

## **Лабораторная работа № 1**

### **ОСНОВНЫЕ ТЕГИ HTML**

**Цель работы:** изучить структуру HTML-документа и использование основных тегов и атрибутов.

#### **Теоретические сведения для выполнения работы**

HTML (Hyper Text Markup Language) – язык разметки гипертекста. Для разметки HTML документа используются теги (флаги разметки).

Тег – это определенная последовательность символов, заключенные между знаками < (больше) и > (меньше).

Для того, чтобы создать HTML документ необходимо:

- открыть HTML-редактор (например, Visual Studio Code);
- набрать произвольный текст и разметить между HTML тегами;
- сохранить файл с расширением .htm или .html.

Теперь если открыть созданный файл с помощью веб-браузера, он будет отображен как веб-страница.

Любая Web-страница структурно разбивается на две части: заголовок и тело. В заголовке указывается служебная информация обо всей странице, а в теле Web-документа описывается ее содержимое вместе с правилами оформления. При этом заголовок Web-страницы ограничивается тэгами <head> и </head>, а тело документа обозначается тэгами <body> и </body>. По правилам хорошего стиля программирования перед <html> ставится идентификатор применяемого стандарта HTML.

В заголовок документа могут входить: тэг, отображающий наименование Web-страницы, тэг стилевого оформления, тэг выполняемого сценария и так называемые метаданные. Для внедрения метаданных в Web-страницу применяется тэг <meta>. Тегов <meta> может быть несколько. Структура Web-страницы представлена на рисунке 1.

Между тэгами <title> и </title> указывается название страницы, которое отображается в строке заголовка окна при просмотре странички в браузере.

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  Тело документа
</body>
</html>

```

Рис. 1 Структура Web-страницы

В теле документа заключается гипертекст, который определяет собственно Web-страницу. Гипертекстом является произвольная часть документа, которую разрабатывает автор и которая отображается браузером. Внутри элемента `<body>` можно использовать все элементы, предназначенные для дизайна Web-страницы. Внутри начального тега `<body>` могут располагаться атрибуты, обеспечивающие установки для всей страницы в целом. Синтаксис использования атрибутов представлен на рисунке 2:

```

<body атрибут_1=«значение1»
    атрибут_2=«значение2»...>

```

Рис. 2 Синтаксис использования атрибутов тега

У тега `<body>` могут быть следующие атрибуты:

- `marginheight` – определяет ширину (в пикселах) верхнего и нижнего полей документа. Работает только в браузерах Netscape.
- `topmargin` – определяет ширину (в пикселах) верхнего и нижнего полей документа. Работает только в браузерах Internet Explorer.
- `marginwidth` – определяет ширину (в пикселах) левого и правого полей документа. Работает только в браузерах Netscape.
- `leftmargin` – определяет ширину (в пикселах) левого и правого полей документа. Работает только в браузерах Internet Explorer.

- background – определяет изображение для "заливки" фона. Значение задается в виде полного URL или имени файла с картинкой в формате GIF или JPG.
- bgcolor – определяет цвет фона документа.
- text – определяет цвет текста в документе.
- link – определяет цвет гиперссылок в документе.
- vlink – определяет цвет гиперссылок на документы, которые вы уже просмотрели.
- alink – определяет цвет подсветки гиперссылок в момент нажатия.

Шрифт задается тегом <font>, который имеет следующие атрибуты: face – имена шрифтов, разделенные запятыми, size – размер от 1 до 7 (по умолчанию 3), color – цвет шрифта.

В HTML существуют специальные теги для заголовков: от <h1> (самого крупного) до <h6> (самого мелкого). Для создания абзаца используется тег <p>.

Каждая HTML-страница имеет свой уникальный адрес в Интернете, который называется универсальным указателем ресурса (URL). Для перехода на Web-страницу используется гиперссылка. За организацию гиперссылок в языке HTML отвечает элемент <a>...</a>, который чаще всего использует следующий шаблон.

```
<a href="Адрес ссылки (или URL)"> текст для щелчка </a>
```

Рис. 3 Создание гиперссылки

Также можно создавать внутренние ссылки, позволяющих переходить между элементами документа. Сначала в нужных местах устанавливается метка с помощью атрибута name

```
<A name="metka">...</a>
```

или

```
<A name="http://адрес/файл.html#метка">...</a>
```

Рис. 4 Создание меток внутренних гиперссылок

Затем определяется ссылка на метку (рисунок 5)

```
<a href="#metka"> текст для щелчка </a>
```

Рис. 5 Создание внутренних гиперссылок

Для хранения изображений используются десятки различных форматов: gif, jpeg, bmp, psx, vmf и др. Однако для работы с изображениями в документах HTML обычно выбирают форматы, которые распознаются браузерами gif, jpeg. Для отображения других форматов необходимо устанавливать плагины или запускать Java-апплеты. Рекомендуется использовать графические файлы размером до 50 Кб.

Для вставки изображения используется тэг <IMG> с атрибутом src, который указывает URL графического файла.

```

```

Рис. 6 Вставка изображения

При организации ссылки, в качестве которой используется изображение, применяется шаблон из комбинации двух тэгов:

```
<a href="Адрес ссылки"> <IMG src="Ссылка на рисунок"></a>
```

Рис. 7 Создание гиперссылки

### Теги списков

Список отличается от обычного текста тем, что пользователю не надо думать о нумерации его пунктов. Если список дополняется новыми пунктами или укорачивается, нумерация корректируется автоматически. Различают маркированный, нумерованный списки, списки с определениями. Шаблон маркированного или нумерованного списка представлен на рисунке 8.

```
<ul>
  <li> элемент 1</li>
  <li> элемент 2</li>
  <li> элемент 3</li>
</ul>
```

Рис. 8 Структура маркированного списка

В тэге маркированного списка можно применить атрибут `type`, принимающий значения `disc` (круг), `circle` или `round` (окружность), `square` (квадрат). В качестве маркеров можно использовать графические изображения. Тогда, после указания `<li>` определяется тэг ``.

Структура нумерованного списка похожа на маркированный:

```
<ol>
  <li> элемент 1</li>
  <li> элемент 2</li>
  <li> элемент 3</li>
</ol>
```

Рис. 9 Структура нумерованного списка

Каждый пункт маркируется элементом упорядоченной последовательности: арабскими или римскими числами, буквами латинского алфавита. Способ нумерации задается при помощи атрибута `type` тега `<ol>`, который может принимать следующие значения:

- 1) `type="1"` — 1,2,3,4
- 2) `type="i"` — i, ii, iii, iv,
- 3) `type="I"` — I, II,III,IV
- 4) `type="a"` — a,b,c,d
- 5) `type="A"` — A,B,C,D

Можно формировать также вложенные списки, структура которых представлена на рисунке 10

```
<ol>
  <li>Пункт 1</li>
  <li>Пункт 2
    <ul>
      <li>Подпункт 2.1.</li>
      <li>Подпункт 2.2.</li>
    </ul>
  </li>
  <li>Пункт 3</li>
</ol>
```

Рис. 10 Структура вложенного списка

## Теги таблиц

Элемент `<table>` служит контейнером для элементов, определяющих содержимое таблицы. Любая таблица состоит из строк и ячеек, которые задаются с помощью тегов `<tr>` и `<td>`. Для отображения границ таблицы используется атрибут `border`. С помощью тега `<th>` можно создать табличный заголовок. Текст элемента `th` центрируется и выделяется жирным шрифтом. С помощью атрибута `colspan` можно указать на сколько столбцов должна быть растянута указанная ячейка, а с помощью атрибута `rowspan` – на сколько строк должна быть растянута указанная ячейка.

## Теги создания веб-форм

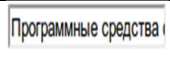
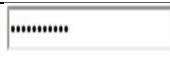


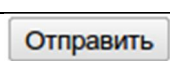
Форма ограничивается тегами `<form>` и `</form>`. Между этими тегами располагаются теги, создающие элементы управления. При необходимости, элементы управления могут размещаться в ячейках таблицы, которая полностью располагается в форме.

Всего существует три тега для вставки элементов управления в страницу: `<input />` — общий тег для большинства элементов ввода, `<select></select>` — выпадающий список, `<textarea>...</textarea>` — текстовое поле.

Большинство элементов управления отображаются при помощи тега `<input />`. Конкретный вид элемента управления зависит от атрибута `type` (`<input type="xxx" />`). Значения атрибута представлены в таблице 1.

Таблица 1

Значения атрибута `type` тега `<input>`

Значение атрибута <code>type</code>	Функциональное назначение	Внешний вид
<code>text</code>	Обычное текстовое поле	
<code>password</code>	«Маскированное» текстовое поле	
<code>checkbox</code>	Флажок	
<code>radio</code>	Переключатель	
<code>submit</code>	Кнопка для отправки формы на сервер	

Продолжение таблицы 1

Значение атрибута type	Функциональное назначение	Внешний вид
reset	Кнопка для очистки полей формы	
button	Обычная кнопка (её действие можно будет задать позднее)	
image	Изображение-кнопка, определяющее координаты нажатия	
file	Кнопка для выбора файла для присоединения к форме координаты нажатия	

Для расширения функциональности форм стандарте HTML5 представлены новые элементы управления, которые уже частично реализованы в браузерах. Если элемент управления не поддерживается браузером, то он будет заменён на обычное текстовое поле, поэтому их рекомендуется использовать. Эти элементы представлены в таблице 2

Таблица 2

Значения атрибута type стандарта HTML5

Значение атрибута type	Функциональное назначение	Внешний вид
search	Поле для поискового запроса. Имеет кнопку очистки и подсказки из истории поиска в браузере.	
email, url, tel	Поля имеют вид обычного текстового поля, однако они используются при обработке браузерами мобильных устройств.	
range	Ползунок, позволяющий указать числовое значение	
number	Текстовое поле с кнопками увеличения/уменьшения числового значения	
time	Поле для ввода времени	
date	Поле для ввода даты	

Также существуют такие атрибуты как `value`, который используется для определения начального текста в текстовых полях или для надписи на кнопках. Атрибут `checked` позволяет отметить флажок или переключатель по умолчанию. Атрибут `size` описывает длину поля ввода. Атрибут `disabled` позволяет «деактивировать» элемент управления.

Также существуют параметры, поддерживаемые ещё не всеми браузерами, но достаточно удобные: атрибут `placeholder` позволяет задать текст, отображаемый если текстовое поле не заполнено, атрибут `autofocus` позволяет установить фокус ввода на элемент после загрузки страницы, атрибут `required` позволяет пометить поле как обязательное для заполнения.

Чтобы из группы переключателей выбирался только один — необходимо всем элементам из группы задать одинаковый атрибут `name`.

Для создания выпадающих списков используется тег `<select>`. Внутри него вкладываются теги `<option>` с возможными вариантами. Тег `<option>` также имеет атрибуты `checked` и `disabled`

Для создания многострочного текстового поля используется парный тег `<textarea>`. Поддерживает атрибуты `rows` и `cols`, задающие количество строк и столбцов в символах.

Тег `<legend>` применяется для создания заголовка группы элементов формы, которая определяется с помощью тега `<fieldset>`. Группа элементов обозначается в браузере с помощью рамки, а текст, который располагается внутри контейнера `<legend>`, встраивается в эту рамку.

### **Тег `<div>`**

Тег `<div>` является блочным элементом и предназначен для выделения фрагмента документа с целью изменения вида содержимого. Как правило, вид блока управляется с помощью стилей.

### **Тег `<span>`**

Тег `<span>` предназначен для определения строчных элементов документа. В отличие от блочных элементов,



таких как <table>, <p> или <div>, с помощью тега <span> можно выделить часть информации внутри других тегов и установить для нее свой стиль. Например, внутри абзаца (тега <p>) можно изменить цвет и размер первой буквы, если добавить начальный и конечный тег <span> и определить для него стиль текста.

## Семантические теги

В стандарт HTML5 были введены семантические теги, с помощью которых можно сделать страницы сайтов более понятными для поисковых систем и браузеров. Следует отметить, что есть несколько семантических тегов, которые рекомендуется использовать для разметки страниц вместо <div> и <span>:

1. <header> – заголовочный блок сайта, обычно содержит навигацию, повторяется на всех страницах сайта;
2. <footer> – заключительная часть смыслового раздела или всего сайта, которая содержит информацию об авторах, список литературы, копирайт и так далее;
3. <nav> – навигационное меню.
4. <main> – основное, не повторяющееся на других страницах, содержание страницы
5. <section> позволяет группировать логически связанное содержимое в документе, может применяться для блока новостей, контактной информации, глав текста, вкладок в диалоговом окне;
6. <article> – это может быть пост на форуме, статья в журнале или газете, заметка в блоге, сообщение пользователя или другая независимая контент-единица
7. тэг <mark> – выделить (подсветить) важную часть в тексте.

В HTML5 можно создавать подписи для иллюстраций с помощью тэгов: <figure> и <figcaption>.

```
<figure>
  <img src='foto.jpg' width='300' height='230' />
  <figcaption>Моя      замечательная      фотогра-
фия</figcaption>
</figure>
```

Рис. 11 Создание подписи рисунка тегом figcaption

## Задания к лабораторной работе № 1

**Задание 1.** Самостоятельно создайте новый документ index.html с произвольным текстом одной тематики со следующим содержанием

**1.1.** В созданном документе должен быть заголовок в центре страницы (теги h1) произвольного цвета, гарнитуры и начертания. Текст заголовка должен совпадать с названием страницы. Три абзаца текста более 4–5 строк каждый абзац. Для каждого абзаца свой шрифт, размер, цвет (тег font). Для каждого абзаца придумать заголовок (h3), а также заголовок для всего текста (h2). Внутри каждого абзаца выделить несколько слов элементами span.

**1.2.** Создайте список следующего вида, соответствующий выбранной тематике

1. Элемент 1
  - Элемент 1.1
  - Элемент 1.2
2. Элемент 2
  - Элемент 2.1
  - Элемент 2.2
  - Элемент 2.3
3. Элемент 3
  - Элемент 3.1
  - Элемент 3.2

Перед списком вставить заголовок, отражающий его содержание (h2).

**1.3.** Таблица, содержащая произвольный текст выбранной тематики с числовой информацией. Перед таблицей вставить заголовок (h2). Форма таблицы может быть произвольная, но обязательно должны быть объединенные ячейки строк и столбцов. Пример таблицы представлен на рисунке 12

Производитель	Товар	Цена товара	
		с НДС	Без НДС
Erich Krause	Карандаш	30	40
	Ластик	20	30
	Ручка гелевая	10	20
<b>Итого</b>		<b>60</b>	<b>90</b>

Рис. 12 Пример таблицы (не повторять его!)

**Задание 2** Создать еще один HTML-документ `se-  
cond.html` с текстовой информацией, по содержанию свя-  
занной с первой страницей со следующим содержимым:

**2.1.** Два элемента `<div>` с произвольным текстом из  
нескольких абзацев. Перед каждым элементом заголовок  
(**h3**).

**2.2.** Любой рисунок. Перед рисунком оформить за-  
головок (**h2**).

**2.3.** Элемент `<span>`, содержащий авторский знак ©  
и ФИО автора

**Задание 3.** Элемент текста на первой web-странице  
и в таблице сделать ссылкой на вторую страницу, а рису-  
нок на второй странице – ссылкой на первую.

**Задание 4.** Создайте новый документ с формой,  
изображённую ниже

## Данные студента

Имя студента	<input type="text" value="обязательно заполнить"/>
Отчество студента	<input type="text" value="обязательно заполнить"/>
Фамилия студента	<input type="text" value="обязательно заполнить"/>
Адрес почты	<input type="text"/>
Контактный номер	<input type="text"/>
Какую социальную сеть вы используете?	<input type="checkbox"/> VK <input type="checkbox"/> Facebook <input type="checkbox"/> Instagram
<input type="button" value="отправить"/>	

Рис. 13 Форма для задания 4

**Задание 5.** Создать третий документ используя се-  
мантические теги. В документе должны быть верхние  
(header) и нижние (footer) колонтитулы, а также инфор-  
мация должна быть разбита на секции (section), также  
должны быть использованы теги `aside`, `figure`, `nav`

Тег `footer` должен содержать данные автора (ФИО,  
курс, группа, телефон и email).

## Контрольные вопросы

1. Как создать простейшую Web-страницу?
2. Что представляет собой структура HTML-документа?
3. Какие теги относятся к служебным, а какие к структурным?
4. В каком элементе указывается заголовок страницы?
5. Для чего используется метаданные? Каким тегом они указываются в структуре HTML-документа?
6. Для чего используется !DOCTYPE?
7. Каким образом можно изменить установки для всей страницы в целом?
8. Какие виды списков предусмотрены в HTML?
9. Как изменить вид маркера в маркированном списке?
10. Как изменить вид маркера в нумерованном списке?
11. Из каких частей состоит список с определениями?
12. Создайте вложенный список из нумерованного, состоящего из двух пунктов, в каждом из которых два маркированных подпункта. Символ маркера квадрат
13. Как создать маркированный список?
14. Как создать нумерованный список?
15. С помощью каких элементов создается таблица?
16. Как объединить несколько столбцов в один?
17. Как объединить несколько строк в одну?
18. Что такое гипертекст? Как создать гиперссылку?
19. Как организовать переход в начало или конец документа?
20. Создайте таблицу с гиперссылкой на другую веб-страницу
21. Каким образом создать гиперссылку, используя графическое изображение?
22. Какие теги называются семантическими? Назовите основные семантические теги
23. Для чего необходимы теги <section> и <article>?
24. Для чего необходимы теги <footer> и <header>?
25. Какой тег определяет элемент формы? Какой атрибут определяет вид элемента формы?
26. Какие элементы необходимы для построения формы?
27. Для чего необходимы теги <nav> и <main>?
28. Для чего предназначен тег <th>?

**Примечание:** ссылки на вспомогательный материал

<https://webref.ru/html>

<https://ru.wikibooks.org/wiki/HTML/HTML5>

<https://tinyurl.com/o298cs6>

[https://intuit.ru/studies/courses/3734/976/lecture/27478](https://intuit.ru/studies/courses/3734/976/lecture/27478?page=2)  
[?page=2](#)

<https://html5css.ru/tags/default.php>