

Модуль 1 «Основы компьютерной геометрии»

Лекция 2 «Введение в HTML, HTML5 и CSS»

к.ф.-м.н., доц. каф. ФН-11, Захаров Андрей Алексеевич,
ауд.:930а(УЛК)
моб.: 8-910-461-70-04,
email: azaharov@bmstu.ru



МГТУ им. Н.Э. Баумана

8 сентября 2019 г.

HTML (HyperText Markup Language) — это *гипертекстовый язык разметки* документа, который позволяет задать местоположение и тип элементов Web-страницы в окне Web-браузера. Слово *гипертекстовый* означает, что фрагменты текста связаны между собой *гиперссылками* — то есть ссылками в документе на другие объекты. А *язык разметки* — это способ встраивать в текст дополнительную информацию. Разметка указывает программам (таким как браузер), как отображать заголовки, абзацы, изображения и т.п. HTML не является языком программирования. Средствами HTML создаётся основная структура веб-страницы, включая текст, рисунки, таблицы, формы и многое другое.

Язык HTML развивается. В настоящее время в разработке находится его пятая версия — HTML5. Она уже является рабочим стандартом W3C, и все современные Web-браузеры поддерживают большинство представленных в ней нововведений.

Web-браузеры, в которых появилась поддержка HTML5:

- ▶ Microsoft Internet Explorer – 9;
- ▶ Mozilla Firefox – 4.0;
- ▶ Google Chrome – 6.0;
- ▶ Opera – 11.1;
- ▶ Apple Safari – 5.0.

Просматривать HTML-документы можно с помощью специальных программ, которые называют Web-браузерами (Web-обозревателями). Web-браузеры отображают документы с форматированием, выполненным на основе исходного кода, описывающего структуру документа. Результат интерпретации HTML-документа, отображаемый в окне Web-браузера, называется Web-страницей. В отличие от HTML-документа Web-страница может содержать не только текст, но и графику, видео, звуковое сопровождение, может реагировать на действия пользователя и т.д. Кроме того, Web-страница может быть результатом интерпретации сразу нескольких HTML-документов. Документы в формате HTML имеют расширение html или htm.

В HTML символы, заключенные в угловые скобки < и >, называются *тегами*, или *дескрипторами*.

Конечный, или *закрывающий*, *тег* (тот, который идет после размечаемого содержимого) содержит символ обратной косой черты после первой угловой скобки. Например, </p> — конечный тег.

Группа из двух тегов (начального и конечного) вместе с заключенным между ними содержимым называется *элементом*. Например, выделим текст курсивом с помощью языка HTML:

```
<i>Текст</i>
```

Некоторые элементы не имеют содержимого и поэтому не нуждаются в конечном теге, например, тег переноса строки
.

Обычно элементы организуются в виде иерархической структуры (в которой одни элементы вложены в другие). Например:

```
<p><i>Текст</i></p>
```

При вложении тегов необходимо соблюдать последовательность их закрытия. Например, такой код использовать нельзя:

```
<p><i>Текст</p></i>
```

В HTML5 наименования тегов и их параметров рекомендуется набирать символами нижнего регистра.

Некоторые теги могут иметь параметры (иногда их называют атрибутами). Они содержат дополнительную информацию об элементе. Параметры указываются после имени тега через пробел в формате `параметр="значение"`. Т.е. значения заключаются в кавычки, а имена параметров в кавычки не заключаются.

Если параметров несколько, то они перечисляются через пробел.

Например:

```
<canvas id="example" width="400" height="400">
```

В этом примере параметру `id` тега `<canvas>` присвоено значение `example`, а параметрам `width` и `height` — значение `400`. Использование русских букв в значениях параметра `id` недопустимо.

HTML. Структура документа

Весь текст HTML-документа расположен между тегами `<html>` и `</html>`. И только тег `<!doctype>` находится вне «всеобъемлющего» тега `<html>`. Этот метатег задаёт, во-первых, версию языка HTML, на которой написана Web-страница, а во-вторых, разновидность данной версии. В случае HTML5 тег `<!doctype>` выглядит так:

```
<!doctype html>
```

Код любой Web-страницы, написанной на HTML5, должен включать в самом начале этот метатег. В противном случае Web-обозреватель не сможет правильно обработать страницу. HTML-документ должен состоять из двух разделов — заголовка (между тегами `<head>` и `</head>`) и содержательной части (между тегами `<body>` и `</body>`).

Пример (HTML5):

```
<!doctype html>
<html>
  <head>
    <title>Название документа</title>
  </head>
  <body>
    <p>Контент документа...</p>
  </body>
</html>
```

HTML5. Раздел head. Техническая информация о документе

Раздел HEAD содержит техническую информацию о странице — заголовок, её описание и ключевые слова для поисковых машин, данные об авторе и времени создания страницы, базовом адресе страницы, кодировке и т. д. Единственным обязательным тегом в разделе HEAD является тег `<title>`. Текст, расположенный между тегами `<title>` и `</title>`, отображается в строке заголовка Web-браузера. Длина заголовка должна быть не более 60 символов, иначе он полностью не поместится в заголовке Web-браузера:

```
<title>Заголовок страницы</title>
```

Очень часто текст между тегами `<title>` и `</title>` используется в результатах, выдаваемых поисковым порталом, в качестве текста ссылки на эту страницу. По этой причине заголовок должен максимально полно описывать содержание страницы. Не следует писать что-то вроде «Главная страница», «Первая страница» и т.п.

Также с помощью непарного тега `<meta>` можно указать кодировку текста:

```
<meta charset="utf-8">
```

Кодировка UTF-8 — это разновидность кодировки Unicode, предназначенная для Web-дизайна. Кодировка Unicode (а значит, и UTF-8) может закодировать все символы всех языков, имеющих на Земле. Для кодирования текста страниц, написанных на языке HTML5, следует применять только кодировку UTF-8. Использование других кодировок не допускается.

В этом разделе располагается все содержимое документа, которые нужно показать пользователю, посещающего веб-страницу, например, текст, изображения, видео, игры, воспроизводимые звуковые дорожки или что-то ещё. Внутри body обычно находятся следующие элементы.

- ▶ Открывающий тег элемента абзаца `<p>`, сообщает браузеру, где в тексте начинается абзац, а закрывающий тег, `</p>`, — где он заканчивается.
- ▶ Элемент `strong` увеличивает толщину шрифта. Если поместить какой-либо текст между открывающим (``) и закрывающим (``) тегами данного элемента, то этот текст будет выделен полужирным шрифтом. Аналогичного эффекта можно добиться с помощью тега ``:

```
<b>Полужирный шрифт</b>
```

- ▶ Элемент `a` (элемент привязки), создаёт на веб-странице *гиперссылку*. Если щёлкнуть по гиперссылке (или *ссылке*), то можно попасть на любую другую веб-страницу. Например:

```
<a href="www.bmstu.ru">Щёлкните здесь!</a>
```

В данном случае, если щёлкнуть кнопкой мыши по фразе «Щёлкните здесь», то произойдет переход на сайт МГТУ им. Н.Э. Баумана.

При помощи каскадных таблиц стилей (CSS — Cascading Style Sheets), можно задавать точные характеристики практически всех элементов Web-страницы. Это позволяет контролировать внешний вид Web-страницы в окне Web-браузера и приближает возможности Web-дизайна к настольным издательским системам. Разработчик может указать параметры шрифта, цвет текста и фона, выравнивание, создать рамку и расположить элементы на странице произвольным образом. Задать стиль можно тремя способами: встроить определение стиля в тег, встроить определения стилей в заголовок HTML-документа или вынести таблицу стилей в отдельный файл.

Встраивание определения стиля в тег

Определение стиля встраивается в любой тег с помощью параметра `style`. Или, иначе, параметр `style` поддерживают все теги. Рассмотрим задание размера шрифта. В HTML это делает тег ``. Размер шрифта указывается с помощью атрибута `size` в условных единицах от 1 до 7. Для задания этого размера в пунктах используется CSS:

```
<font style="font-size: 12pt">Текст</font>
```

Применение стилей позволяет задавать точные характеристики практически всех элементов Web-страницы.

Значение параметра `style` (`font-size: 12pt`) называется *определением стиля* или *стилем*. Элемент определения стиля (`font-size`) называется атрибутом. Каждый атрибут имеет *значение* (`12pt`), указываемое после двоеточия.

Если определение стиля состоит из нескольких атрибутов, то они указываются через точку с запятой:

```
<font style="font-size: 12pt; color: red">Текст</font>
```

Если какое-либо значение атрибута требует наличия кавычек, то оно указывается в апострофах:

```
<font style="font-size: 12pt; color: red;  
font-family: 'Times New Roman'">Текст</font>
```

Встраивание определения стилей в заголовок HTML-документа

Все определения стилей можно собрать в одном месте. В этом случае стили указываются между тегами `<style>` и `</style>`. Сам тег `<style>` должен быть расположен в разделе `head` HTML-документа.

Атрибуты определения стиля, указанные между тегами `<style>` и `</style>`, заключаются в фигурные скобки. Если атрибутов несколько, то они перечисляются через точку с запятой:

```
<Селектор> {  
    <Атрибут 1>: <Значение 1>;  
    <Атрибут 2>: <Значение 2>;  
    . . .  
    <Атрибут n-1>: <Значение n-1>;  
    <Атрибут n>: <Значение n>  
}
```

Селектор используется для привязки стиля к элементу страницы, на который он должен распространять свое действие. Например:

```
body {  
    font-family: Arial;  
}
```

Селектором у этого стиля выступает имя элемента `<body>`.

Вынесение таблицы стилей в отдельный файл

Таблицу стилей можно вынести в отдельный файл. Файл с таблицей стилей обычно имеет расширение `css` и может редактироваться любым текстовым редактором.

Отдельный файл с таблицей стилей прикрепляется к HTML-документу с помощью одинарного тега `<link>`. В параметре `href` указывается абсолютный или относительный URL-адрес файла, а в параметре `rel` должно быть значение `stylesheet`, показывающее, что присоединяемый таким образом документ содержит таблицу стилей:

```
<head>
  <meta charset="utf-8" />
  <title>Draw line spline</title>
  <link rel="stylesheet" href="style.css">
</head>
```

Преимущество внешних таблиц стилей в том, что их можно привязать сразу к нескольким Web-страницам, задав тем самым для них одинаковое представление.