

# ОСНОВЫ WEB- КОНСТРУИРОВАНИЯ

# ИСТОРИЯ ИНТЕРНЕТА

Для начала мы хотим немного рассказать про интернет. Это поможет вам увидеть общую картину и понять, что и почему вы сейчас будете изучать.

История интернета начинается в 60-годы XX века. Впервые концепцию описал один американский учёный и назвал её «Галактическая сеть».

В 1969 американское агентство DARPA начало создавать экспериментальную сеть «с коммутацией пакетов». Её называли ARPANET.

*Коммутация пакетов - способ передачи данных по сети. Принцип работы очень простой: делим информацию на маленькие пакеты и отправляем их независимо друг от друга. Это нужно для надёжности, скорости и эффективности.*

В декабре 1970 года Network Working Group придумала протокол управления сетью, а в 1971 - 1972 его реализовали в ARPANET. Благодаря этому, появилась возможность создавать сетевые приложения. Первым приложением стала электронная почта, её сделали в 1972-м.

Но это всё научные исследования. Интернет, каким мы его знаем, придумал Тим Бернерс-Ли. Он изобрёл технологии URI/URL, HTTP, и HTML.

# URL

Первая важная технология, которая позволила появиться интернету - URL. Она применяется для обозначения адресов почти всех ресурсов Интернета: сайты и сервера приложений вроде электронной почты. URL сочетает в себе две технологии:

1. **URI** - стандарт записи уникального адреса. Например, адрес сайта это URI: <http://google.com>
2. **DNS** - система доменных имён. У любого устройства в сети есть свой числовой адрес, он называется **IP-адрес**. Но людям удобнее запоминать буквы - доменные имена. DNS просто помогает найти IP-адрес по доменному имени. Пример: 173.194.44.40 это IP-адрес, а google.com - доменное имя, которое ему соответствует.

# HTTP

**HyperText Transfer Protocol** (ГИПЕРТЕКСТ ТРАНСФЕР ПРОТОКОЛ) – ПРОТОКОЛ ПЕРЕДАЧИ ГИПЕРТЕКСТА. ОХ УЖ ЭТИ УЧЁНЫЕ, ЛЮБЯТ НАЗЫВАТЬ ПРОСТЫЕ ВЕЩИ СЛОЖНЫМИ ИМЕНАМИ. ДАВАЙТЕ РАЗБИРАТЬСЯ.

**HTTP** – ПРОТОКОЛ (ТО ЕСТЬ РЕГЛАМЕНТ, НАБОР УСЛОВНЫХ ОБОЗНАЧЕНИЙ), КОТОРЫЙ УСТАНАВЛИВАЕТ ОПРЕДЕЛЁННЫЙ ФОРМАТ ОБЩЕНИЯ МЕЖДУ КЛИЕНТОМ И СЕРВЕРОМ. СУТЬ ЕГО ЗАКЛЮЧАЕТСЯ В ТОМ, ЧТО КЛИЕНТ ПОСЫЛАЕТ ЗАПРОСЫ НА URL, А СЕРВЕР ЕМУ ОТВЕЧАЕТ.

**КЛИЕНТ** – ТОТ, КТО ПОСЫЛАЕТ ЗАПРОСЫ. КОГДА ВЫ ОТКРЫВАЕТЕ САЙТ, ВЫ – КЛИЕНТ.

**СЕРВЕР** – ТОТ, КТО ПРИНИМАЕТ ЗАПРОСЫ И ОТВЕЧАЕТ НА НИХ. КОГДА ВЫ ОТКРЫВАЕТЕ САЙТ ТОГО ЖЕ ГУГЛА, СЕРВЕР ГУГЛА ПОСЫЛАЕТ ВАМ ВЕБ-СТРАНИЧКУ – ОТВЕЧАЕТ НА ВАШ ЗАПРОС.

А ТЕПЕРЬ ПРО ЗАПРОСЫ. Что значит "послать запрос"? Если не углубляться в сложные научные определения, это означает отправить по сети сообщение с каким-нибудь требованием. Это работает так:

1. вы посылаете запрос
2. между вами и сервером устанавливается соединение
3. сервер формирует ответ, посылает его вам
4. соединение закрывается

Вот и всё.

Так вот, протокол передачи гипертекста. "Текста", потому что сервер отвечает на запрос текстом определённого формата. А так как любые данные можно представить в виде текста, то получается, что через HTTP можно пересылать всё: картинки с котиками, гифки с котиками, и даже видео с котиками.

# HTML

**HyperText Markup Language** (ГИПЕРТЕКСТ МАРКАП ЛЭНГУИДЖ) - ЯЗЫК РАЗМЕТКИ ГИПЕРТЕКСТА. ОПЯТЬ РАЗБИРАЕМСЯ В СЛОЖНЫХ ТЕРМИНАХ. В ЭТОЙ АББРЕВИАТУРЕ НАМ ИНТЕРЕСНО СЛОВО "РАЗМЕТКА".

РАЗМЕТКА - ЧТО ЭТО ВООБЩЕ ТАКОЕ? ПРЕДСТАВЬТЕ, ЧТО ВЫ ПЕРЕДАЁТЕ ТЕКСТ ПО СЕТИ. КАК СДЕЛАТЬ В ТЕКСТЕ ЗАГОЛОВОК? ВЫДЕЛИТЬ АБЗАЦ? ПОДЧЕРКНУТЬ СЛОВО? САМЫЙ ПРОСТОЙ ВАРИАНТ - ПОМЕТИТЬ НАЧАЛО И КОНЕЦ ВЫДЕЛЯЕМОГО ФРАГМЕНТА УСЛОВНЫМИ МЕТКАМИ. НАПРИМЕР:

<заголовок>HTML</заголовок><полужирный>HyperText  
Markup Language</полужирный> <курсив>(ГИПЕРТЕКСТ МАРКАП  
ЛЭНГУИДЖ)</курсив> - ЯЗЫК РАЗМЕТКИ ГИПЕРТЕКСТА.

HTML позволил создавать документы, в которых были заголовки, абзацы, ссылки, таблицы, картинки и куча всего прочего.

ТЕПЕРЬ МЫ ЗНАЕМ, ЧТО ТАКОЕ РАЗМЕТКА, ДАВАЙТЕ УЗНАЕМ, ЧТО ТАКОЕ HTML. ЕСТЬ ВЕЩИ, КОТОРЫЕ ПРОЩЕ ПОКАЗАТЬ, ЧЕМ ОБЪЯСНИТЬ. HTML - ОДНА ИЗ НИХ.

ВСЁ ГЕНИАЛЬНОЕ ПРОСТО. ВОТ КАК БЫ ВЫГЛЯДЕЛ ТЕКСТ ЭТОГО СЛАЙДА НА HTML:

```
<h1>HTML</h1><p>ТЕПЕРЬ МЫ ЗНАЕМ, ЧТО ТАКОЕ <i>РАЗМЕТКА</i>, ДАВАЙ  
УЗНАЕМ, ЧТО ТАКОЕ HTML. ЕСТЬ ВЕЩИ, КОТОРЫЕ ПРОЩЕ ПОКАЗАТЬ, ЧЕМ ОБЪЯСНЯТЬ.  
HTML - ОДНА ИЗ НИХ.</p> <p>ВСЁ ГЕНИАЛЬНОЕ ПРОСТО. ВОТ КАК БЫ ВЫГЛЯДЕЛ  
ТЕКСТ ЭТОГО СЛАЙДА НА HTML:</p>
```

**h1** - ЗАГОЛОВОК ПЕРВОГО УРОВНЯ (HEADER 1)

**p** - АБЗАЦ (PARAGRAPH)

# ТЕГИ

## ЧТО ТАКОЕ ТЕГ

Тег — это синтаксическая единица языка HTML, которая выделяет или создаёт элемент. Это набор символов, с помощью которого браузер понимает, где элемент создается, начинается и заканчивается. Есть 2 вида тегов: двойные и одинарные.

## Двойные теги

Двойные теги показывают начало и конец элемента. Начало элемента обозначается открывающим тегом `<...>`, а конец — закрывающим `</...>`.

**Двойной тег обязательно должен быть закрыт.** Даже несмотря на то, что современные браузеры умеют в некоторых случаях понимать разметку без закрытых тегов, лучше всегда закрывать их.



## ОДИНАРНЫЕ ТЕГИ

ОДИНАРНЫЕ ТЕГИ ПРОСТО НЕ ИМЕЮТ ПАРЫ. ПРИМЕРЫ: ТЕГ ПЕРЕНОСА СТРОКИ `<BR>` ИЛИ ГОРИЗОНТАЛЬНОЙ ЛИНИИ `<HR>`.

СТАРЫЕ БРАУЗЕРЫ ТРЕБОВАЛИ ЗАКРЫВАТЬ ОДИНАРНЫЕ ТЕГИ: `<BR />`, СЕЙЧАС ТАКИХ БРАУЗЕРОВ ПРАКТИЧЕСКИ НЕ ОСТАЛОСЬ И ДОПУСТИМО ИСПОЛЬЗОВАТЬ ОБА ВАРИАНТА СИНТАКСИСА.

## ВЛОЖЕННОСТЬ ТЕГОВ

ТЕГИ МОЖНО ВКЛАДЫВАТЬ ДРУГ В ДРУГА.

ПРИМЕР:

```
<P><EM>КУРСИВ ВНУТРИ АБЗАЦА</EM></P>
```

НО ПРИ ВЛОЖЕНИИ ТЕГА НУЖНО ВСЕГДА ПОМНИТЬ О ТОМ, ЧТО **ВНУТРЕННИЙ ТЕГ НУЖНО ЗАКРЫВАТЬ РАНЬШЕ ВНЕШНЕГО**. ТО ЕСТЬ ВОТ ТАКОЙ КОД НЕДОПУСТИМ:

```
<P><EM>Я УЖЕ ГОВОРИЛ ТЕБЕ, ЧТО ТАКОЕ БЕЗУМИЕ?</P></EM>
```

# АТРИБУТЫ

- АТРИБУТЫ — ЭТО СВОЙСТВА ТЕГА. С ПОМОЩЬЮ НИХ МЫ ЗАДАЁМ ПАРАМЕТРЫ ТЕГА.

СРАЗУ ВОЗЬМЁМ ПРИМЕР: ТЕГ <A> — ССЫЛКА. ДЛЯ ЗАДАНИЯ АДРЕСА, КУДА БУДЕТ ВЕСТИ ЭТА ССЫЛКА, НАМ ПОНАДОБИТСЯ АТРИБУТ HREF. ВОТ ТАК БУДЕТ ВЫГЛЯДЕТЬ ССЫЛКА НА СТРАНИЦУ ИТС ВКОНТАКТЕ:

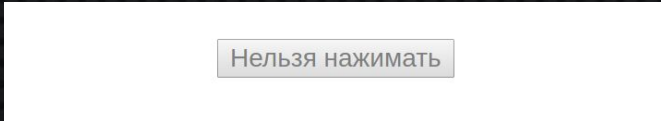
- <A HREF="HTTPS://VK.COM/ITS.DIGITAL">ИТС ВКОНТАКТЕ</A>

АТТРИБУТ УКАЗЫВАЕТСЯ ВНУТРИ ТЕГА, ЗНАЧЕНИЕ АТТРИБУТА УКАЗЫВАЕТСЯ ВНУТРИ КАВЫЧЕК.  
АТТРИБУТЫ ОТДЕЛЯЮТСЯ ДРУГ ОТ ДРУГА ПРОБЕЛАМИ. ПРИМЕР ССЫЛКИ НА СТРАНИЦУ ИТС,  
КОТОРАЯ ОТКРОЕТСЯ В НОВОЙ ВКЛАДКЕ:

```
<A href="https://vk.com/its.digital" target="_blank">ИТС Вконтакте</A>
```

У АТТРИБУТА МОЖЕТ НЕ БЫТЬ ЗНАЧЕНИЯ, ТОГДА НАЛИЧИЕ АТТРИБУТА ВКЛЮЧАЕТ КАКОЙ-ТО ПАРАМЕТР, А ОТСУТСТВИЕ - ОТКЛЮЧАЕТ. НАПРИМЕР, АТТРИБУТ DISABLED. ЕСЛИ КНОПКЕ <BUTTON> ЗАДАТЬ АТТРИБУТ DISABLED, ОНА СТАНЕТ СЕРОЙ И НА НЕЁ НЕВОЗМОЖНО БУДЕТ НАЖАТЬ.

```
<BUTTON DISABLED>Нельзя нажимать</BUTTON> РЕЗУЛЬТАТ:
```



Нельзя нажимать

# ОСОБЕННОСТИ ИНТЕРПРЕТАЦИИ HTML

## **Перенос строки только через тег**

Возможно, у вас возник вопрос, зачем нужен тег переноса строки, если можно просто нажать `ENTER`. Дело в том, что HTML воспринимает перенос строки как пробел. Это нужно потому, что редакторы кода не переносят строки, которые не помещаются в экран - так удобнее писать код. Поэтому чтобы длинный текст влезал в экран, в коде ставятся переносы строки, которые не нужны, когда страница показывается в браузере.

## **Несколько пробелов, идущих подряд, считаются за один**

Так просто удобнее форматировать код в редакторе. Из-за того, что теги вкладываются друг в друга, для удобного восприятия кода вложенность показывают отступами - пробелами. Пример:

```
<article>
```

```
<h1>В Индонезии после землетрясения началось извержение вулкана</h1>
```

```
<p>
```

```
В Индонезии на острове  
Сулавеси проснулся вулкан Сопутан,  
выпустив столб пепла высотой около четырех метров, пишут  
«Новые Известия». Извержение вулкана началось сегодня утром.  
Местным жителям рекомендовали не приближаться к нему и  
использовать респираторы.
```

```
</p>
```

```
<p>
```

```
Подземные толчки стали причиной возникновения цунами,  
высота волн достигала 7 м. По последним данным, в результате  
стихийных бедствий погибли более 1400 человек. На оказание  
помощи пострадавшим от землетрясения в Индонезии ООН выделила  
$15 млн.
```

```
</p>
```

```
</article>
```

## Произвольный регистр

<BR> даст такой же результат, что и <BR>, и <Br>, и <br>. **Несмотря на это, писать разметку лучше в нижнем регистре - это негласное правило.**

## Перенос строки в теге

При определении тега и его атрибутов можно переносить строку. Это полезно для длинных определений.

Например, для этого изображения:

```
<IMG  
SRC="HTTP://EXAMPLE.COM/CAT.JPG"  
TITLE="Мурка"  
ALT="Рыжая кошка валяется в снегу"  
WIDTH="640"  
HEIGHT="480">
```





# <!DOCTYPE>

Первым тегом в любом HTML документе должен идти тег <!DOCTYPE>. Он говорит браузеру, по какому стандарту написана страница. На заре веба HTML существовал в разных несовместимых версиях, поэтому для их одновременной поддержки нужно было указывать версию. Сейчас все пришли к одному стандарту - HTML5. Поэтому для всех сайтов, которые создаются сегодня, нужно указывать <!DOCTYPE html> - так обозначается HTML5.



## <HEAD>

В ТЕГЕ <HEAD> ХРАНИТСЯ ИНФОРМАЦИЯ О СТРАНИЦЕ. ЗДЕСЬ

УКАЗЫВАЮТ КОДИРОВКУ <META

CHARSET="...">, ИМЯ

СТРАНИЦЫ <TITLE>...</TITLE>, СПЕЦИАЛЬНУЮ

ИНФОРМАЦИЮ ДЛЯ ПОИСКОВИКОВ, А ЕЩЁ ТУТ

ПОДКЛЮЧАЮТСЯ СТИЛЕВЫЕ ФАЙЛЫ И СКРИПТЫ.

ТЕГ <HEAD> НЕ ОТОБРАЖАЕТСЯ. ЕГО ЦЕЛЬ —

СКАЗАТЬ БРАУЗЕРУ ИНФОРМАЦИЮ О СТРАНИЦЕ.



# ПРАКТИКА: СОЗДАНИЕ ВЕБ-СТРАНИЦЫ

```
<!DOCTYPE html>

<html>
  <head>
    <meta charset="utf-8">
  </head>

  <body>
    <h1>Заголовок</h1>
    <p>Абзац</p>
  </body>
</html>
```

СОХРАНИТЕ ФАЙЛ. ПРИ СОХРАНЕНИИ, УКАЖИТЕ КОДИРОВКУ UTF-8. ОТКРОЙТЕ ФАЙЛ В БРАУЗЕРЕ. ВЫ УВИДИТЕ ВОТ ЭТО:

**Заголовок**

Абзац

СПАСИБО  
ЗА ВНИМАНИЕ!

Презентацию подготовил: ученик 11 класса,  
Никитенко Александр