

`XML. HTML`




# Что такое XML?

- **XML** (*eXtensible Markup Language*) — расширяемый язык разметки.
- Его главная составная часть — теги. Пример:  
`<book></book>`
- Теги бывают открывающими и закрывающими. У закрывающего есть дополнительный символ — “/”, это видно на примере выше. Каждому открывающему тегу должен соответствовать закрывающий. Они показывают, где начинается и где заканчивается описание каждого элемента в файле.



# Преимущества XML для описания данных

- 1. Легко читается и человеком, и компьютером.
  - 2. Поскольку данные хранятся в простом текстовом формате, при их передаче с одного компьютера на другой не возникнет никаких проблем с совместимостью.
- 

# Пример XML-файла

```
<?xml version="1.0" encoding="UTF-8"?>
<carstore>
  <car category="truck">
    <model lang="en">Scania R 770</model>
    <year>2005</year>
    <price currency="US dollar">200000.00</price>
  </car>
  <car category="sedan">
    <title lang="en">Ford Focus</title>
    <year>2012</year>
    <price currency="US dollar">20000.00</price>
  </car>
  <car category="sport">
    <title lang="en">Ferrari 360 Spider</title>
    <year>2018</year>
    <price currency="US dollar">150000.00</price>
  </car>
</carstore>
```

# Префиксы и пространства имен

```
<?xml version="1.0" encoding="UTF-8"?>
<carstore xmlns:real="http://testproject.developersgroup1.companyname/department2/namespaces/real"
  xmlns:toy="http://testproject.developersgroup1.companyname/department2/namespaces/toy">
  <real:car category="truck">
    <model lang="en">Scania R 770</model>
    <year>2005</year>
    <price currency="US dollar">200000.00</price>
  </real:car>
  <toy:car category="sedan">
    <title lang="en">Ford Focus</title>
    <year>2012</year>
    <price currency="US dollar">100.00</price>
  </toy:car>
</carstore>
```



# XML для описания данных

## Достоинства:

- легкость чтения, подача в простой форме;
- стандартный вид кодировки;
- возможность создания разных структур (списков, схем, деревьев);
- популярность в разных сферах программирования.

## Недостатки:

- Чрезмерный синтаксис, большое количество сущностей и тегов;
- отсутствуют стандартные указания типа объекта.




# HTML



- **HTML** (*HyperText Markup Language* — «язык гипертекстовой разметки») — стандартизированный язык разметки документов.
- Большинство веб-страниц содержат описание разметки на языке HTML (или XHTML). Язык HTML интерпретируется браузерами; полученный в результате интерпретации форматированный текст отображается на экране монитора компьютера или мобильного устройства.
- Язык XHTML является более строгим вариантом HTML, он следует синтаксису XML и является приложением языка XML в области разметки гипертекста.



# Пример HTML-документа




```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Тестовая страница</title>
  </head>
  <body>
    <p>Это — моя страница</p>
  </body>
</html>
```





# Объявление типа документа



```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

# Специальные символы

Буквенный символ	Символьный эквивалент
<	&lt;
>	&gt;
"	&quot;
'	&apos;
&	&amp;

Чтобы превратить часть содержимого HTML-файла в комментарий, нужно поместить её в специальные маркеры <!-- и -->



# Meta

1. name — тип элемента, то есть какие именно метаданные он содержит.

2. content — сами метаданные.

Два полезных элемента метаданных — указание автора страницы и краткое описание её содержимого. Рассмотрим эти элементы на примере:

```
<meta name="author" content="CC">
```

```
<meta name="description" content="Пример">
```

В сети существует множество других типов метаданных. Многие из них — это собственные форматы, созданные для предоставления определенным сайтам (например, социальных сетей) специальной информации, которую они могут использовать.

# Иконки сайта

```
<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">
```

Для разных устройств можно указывать разные иконки. Например, на главной странице MDN:

```
<!-- Для iPad 3 с Retina-экраном высокого разрешения: -->
```

```
<link rel="apple-touch-icon-precomposed" sizes="144x144"  
href="https://developer.cdn.mozilla.net/static/img/favicon144.a6e4162070f4.png">
```

```
<!-- Для iPhone с Retina-экраном высокого разрешения: -->
```

```
<link rel="apple-touch-icon-precomposed" sizes="114x114"  
href="https://developer.cdn.mozilla.net/static/img/favicon114.0e9fabd44f85.png">
```

```
<!-- Для iPad первого и второго поколения: -->
```

```
<link rel="apple-touch-icon-precomposed" sizes="72x72"  
href="https://developer.cdn.mozilla.net/static/img/favicon72.8ff9d87c82a0.png">
```

```
<!-- Для iPhone, iPod Touch без Retina и устройств с Android 2.1+: -->
```

```
<link rel="apple-touch-icon-precomposed"  
href="https://developer.cdn.mozilla.net/static/img/favicon57.a2490b9a2d76.png">
```

```
<!-- Для других случаев - обычный favicon -->
```

```
<link rel="shortcut icon"  
href="https://developer.cdn.mozilla.net/static/img/favicon32.e02854fdcf73.png">
```

# Основной язык страницы

```
<html lang="en-US">  
<html lang="ru">
```

Это полезно во многих случаях. Ваш HTML-документ будет более эффективно индексироваться поисковыми системами, если его язык установлен (что позволяет ему правильно отображаться в языковых результатах), и он полезен людям с нарушением зрения, которые используют программы, читающие страницы вслух (например, слово "шесть" пишется одинаково как на французском, так и на английском языках, но произносится по-разному.).

Можно также указать язык для части документа. Например, мы могли бы установить язык для части страницы на японском:

```
<p>Пример на японском: <span lang="jp"> ご飯が熱い。 </span>.</p>
```



# Форматирование текста

Большинство структурированных текстов состоят из параграфов и заголовков, независимо от того, читаете ли вы рассказ, или газету, или учебник, журнал и т.д.

Упорядоченный контент делает чтение более легким и приятным.

В HTML каждый абзац заключен в элемент `<p>`, подобно:

```
<p>Я параграф, да, это я.</p>
```

Каждый заголовок заключен в элемент заголовка `<h1>`:

```
<h1>Я заголовок истории.</h1>
```

Имеется шесть элементов заголовка: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` и `<h6>`. Каждый элемент представляет разный уровень контента в документе; `<h1>` представляет главный заголовок, `<h2>` представляет подзаголовки, `<h3>` представляет под-подзаголовки и так далее.



# СПИСКИ

## □ Неупорядоченные

<ul>

<li>молоко</li>

<li>яйца</li>

<li>хлеб</li>

<li>хумус</li>

</ul>

## □ Упорядоченные

<ol>

<li>Доедьте до конца дороги</li>

<li>Поверните направо</li>

<li>Едьте прямо через первые два перекрестка с круговым движением</li>

<li>Поверните налево на третьем перекрестке</li>

<li>Школа справа от вас, в 300 метрах вверх по дороге</li>

</ol>



# Акцент

Для курсивного текста в HTML используется элемент `<em>` (выделение). Кроме того, чтобы сделать документ более интересным для чтения, они распознаются программами, считывающими с экрана, и произносятся другим тоном.

`<p>Я <em>рад</em>, что ты не <em>опоздал</em>.</p>`

Для полужирного текста в HTML используется элемент `<strong>` (важное значение). Помимо того, что документ становится более полезным, они распознаются программами, считывающими с экрана, и говорят другим тоном.

`<p>Эта жидкость <strong>очень токсична</strong>.</p><p>Я рассчитываю на тебя. <strong>Не </strong>опаздывай!</p>`




# Изображения на странице

- ``
- Если графический файл находится в одной папке с HTML-документом, то в качестве пути достаточно указать только имя файла

# Таблицы

- Таблица состоит из строк и столбцов ячеек, которые могут содержать текст и рисунки. Для добавления таблицы на веб-страницу используется тег `<table>`. Этот элемент служит контейнером для элементов, определяющих содержимое таблицы. Любая таблица состоит из строк и ячеек, которые задаются соответственно с помощью тегов `<tr>` и `<td>`. Таблица должна содержать хотя бы одну ячейку. Допускается вместо тега `<td>` использовать тег `<th>`. Текст в ячейке, оформленной с помощью тега `<th>`, отображается браузером шрифтом жирного начертания и выравнивается по центру ячейки. В остальном, различия между ячейками, созданными через теги `<td>` и `<th>` нет.



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ter table</title>
  </head>
  <body>
    <table border="1">
      <tr>
        <th>Ячейка 1</th>
        <th>Ячейка 2</th>
      </tr>
      <tr>
        <td>Ячейка 3</td>
        <td>Ячейка 4</td>
      </tr>
    </table>
  </body>
</html>
```

# Объединение ячеек

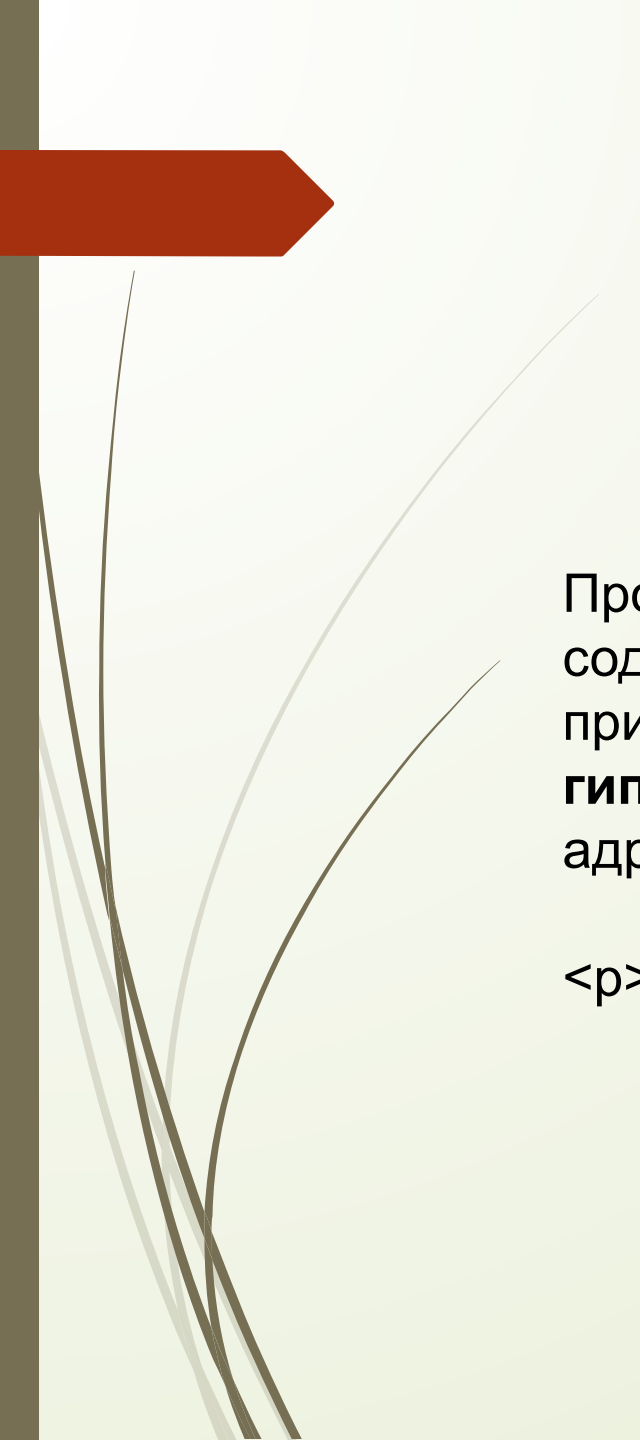
- Для объединения двух и более ячеек в одну используются атрибуты `colspan` и `rowspan` тега `<td>`. Атрибут `colspan` устанавливает число ячеек объединяемых по горизонтали. Аналогично работает и атрибут `rowspan`, с тем лишь отличием, что объединяет ячейки по вертикали. Перед добавлением атрибутов проверьте число ячеек в каждой строке, чтобы не возникло ошибок. Так, `<td colspan="3">` заменяет три ячейки, поэтому в следующей строке должно быть три тега `<td>` или конструкция вида `<td colspan="2">...</td><td>...</td>`. Если число ячеек в каждой строке не будет совпадать, появятся пустые фантомные ячейки.



# Гиперссылки



- Гиперссылки позволяют нам связывать документы с любым другим документом (или ресурсом), с которым необходимо.
- URL-адрес может указывать на файлы HTML, текстовые файлы, изображения, текстовые документы, видео и аудиофайлы и все остальное. Если веб-браузер не знает, как отображать или обрабатывать файл, он спросит, хотите ли вы открыть файл (в этом случае обязанность открытия или обработки файла передаётся в соответствующее локальное приложение на устройстве) или загрузить файл (в этом случае вы можете попытаться разобраться с ним позже).



Простая ссылка создаётся путём обёртывания текста (или другого содержимого), который вы хотите превратить в ссылку, в элемент `<a>`, и придания этому элементу атрибута `href` (который также известен как **гипертекстовая ссылка**, или **цель**), который будет содержать веб-адрес, на который вы хотите указать ссылку.

```
<p>Ссылка на <a href="http://bmstu-kaluga.ru"
    title="Наш родной университет">сайт университета </a>.</p>
```

# Ссылки на части документа

Можно ссылаться на определенную часть документа HTML (известную как **фрагмент документа**), а не только на верхнюю часть документа. Для этого вам сначала нужно назначить атрибут `id` элементу, с которым вы хотите связаться. Обычно имеет смысл ссылаться на определённый заголовок, поэтому это выглядит примерно так:

```
<h2 id="mail">Почтовый адрес</h2>
```

Затем, чтобы связаться с этим конкретным `id`, вы должны включить его в конец URL-адреса, которому предшествует знак решётки, например:

```
<p>Хотите написать мне письмо? Используйте наш <a  
href="contacts.html#mail">почтовый адрес</a>.</p>
```

# Сохранение файлов

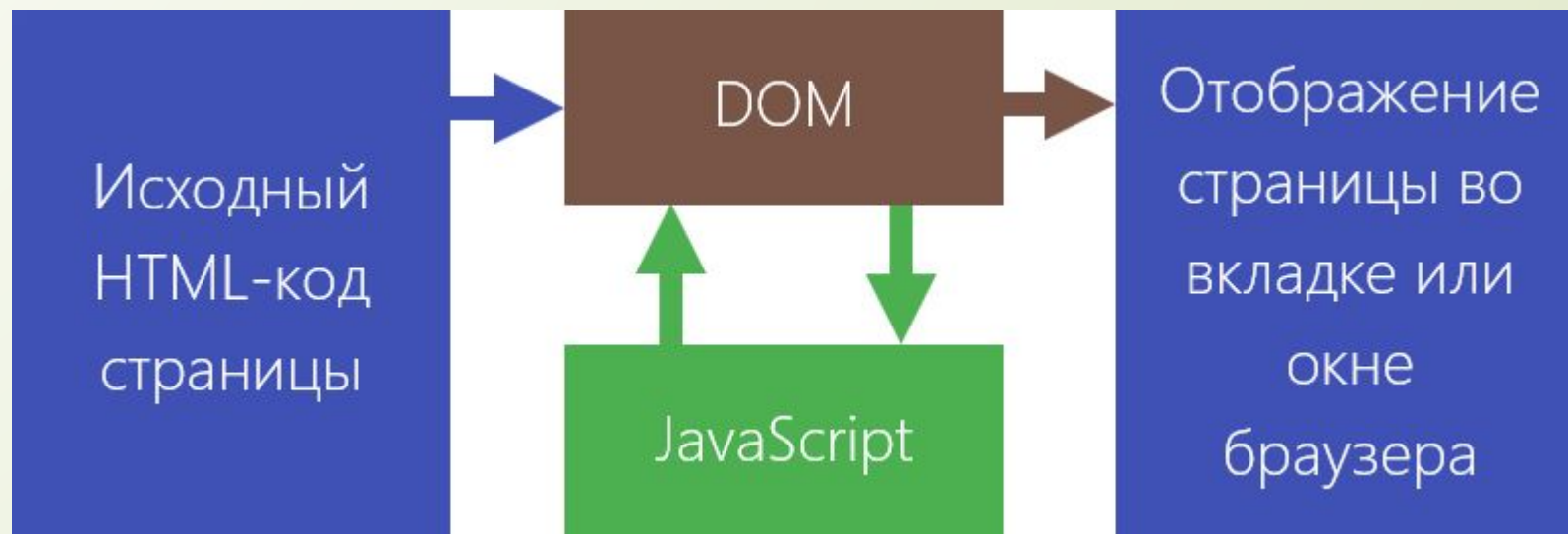
- При наличии атрибута `download` браузер не переходит по ссылке, а предложит скачать документ, указанный в адресе ссылки.

`<a download>Ссылка</a>`

- По умолчанию, при переходе по ссылке документ открывается в текущем окне или фрейме. При необходимости, это условие может быть изменено атрибутом `target` тега `<a>`
- В качестве значения используется имя окна или фрейма, заданное атрибутом `name`. Если установлено несуществующее имя, то будет открыто новое окно. В качестве зарезервированных имен используются следующие:
  - `_blank` – загружает страницу в новое окно браузера.
  - `_self` – загружает страницу в текущее окно.
  - `_parent` – загружает страницу во фрейм-родитель, если фреймов нет, то это значение работает как `_self`.
  - `_top` – отменяет все фреймы и загружает страницу в полном окне браузера, если фреймов нет, то это значение работает как `_self`.



# DOM (Data Object Model)







# Web-сервера



- ▣ *HyperText Transfer Protocol* — «протокол передачи гипертекста») — протокол прикладного уровня передачи данных, изначально — в виде гипертекстовых документов в формате HTML, в настоящее время используется для передачи произвольных данных.
- ▣ **Веб-сервер** — сервер, принимающий HTTP-запросы от клиентов, и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, файлом, медиа-потокom или другими данными.



- Идентификация приложения в сети

- <адрес>:<порт>

- Порты http: 80; 8080

- https: 443

- Наиболее популярные веб-серверы:

1. Apache

2. Nginx

3. IIS