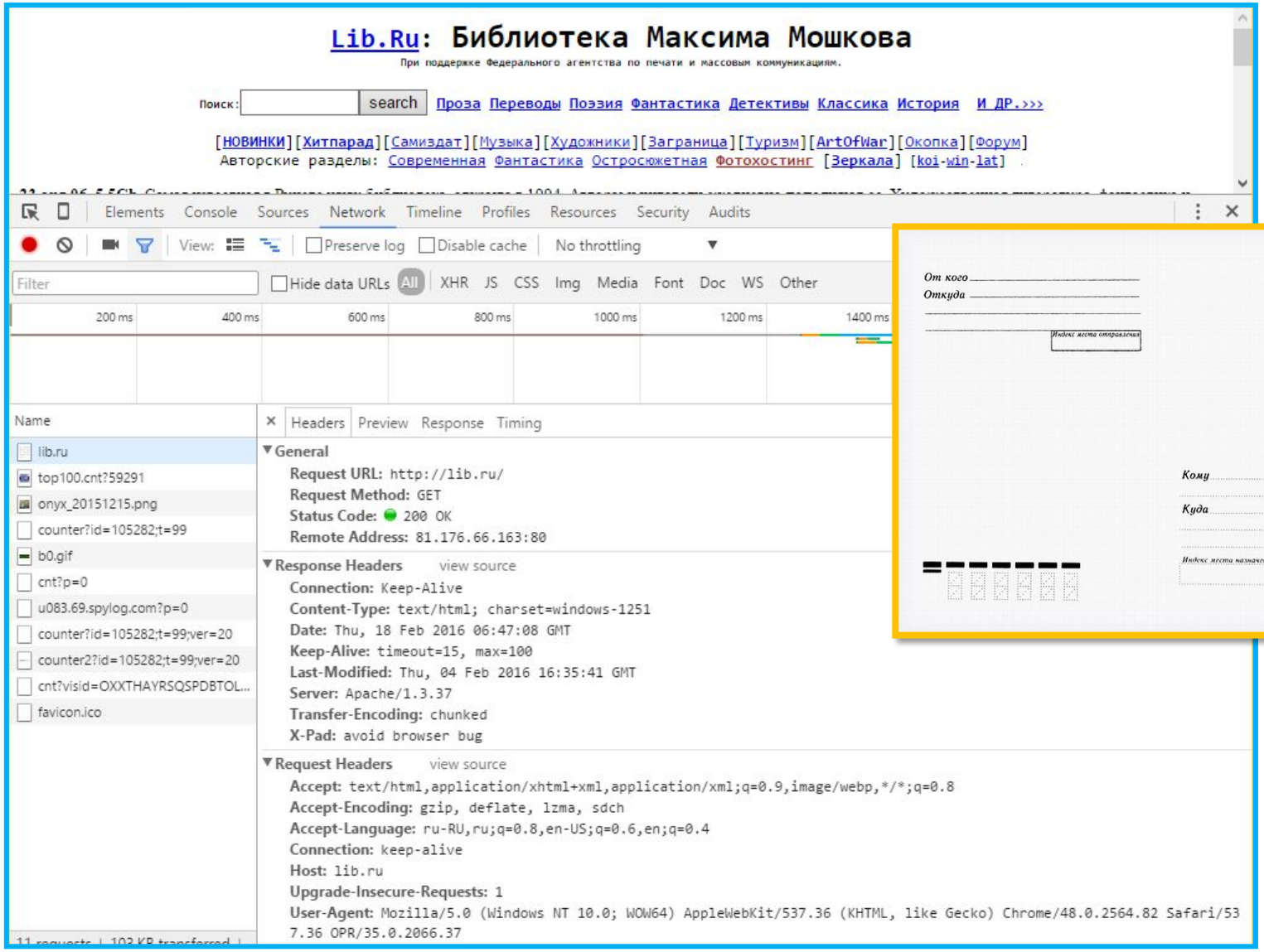


Протокол HTTP



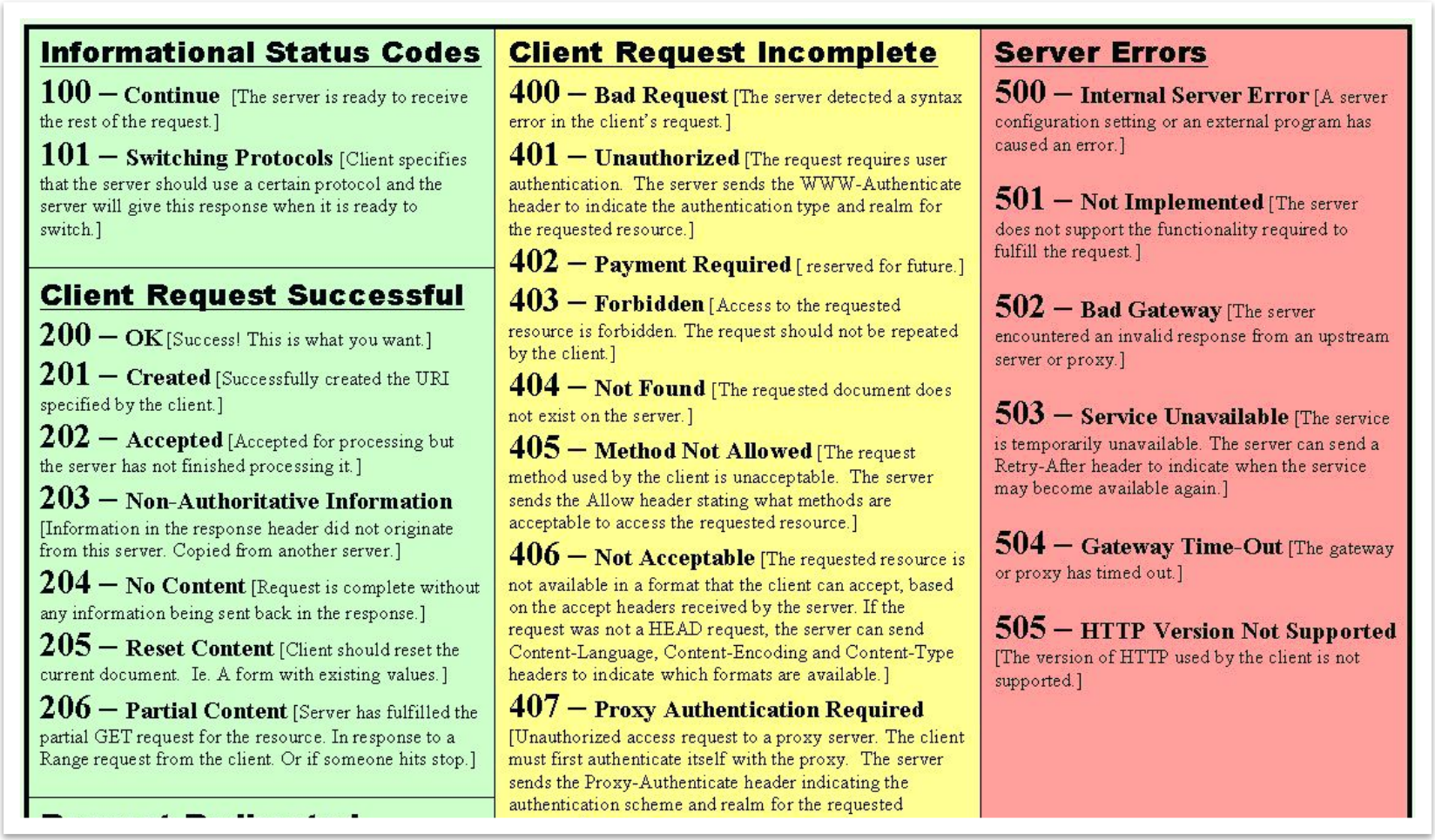
*Закладка **Network**, консоли разработчика, содержит все детали общения браузера с веб-серверами.*

Протокол



Браузер способен показать все детали HTTP-переписки между ним и сервером.

Коды состояний (статусов)



Informational Status Codes

100 – Continue [The server is ready to receive the rest of the request.]

101 – Switching Protocols [Client specifies that the server should use a certain protocol and the server will give this response when it is ready to switch.]

Client Request Successful

200 – OK [Success! This is what you want.]

201 – Created [Successfully created the URI specified by the client.]

202 – Accepted [Accepted for processing but the server has not finished processing it.]

203 – Non-Authoritative Information [Information in the response header did not originate from this server. Copied from another server.]

204 – No Content [Request is complete without any information being sent back in the response.]

205 – Reset Content [Client should reset the current document. Ie. A form with existing values.]

206 – Partial Content [Server has fulfilled the partial GET request for the resource. In response to a Range request from the client. Or if someone hits stop.]

Client Request Incomplete

400 – Bad Request [The server detected a syntax error in the client's request.]

401 – Unauthorized [The request requires user authentication. The server sends the WWW-Authenticate header to indicate the authentication type and realm for the requested resource.]

402 – Payment Required [reserved for future.]

403 – Forbidden [Access to the requested resource is forbidden. The request should not be repeated by the client.]

404 – Not Found [The requested document does not exist on the server.]

405 – Method Not Allowed [The request method used by the client is unacceptable. The server sends the Allow header stating what methods are acceptable to access the requested resource.]

406 – Not Acceptable [The requested resource is not available in a format that the client can accept, based on the accept headers received by the server. If the request was not a HEAD request, the server can send Content-Language, Content-Encoding and Content-Type headers to indicate which formats are available.]

407 – Proxy Authentication Required [Unauthorized access request to a proxy server. The client must first authenticate itself with the proxy. The server sends the Proxy-Authenticate header indicating the authentication scheme and realm for the requested

Server Errors

500 – Internal Server Error [A server configuration setting or an external program has caused an error.]

501 – Not Implemented [The server does not support the functionality required to fulfill the request.]

502 – Bad Gateway [The server encountered an invalid response from an upstream server or proxy.]

503 – Service Unavailable [The service is temporarily unavailable. The server can send a Retry-After header to indicate when the service may become available again.]

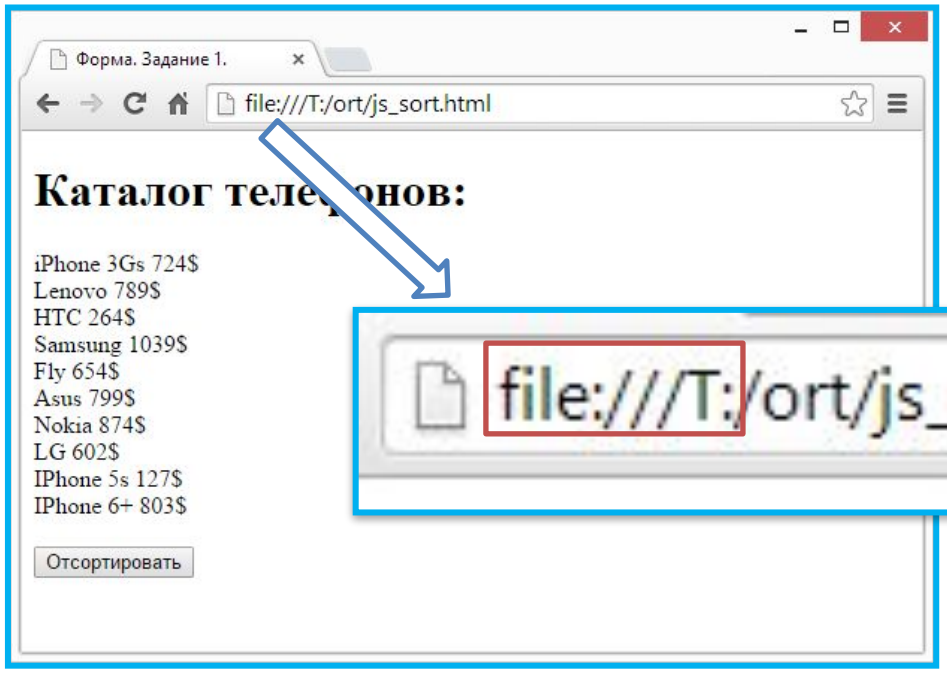
504 – Gateway Time-Out [The gateway or proxy has timed out.]

505 – HTTP Version Not Supported [The version of HTTP used by the client is not supported.]

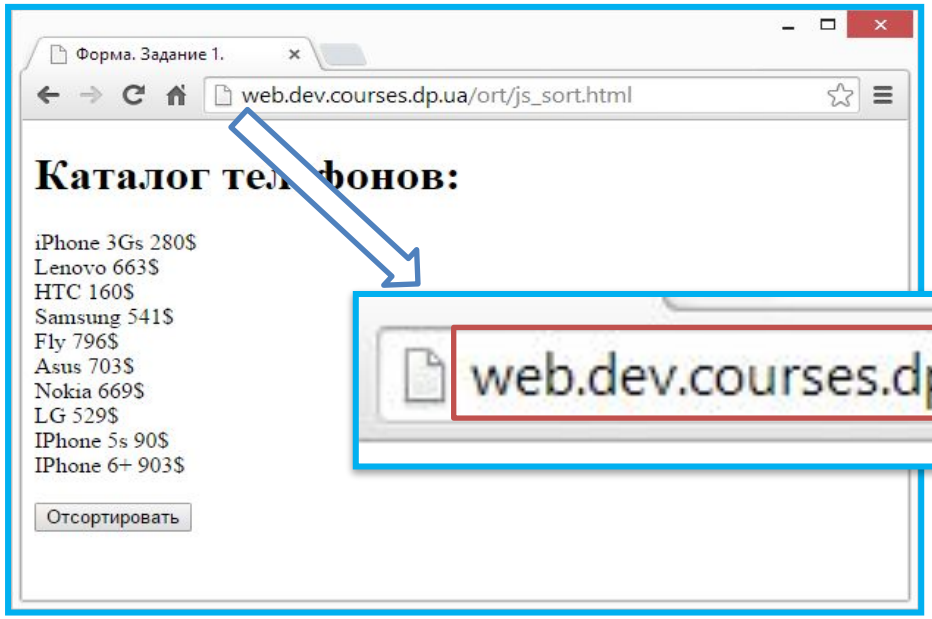
Коды состояний – ответ сервера о том, выполнен ли запрос, а если нет, то почему.

Подробнее:

https://ru.wikipedia.org/wiki/Список_кодов_состояния_HTTP



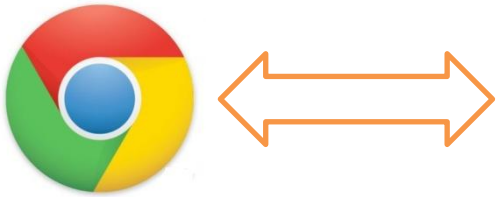
Локальный файл (на своём ПК)



Сетевой файл (на удалённом сервере)

Для браузера нет большой разницы неважно откуда получен файл

Web-сервер



Web-сервер – программа задача которой получать запросы от клиентов и возвращать им HTML-документ (и не только).

Подробнее: https://ru.wikipedia.org/wiki/Сравнение_веб-серверов

Open Server

Open Server *— это портативный локальный веб-сервер, имеющий многофункциональную управляющую программу и большой выбор подключаемых компонентов.*

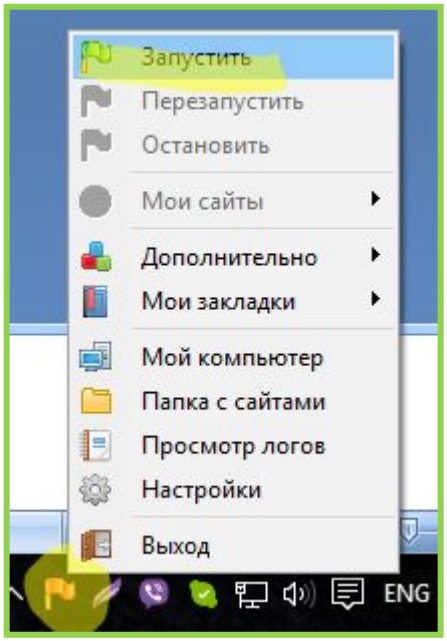
Скачать можно по адресу:

http://open-server.ru/download/

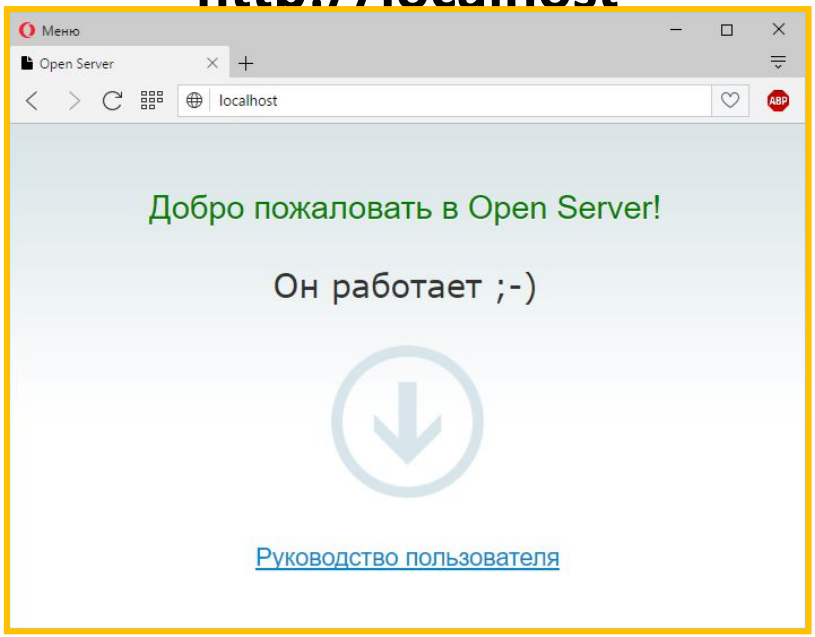
Open Server *– не единственный представитель программ типа «локальный сервер» для веб-разработчиков, подробнее:* http://habrahabr.ru/post/144242/

Open Server

Включите **Open Server**:

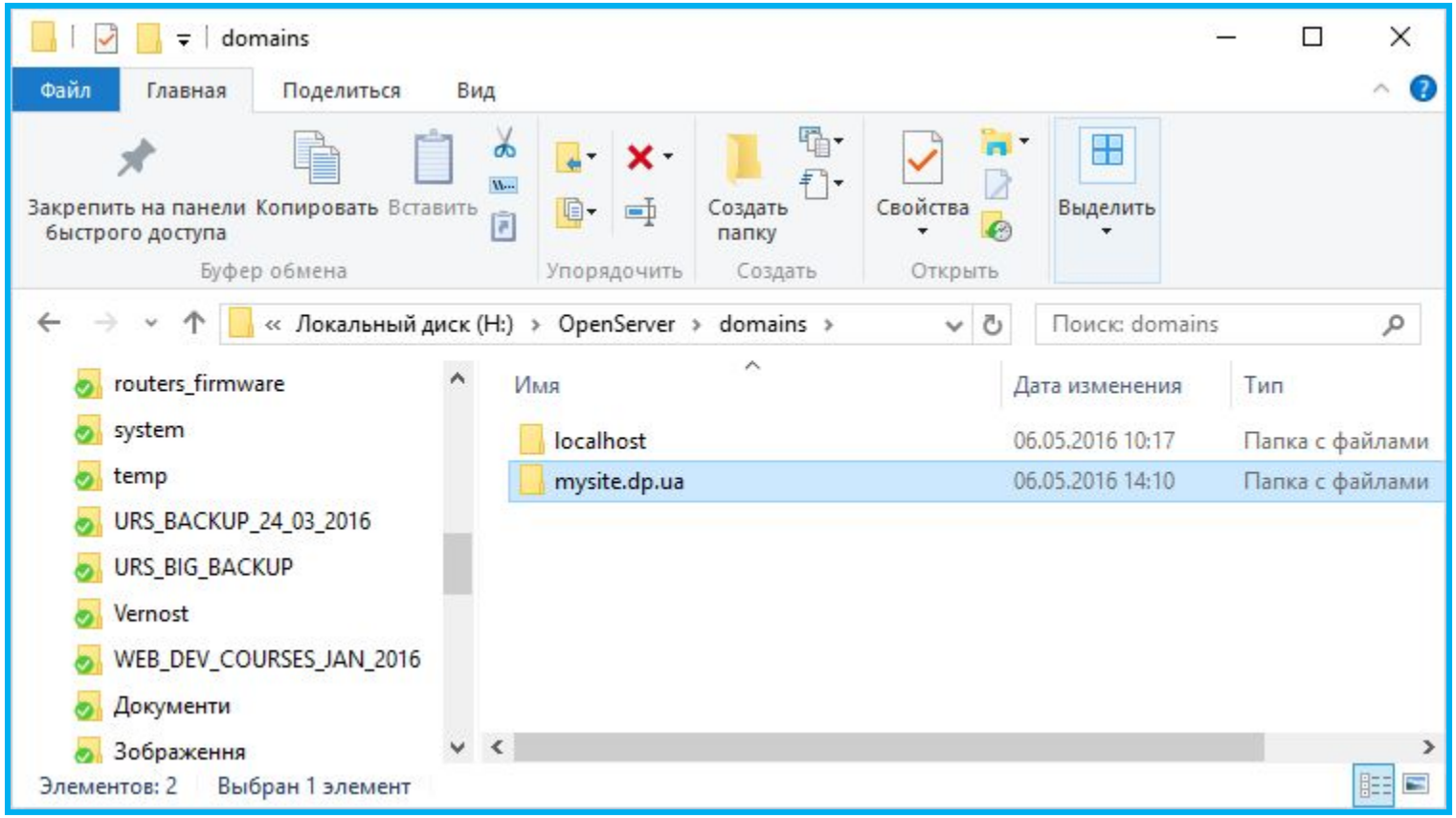


Введите в адресной строке браузера:
http://localhost



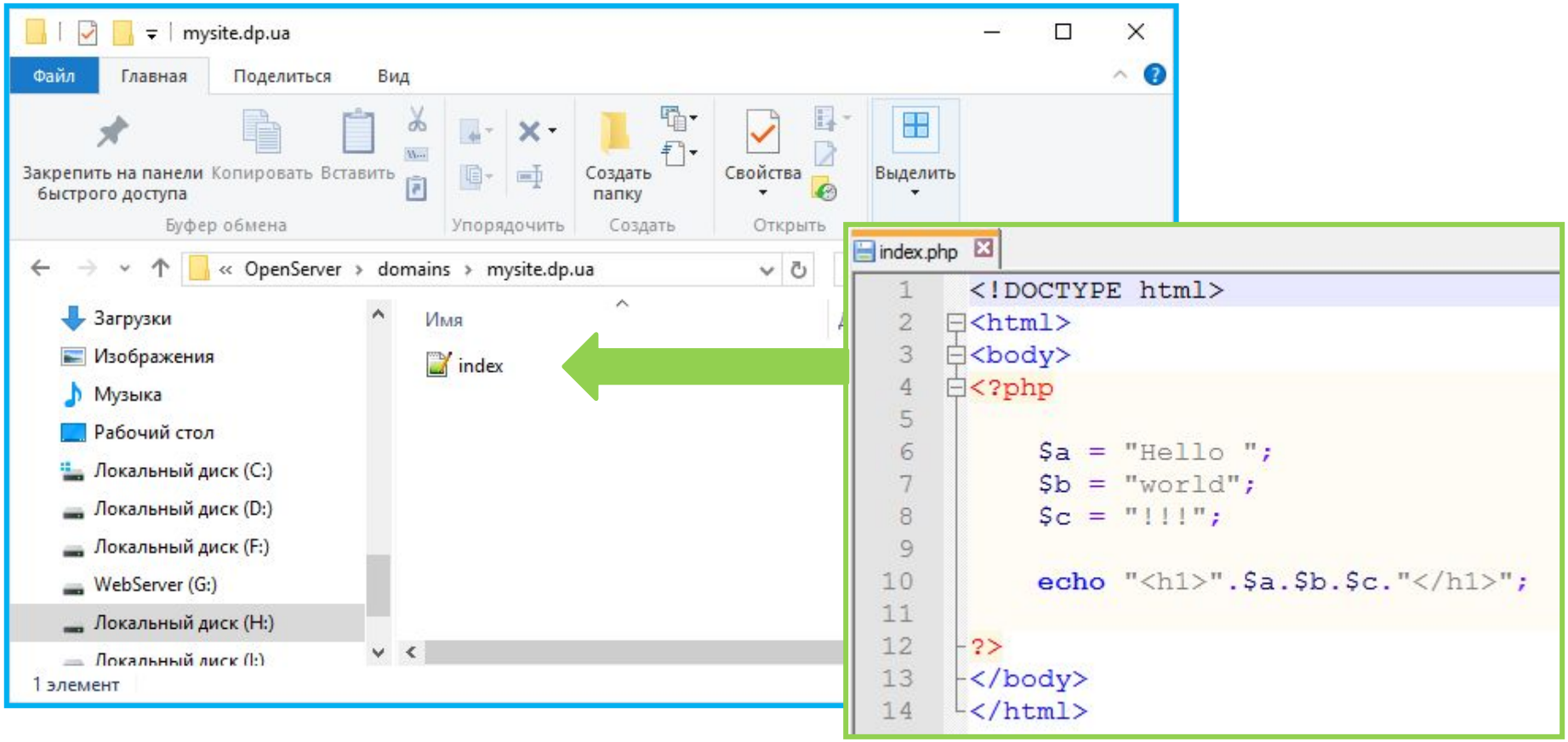
Признак успешной работы Open Server

Open Server



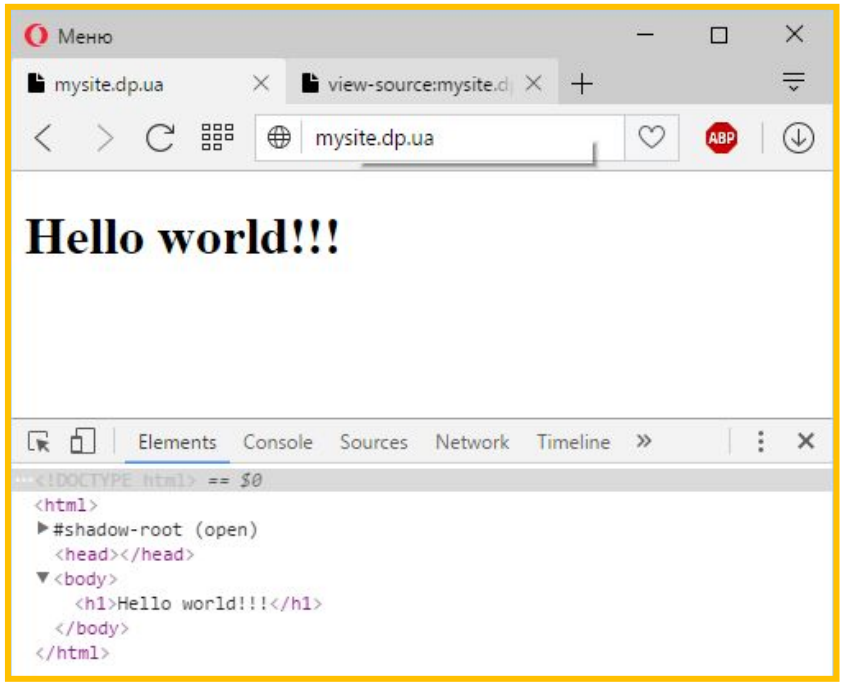
*Для «создания» сайта в **Open Server** откройте каталог в котором он установлен, а в нём каталог **domains**. В нём создайте каталог имя которого будет соответствовать имени (доменному) вашего создаваемого сайта. Например: **mysite.dp.ua***

Open Server

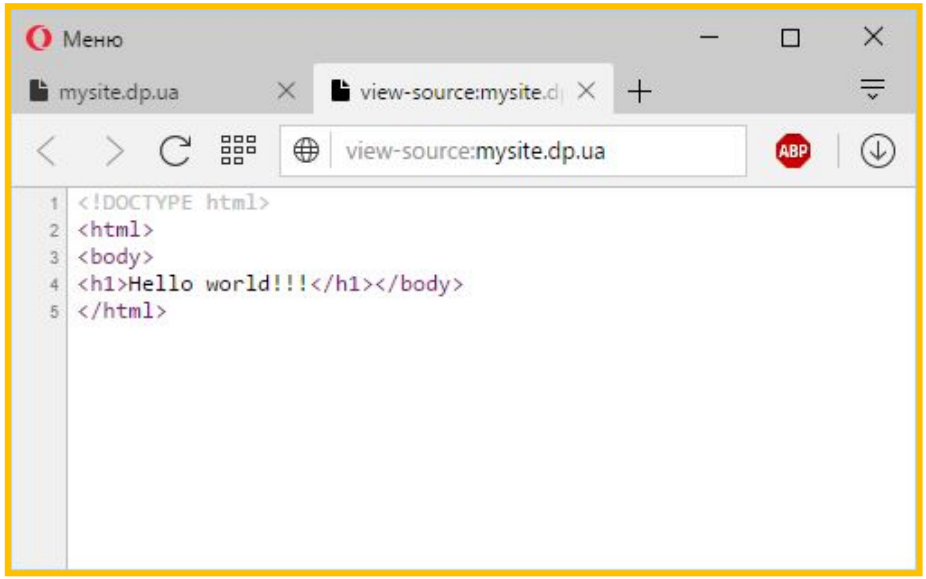


Создайте файл с именем index.php, напишите в нём приведенный код, и поместите его в каталог созданного сайта. После этого откройте сайт в браузере, зайдя на ***http://mysite.dp.ua***

Open Server



Результат



*Исходный код полученной страницы (**Ctrl+U**)*

*Создайте файл с именем **index.php**, напишите в нём приведенный код, и поместите его в каталог созданного сайта. После этого откройте сайт в браузере, зайдя на* ***http://mysite.dp.ua***

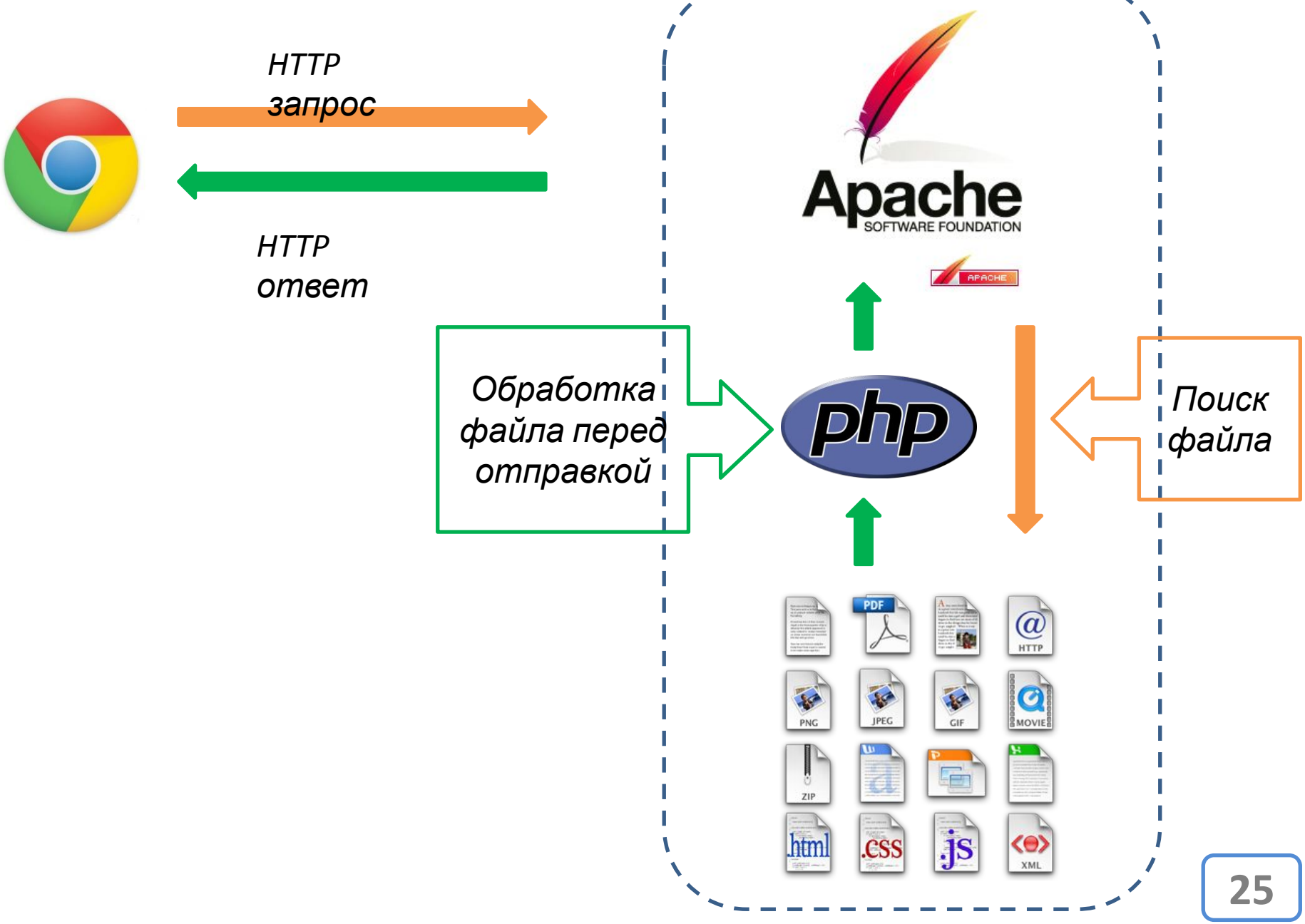
Веб-сервер

```php
<!DOCTYPE html>
<html>
<body>
<?php

    $a = "Hello ";
    $b = "world";
    $c = "!!!";

    echo "<h1>".$a.$b.$c."</h1>";

?>
</body>
</html>
```

???

```html
<!DOCTYPE html>
<html>
<body>
<h1>Hello world!!!</h1></body>
</html>
```

*В процессе передачи файла в страницу были внесены изменения, полностью пропал **PHP** код а вместо него появилась **HTML**-разметка.*

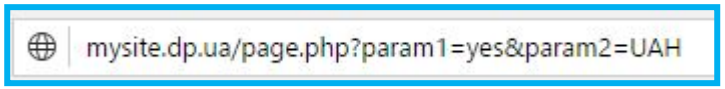

HTTP
запрос
HTTP
ответ
Apache
SOFTWARE FOUNDATION
APACHE
Обработка
файла перед
отправкой
php
Поиск
файла
PDF
HTTP
PNG
JPEG
GIF
MOVIE
ZIP
html
.css
.js
XML

Задача PHP - обработка (переработка, модификация) страницы перед отправкой её клиенту.

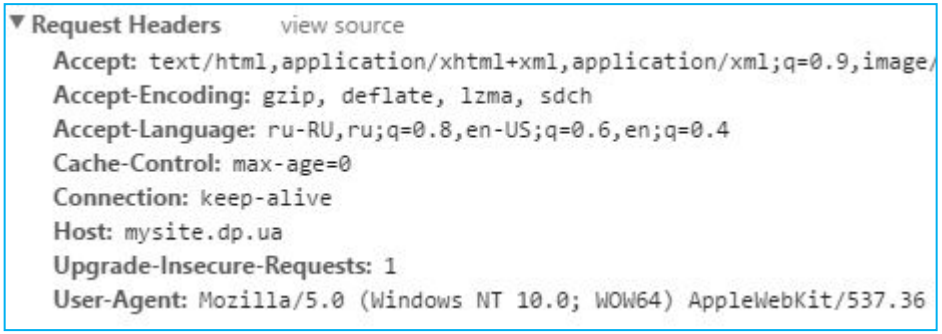
В зависимости от чего?

В зависимости от:

1. **Параметров переданных странице;**



2. **Данных переданных в заголовке HTTP запроса;**



3. **Данных о предыдущем посещении страницы.**

(cookie файлы, сессии)

Задача PHP - обработка (переработка, модификация) страницы перед отправкой её клиенту.

Как он это делает?

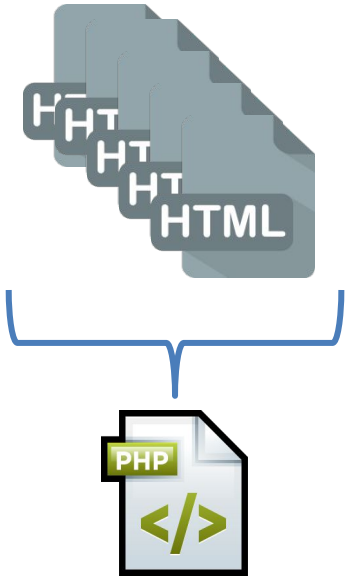
Выводя или не выводя те или иные фрагменты HTML-кода.

Задача PHP - обработка (переработка, модификация) страницы перед отправкой её клиенту.

А

Точка обмен данными

Обработка, хранение, защита и использование данных (и защита кода). ***Данные не хранятся у клиента!***

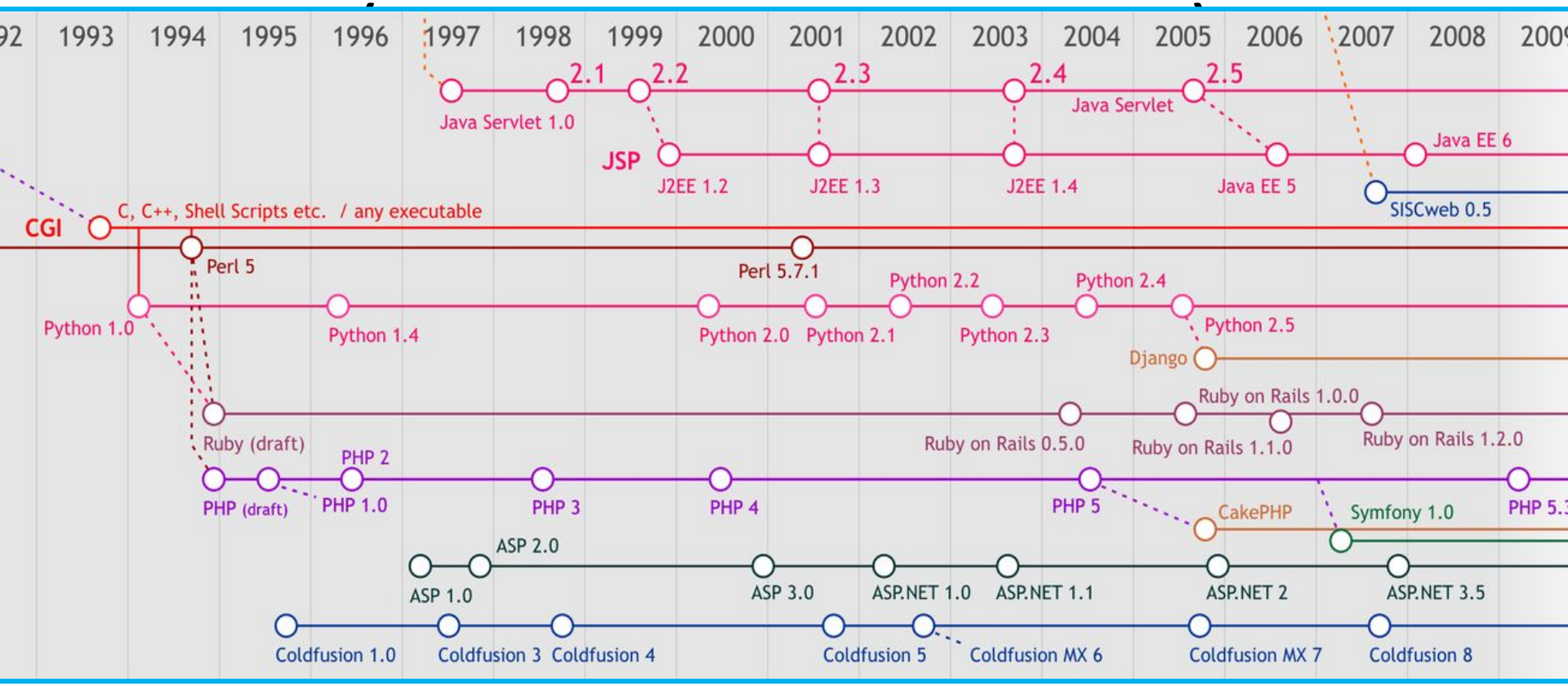


Меньше файлов, один файл может выводить различную информацию, в зависимости от ситуации.

Языки на серверной стороне

Подробнее: https://ru.wikipedia.org/wiki/Веб-программирование

Языки программирования



Подробнее: https://ru.wikipedia.org/wiki/Веб-программирование

PHP

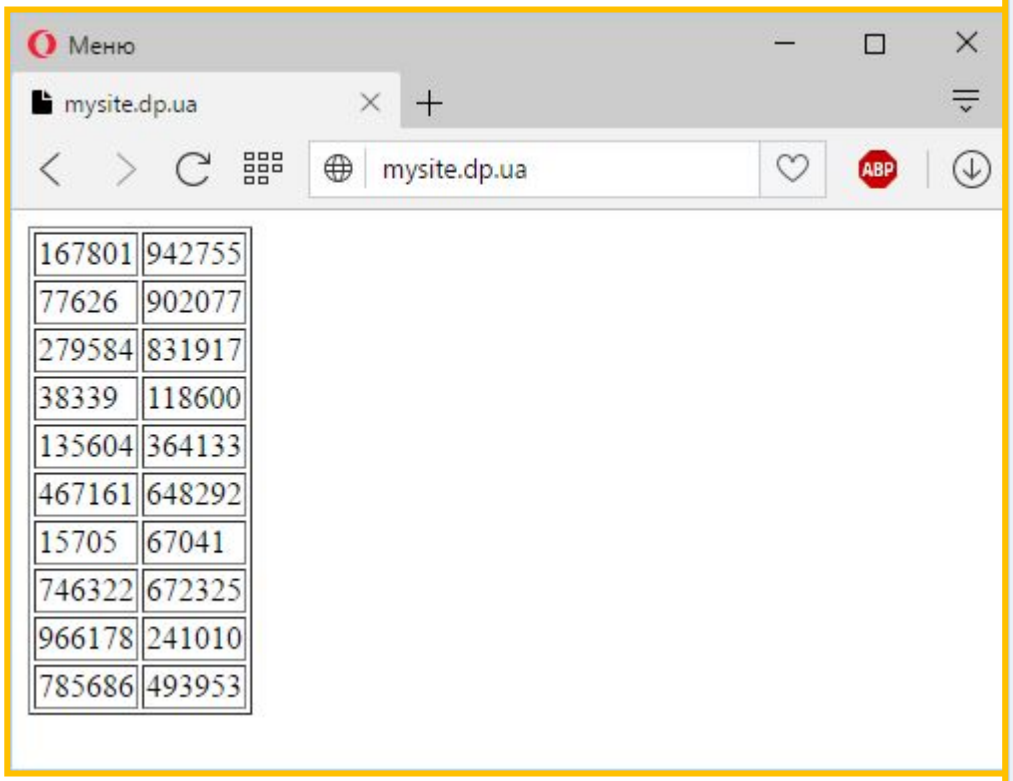
???

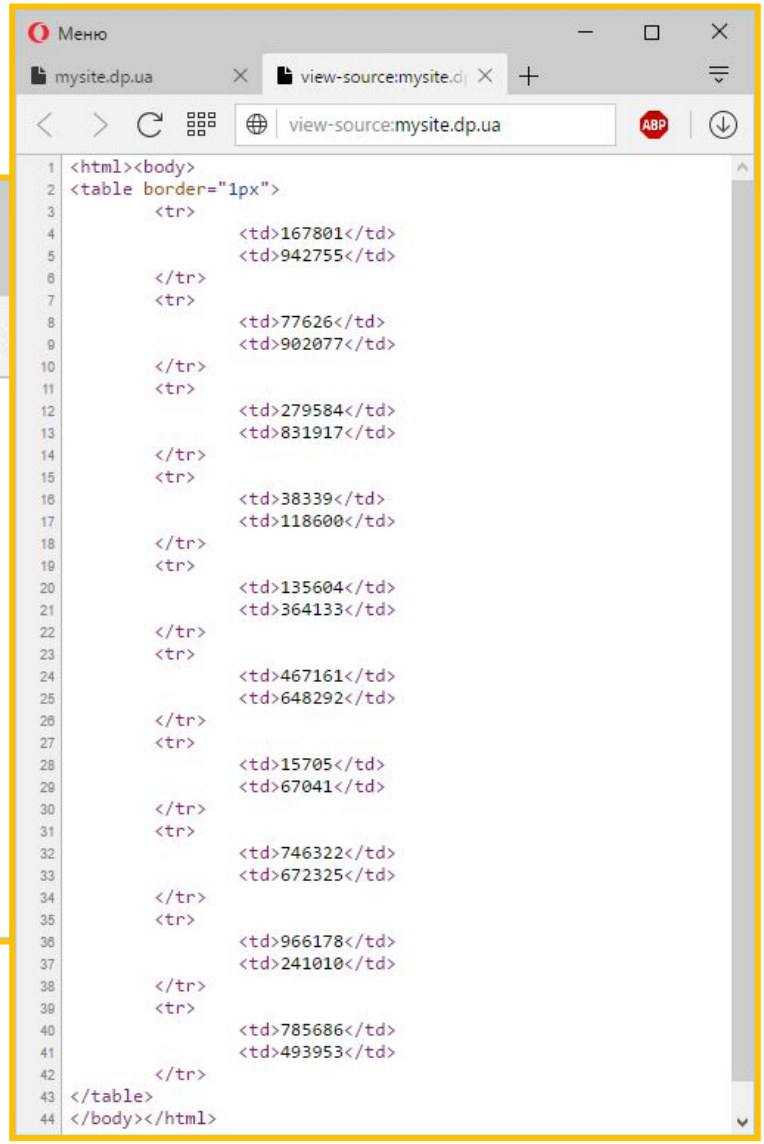
```
1  <html><body>
2  <table border="1px">
3  <?php
4
5      mt_srand(time());
6
7      for($i = 0; $i < 10; $i++){
8  ?>
9      <tr>
10         <td><?php echo mt_rand(1000, 999999); ?></td>
11         <td><?php echo mt_rand(1000, 999999); ?></td>
12     </tr>
13 <?php
14     }
15 ?>
16 </table>
17 </body></html>
```

PHP – средство генерации HTML-кода

*Поместите этот код в файл **page.php** (в каталоге вашего сайта), а затем откройте в браузере адрес: http://mysite.dp.ua/page.php*

PHP





***PHP** – средство генерации **HTML**-кода*

PHP

PHP в HTML

или

HTML в PHP

вот в чём вопрос…

PHP

```php
<!DOCTYPE html>
<html>
<body>
<p>Привет!!!</p>
<p>Ваш браузер: <?php echo $_SERVER["HTTP_USER_AGENT"]; ?></p>
<p>Ваш IP: <?php echo $_SERVER["REMOTE_ADDR"]; ?></p>
</body>
</html>
```

PHP в HTML

или

```php
<?php
    echo "<!DOCTYPE html>";
    echo "<html><body>";
    echo "<p>Привет!!!</p>";
    echo "<p>Ваш браузер: ".$_SERVER["HTTP_USER_AGENT"]."</p>";
    echo "<p>Ваш IP: ".$_SERVER["REMOTE_ADDR"]."</p>";
    echo "</body></html>";
?>
```

HTML в PHP

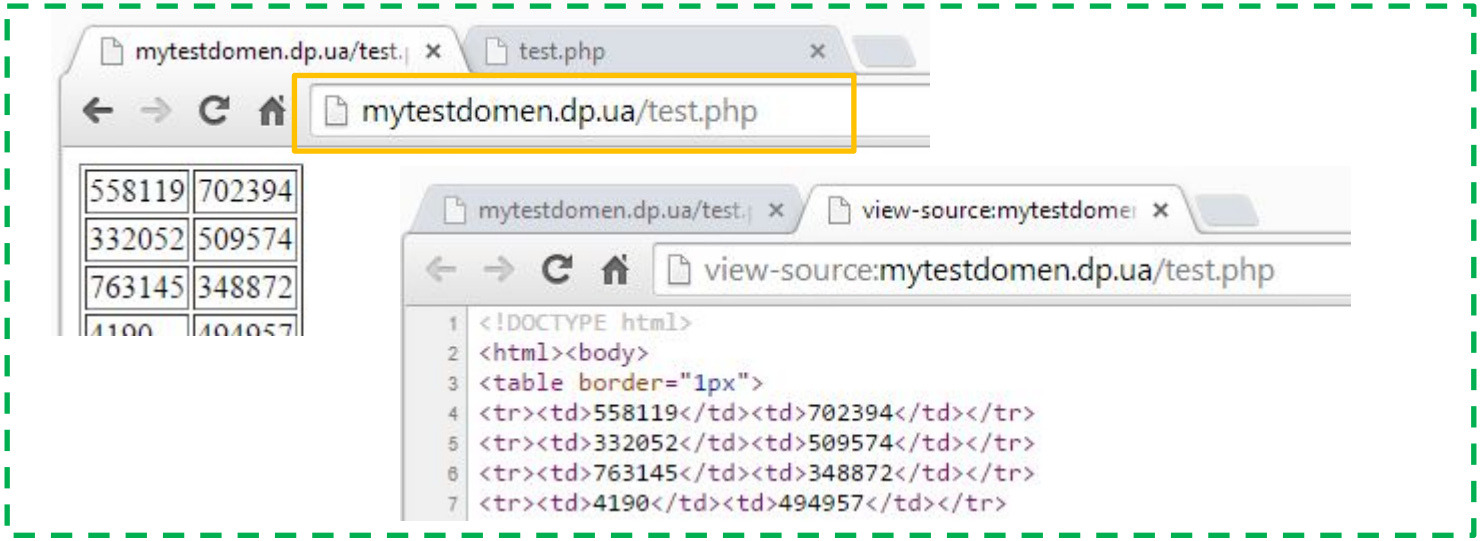
PHP

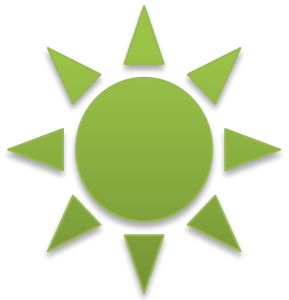
PHP в HTML == **HTML в PHP**

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4 <p>Привет!!!</p>
5 <p>Ваш браузер: Mozilla/5.0 (Windows NT 6.3;
  WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/39.0.2171.95 Safari/537.36</p>
6 <p>Ваш IP: 127.0.0.1</p>
7 </body>
8 </html>
```

Независимо от подхода, результатом будет HTML-документ.

PHP





PHP будет работать только если запрос к файлу пройдёт через web-сервер.

W3Schools PHP Tutorial

http://www.w3schools.com/php/

Книга

Научитесь создавать динамические Web-страницы с использованием PHP!
100 миллионов
PHP 5
для
"ЧАЙНИКОВ"™
Для сомневающихся
Исчерпывающее описание PHP 5 для Windows, Unix, Linux и Mac
Джанет Валейд
Автор книги
PHP & MySQL for Dummies
ДИАЛЕКТИКА