

Язык HTML — язык тегов. Теги описывают структуру HTML-документа. Теги оформляются угловыми скобками <имя тега>, между которыми прописывается имя тега.

Теги располагаются в HTML-документе в соответствии с правилами разметки (порядок следования, правило вложенности тегов), создавая разделы будущей веб-страницы.

Кроме тегов, HTML-документы могут содержать [специальные символы](#).

Браузер просматривает (интерпретирует) HTML-документ, выстраивая его структуру (DOM) и отображая ее в соответствии с инструкциями, включенными в этот файл (таблицы стилей, скрипты). Если разметка правильная, то в окне браузера будет отображена HTML-страница, содержащая HTML-элементы — заголовки, таблицы, изображения и т.д.

Процесс интерпретации (парсинг) начинается прежде, чем веб-страница полностью загружена в браузер. Браузеры обрабатывают HTML-документы последовательно, с самого начала, при этом обрабатывая CSS и соотнося таблицы стилей с элементами страницы.

Практически все теги имеют атрибуты (глобальные, применяемые для всех html-элементов, и собственные), указываемые в формате атрибут="значение". Атрибуты позволяют изменять свойства элемента, для которого они заданы. Атрибуты прописываются в начальном теге элемента.

# Структура веб-страницы

## 1. Структура HTML-документа

### 1.1. Тег <html>

### 1.2. Тег <head>

#### 1.2.1. Тег <title>

#### 1.2.2. Тег <meta>

#### 1.2.3. Тег <style>

#### 1.2.4. Тег <link>

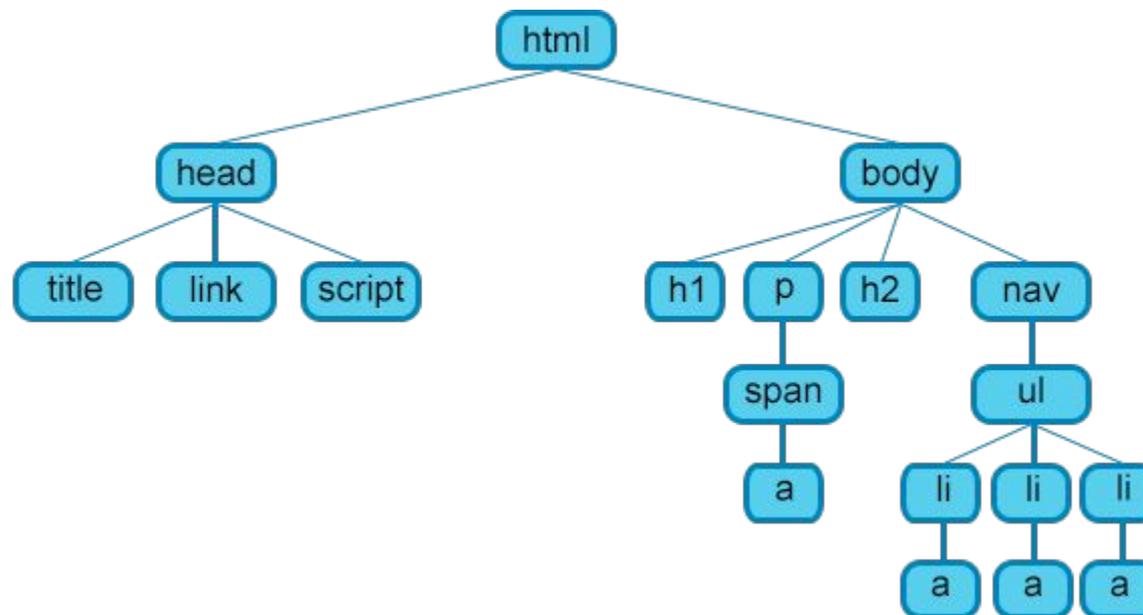
#### 1.2.5. Тег <script>

### 1.3. Тег <body>

Язык HTML следует правилам, которые содержатся в файле объявления типа документа (Document Type Definition, или DTD). DTD представляет собой XML-документ, определяющий, какие теги, атрибуты и их значения действительны для конкретного типа HTML. Для каждой версии HTML есть свой DTD.

DOCTYPE отвечает за корректное отображение веб-страницы браузером. DOCTYPE определяет не только версию HTML (например, html), но и соответствующий DTD-файл в Интернете.

Элементы, находящиеся внутри тега <html>, образуют дерево документа, так называемую объектную модель документа, DOM (document object model). При этом элемент <html> является корневым элементом.



## Элемент <html>

Является корневым элементом документа. Все остальные элементы содержатся внутри тегов <html>...</html>. Все, что находится за пределами тегов, не воспринимается браузером как код HTML и никак им не обрабатывается. Для элемента доступны атрибуты manifest и xmlns, а также [глобальные атрибуты](#).

### manifest

С помощью значения атрибута указывается путь к документу кэша манифеста, например:  
<html manifest="about\_company.appcache">

### xmlns

Задаёт пространство имен только для XHTML документа. Принимаемое значение xmlns="http://www.w3.org/1999/xhtml". По адресу http://www.w3.org/1999/xhtml находится справочник по пространству имён XHTML — определение имен элементов (тегов) и их атрибутов, которые используются в данной версии XHTML.

## Элемент `<head>`

Раздел `<head>...</head>` содержит техническую информацию о странице: заголовок, описание, ключевые слова для поисковых машин, кодировку и т.д. Введенная в нем информация не отображается в окне браузера, однако содержит данные, которые указывают браузеру, как следует обрабатывать страницу.

Для элемента доступны [глобальные атрибуты](#)

## Элемент <title>

Обязательным тегом раздела <head> является тег <title>. Текст, размещенный внутри этого тега, отображается в строке заголовка веб-браузера. Длина заголовка должна быть не более 60 символов, чтобы полностью поместиться в заголовке. Текст заголовка должен содержать максимально полное описание содержимого веб-страницы.

Для элемента доступны [глобальные атрибуты](#).

## Элемент <meta>

Необязательным тегом раздела <head> является одинарный тег <meta>. С его помощью можно задать описание содержимого страницы и ключевые слова для поисковых машин, автора html-документа и прочие свойства метаданных. Элемент <head> может содержать несколько элементов <meta>, потому что в зависимости от используемых атрибутов они несут различную информацию.

```
<meta name="description" content="Описание содержимого страницы">
```

Описание содержимого страницы и ключевые слова одновременно можно указывать на нескольких языках, например, на русском и английском:

```
<meta name="description" lang="ru" content="Описание содержимого  
страницы">
```

```
<meta name="description" lang="en" content="Description">
```

```
<meta name="keywords" lang="ru" content="Ключевые слова через запятую">
```

```
<meta name="keywords" lang="en" content="Keywords">
```

## Элемент <style>

Внутри этого элемента задаются стили, которые используются на странице. Для задания стилей в html-документе используется язык CSS. Таких элементов на странице может быть несколько.

Для элемента доступны атрибуты `media`, `scoped`, `type`, а также [глобальные атрибуты](#).

Внутри этого элемента можно записывать код форматирования как самих элементов веб-страницы, так и веб-страницы целиком.

```
<style type="text/css">
```

```
.paper {  
width: 200px;  
height: 300px;  
background-color: #ef4444;  
color: #666666;
```

## Элемент <link>

Задать стили для документа можно также при помощи другого способа — записать их в отдельный файл с расширением .css, например, style.css.

Подключить файл со стилями к веб-странице можно двумя способами:

через директиву @import url

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>@import url(style.css);
```

```
</style>
```

```
<meta>
```

```
<title> </title>
```

```
</head>
```

с использованием элемента <link>. Элемент не требует закрывающего тега. Данный элемент определяет отношение между текущей страницей и другими документами. Таких элементов на странице может быть несколько. Запись будет иметь следующий вид:

```
<link rel="stylesheet" href="style.css" type="text/css">
```

Для элемента доступны атрибуты href, hreflang, media, rel, type, а также [глобальные атрибуты](#).

## Элемент <script>

Элемент <script> позволяет присоединять к документу различные сценарии.

Закрывающий тег обязателен, при этом текст сценария может располагаться либо внутри этого элемента, либо во внешнем файле. Если текст сценария расположен во внешнем файле, то он подключается с помощью атрибутов элемента. Для элемента доступны атрибуты `async`, `charset`, `defer`, `src`, `type`, а также [глобальные атрибуты](#).

`Async` - Атрибут указывает на то, что сценарий будет выполняться асинхронно с остальной частью страницы (сценарий начнет выполняться одновременно с загрузкой страницы).

`Charset` - Определяет кодировку символов

`Defer` - Интерпретация сценариев откладывается до окончания отображения документа на устройстве пользователя.

`Src` - Указывает на месторасположение файла со сценарием, значение атрибута — это url файла, содержащего JavaScript-программу.

`Type` - Используются для объявления языка сценария, использованного при составлении содержимого тега.

## Элемент <body>

В этом разделе располагается все содержимое документа. Для элемента доступны [глобальные атрибуты](#).

HTML теги — основа языка HTML. Каждый тег несет в себе определенную информацию, может описывать документ в общем или способ форматирования текста. Все содержимое веб-страницы задается с помощью тегов.

Тег помещается в угловые скобки <тег>. Чаще всего для тега задается парный закрывающий тег, но в некоторых случаях он отсутствует. Информация, заключенная между открывающим и закрывающим тегом, называется его контейнером.

## Корневой элемент

<code>&lt;html&gt;&lt;/html&gt;</code>	корневой элемент html-документа	block
--	---------------------------------	-------

## Служебные теги

<code>&lt;!--...--&gt;</code>	комментарий	none
<code>&lt;!DOCTYPE&gt;</code>	объявление типа документа	none
<code>&lt;head&gt;&lt;/head&gt;</code>	контейнер для метаданных html-документа	none
<code>&lt;title&gt;&lt;/title&gt;</code>	заголовок / имя html-документа	none
<code>&lt;meta&gt;</code>	мета-данные веб-страницы	none
<code>&lt;link&gt;</code>	подключает внешние таблицы стилей	none
<code>&lt;script&gt;&lt;/script&gt;</code>	подключает сценарии к странице	none
<code>&lt;style&gt;&lt;/style&gt;</code>	подключает встраиваемые таблицы стилей	none
<code>&lt;base&gt;</code>	базовый url-адрес, относительно которого вычисляются относительные адреса	none

## Таблицы

<code>&lt;table&gt;&lt;/table&gt;</code>	html-таблица	table
<code>&lt;tr&gt;&lt;/tr&gt;</code>	строка таблицы	table-row
<code>&lt;th&gt;&lt;/th&gt;</code>	заголовок столбца таблицы	table-cell
<code>&lt;td&gt;&lt;/td&gt;</code>	ячейка таблицы	table-cell
<code>&lt;thead&gt;&lt;/thead&gt;</code>	блок заголовков таблицы	table-header-group
<code>&lt;tbody&gt;&lt;/tbody&gt;</code>	тело таблицы	table-row-group
<code>&lt;tfoot&gt;&lt;/tfoot&gt;</code>	нижний колонтитул таблицы	table-footer-group
<code>&lt;caption&gt;&lt;/caption&gt;</code>	подпись к таблице	table-caption
<code>&lt;col&gt;</code>	выбирает для форматирования столбцы	table-column
<code>&lt;colgroup&gt;&lt;/colgroup&gt;</code>	контейнер для одного или нескольких <code>&lt;col&gt;</code>	table-column-group

## Текст

<code>&lt;h1&gt;&lt;/h1&gt;</code> - <code>&lt;h6&gt;&lt;/h6&gt;</code>	заголовки шести уровней	block
<code>&lt;p&gt;&lt;/p&gt;</code>	параграфы в тексте	block
<code>&lt;br&gt;</code>	перенос текста на новую строку	none
<code>&lt;hr&gt;</code>	горизонтальная линия	block
<code>&lt;wbr&gt;</code>	возможное место разрыва длинной строки	none
<code>&lt;blockquote&gt;&lt;/blockquote&gt;</code>	большая цитата	block
<code>&lt;cite&gt;&lt;/cite&gt;</code>	источник цитирования	inline
<code>&lt;q&gt;&lt;/q&gt;</code>	краткая цитата	inline
<code>&lt;code&gt;&lt;/code&gt;</code>	фрагмент программного кода	inline
<code>&lt;kbd&gt;&lt;/kbd&gt;</code>	текст, вводимый пользователем с клавиатуры	inline

## Изображения

<code>&lt;img&gt;</code>	html-изображения	inline-block
<code>&lt;map&gt;&lt;/map&gt;</code>	активные области на карте-изображении	inline
<code>&lt;area&gt;</code>	гиперссылка с текстом или активная область внутри карты-изображения	inline
<code>&lt;canvas&gt;&lt;/canvas&gt;</code>	холст-контейнер для динамического отображения изображений	inline-block

## Ссылки

<code>&lt;a&gt;&lt;/a&gt;</code>	гиперссылка	inline
----------------------------------	-------------	--------

## Списки

<code>&lt;ol&gt;&lt;/ol&gt;</code>	упорядоченный нумерованный список	block
<code>&lt;ul&gt;&lt;/ul&gt;</code>	маркированный список	block
<code>&lt;li&gt;&lt;/li&gt;</code>	элемент списка	list-item
<code>&lt;dl&gt;&lt;/dl&gt;</code>	контейнер для термина и его описания	block
<code>&lt;dt&gt;&lt;/dt&gt;</code>	задаёт термин	block
<code>&lt;dd&gt;&lt;/dd&gt;</code>	расшифровывает термин	block

## Группировка контента

<code>&lt;body&gt;&lt;/body&gt;</code>	тело html-документа	block
<code>&lt;div&gt;&lt;/div&gt;</code>	контейнер для разделов html-документа, группирует блочные элементы	block
<code>&lt;span&gt;&lt;/span&gt;</code>	контейнер для строчных элементов	inline
<code>&lt;header&gt;&lt;/header&gt;</code>	секция для вводной информации сайта или группы навигационных ссылок	block
<code>&lt;footer&gt;&lt;/footer&gt;</code>	секция для нижнего колонтитула документа или раздела	block
<code>&lt;section&gt;&lt;/section&gt;</code>	логическая область (раздел) страницы, обычно с заголовком	block
<code>&lt;article&gt;&lt;/article&gt;</code>	раздел контента, образующий независимую часть документа или сайта	block
<code>&lt;aside&gt;&lt;/aside&gt;</code>	контент страницы, имеющий косвенное отношение к основному контенту	block
<code>&lt;hgroup&gt;&lt;/hgroup&gt;</code>	секция для группы заголовков	block
<code>&lt;nav&gt;&lt;/nav&gt;</code>	раздел документа, содержащий навигационные ссылки по сайту	block
<code>&lt;figure&gt;&lt;/figure&gt;</code>	независимый контейнер для такого контента как изображения, диаграммы и т.п.	block
<code>&lt;figcaption&gt;&lt;/figcaption&gt;</code>	заголовок для элемента <code>&lt;figure&gt;</code>	block
<code>&lt;details&gt;&lt;/details&gt;</code>	контейнер с дополнительными сведениями, который можно открыть или закрыть	block
<code>&lt;summary&gt;&lt;/summary&gt;</code>	видимый заголовок для элемента <code>&lt;details&gt;</code>	block

<code>&lt;blockquote&gt;&lt;/blockquote&gt;</code>	большая цитата	block
<code>&lt;cite&gt;&lt;/cite&gt;</code>	источник цитирования	inline
<code>&lt;q&gt;&lt;/q&gt;</code>	краткая цитата	inline
<code>&lt;code&gt;&lt;/code&gt;</code>	фрагмент программного кода	inline
<code>&lt;kbd&gt;&lt;/kbd&gt;</code>	текст, вводимый пользователем с клавиатуры	inline
<code>&lt;pre&gt;&lt;/pre&gt;</code>	выводит текст с пробелами и переносами	block
<code>&lt;samp&gt;&lt;/samp&gt;</code>	результат выполнения сценария	inline
<code>&lt;var&gt;&lt;/var&gt;</code>	выделяет переменные из программ	inline
<code>&lt;del&gt;&lt;/del&gt;</code>	перечёркивает текст, помечая как удаленный	inline
<code>&lt;s&gt;&lt;/s&gt;</code>	перечёркивает неактуальный текст	inline
<code>&lt;dfn&gt;&lt;/dfn&gt;</code>	выделяет термин курсивом	inline
<code>&lt;em&gt;&lt;/em&gt;</code>	выделяет важные фрагменты текста курсивом	inline
<code>&lt;i&gt;&lt;/i&gt;</code>	выделяет текст курсивом без акцента	inline
<code>&lt;strong&gt;&lt;/strong&gt;</code>	выделяет полужирным важный текст	inline
<code>&lt;b&gt;&lt;/b&gt;</code>	задает полужирное начертание отрывка текста, без дополнительного акцента	inline
<code>&lt;ins&gt;&lt;/ins&gt;</code>	подчёркивает изменения в тексте	inline
<code>&lt;u&gt;&lt;/u&gt;</code>	выделяет отрывок текста подчёркиванием, без дополнительного акцента	inline
<code>&lt;mark&gt;&lt;/mark&gt;</code>	выделяет фрагменты текста желтым фоном	inline
<code>&lt;small&gt;&lt;/small&gt;</code>	отображает текст шрифтом меньшего размера	inline
<code>&lt;sub&gt;&lt;/sub&gt;</code>	подстрочное написание символов	inline
<code>&lt;sup&gt;&lt;/sup&gt;</code>	надстрочное написание символов	inline
<code>&lt;time&gt;&lt;/time&gt;</code>	дата / время документа или статьи	inline
<code>&lt;abbr&gt;&lt;/abbr&gt;</code>	аббревиатура или акроним	none
<code>&lt;address&gt;&lt;/address&gt;</code>	контактные данные автора документа или статьи	block
<code>&lt;bdi&gt;&lt;/bdi&gt;</code>	изолирует текст, читаемый справа налево	inline
<code>&lt;bdo&gt;&lt;/bdo&gt;</code>	задаёт направление написания текста	inline
<code>&lt;ruby&gt;&lt;/ruby&gt;</code>	контейнер для Восточно-Азиатских символов и их расшифровки	inline
<code>&lt;rp&gt;&lt;/rp&gt;</code>	тег для скобок вокруг символов	none
<code>&lt;rt&gt;&lt;/rt&gt;</code>	расшифровка символов	block

<code>&lt;figure&gt;&lt;/figure&gt;</code>	независимый контейнер для такого контента как изображения, диаграммы и т.п.	block
<code>&lt;figcaption&gt;&lt;/figcaption&gt;</code>	заголовок для элемента <code>&lt;figure&gt;</code>	block
<code>&lt;details&gt;&lt;/details&gt;</code>	контейнер с дополнительными сведениями, который можно открыть или закрыть	block
<code>&lt;summary&gt;&lt;/summary&gt;</code>	видимый заголовок для элемента <code>&lt;details&gt;</code>	block
<code>&lt;noscript&gt;&lt;/noscript&gt;</code>	секция, не поддерживающая скрипт	block

## Формы

<code>&lt;form&gt;&lt;/form&gt;</code>	html-форма	block
<code>&lt;input&gt;</code>	многофункциональные поля формы	inline-block
<code>&lt;label&gt;&lt;/label&gt;</code>	текстовая метка для элемента <code>&lt;input&gt;</code>	inline
<code>&lt;datalist&gt;&lt;/datalist&gt;</code>	контейнер для выпадающего списка элемента <code>&lt;input&gt;</code> , значения добавляются в <code>&lt;option&gt;</code>	none
<code>&lt;select&gt;&lt;/select&gt;</code>	элемент управления с выбором значений из предложенных вариантов <code>&lt;option&gt;</code>	inline-block
<code>&lt;optgroup&gt;&lt;/optgroup&gt;</code>	контейнер с заголовком для группы элементов <code>&lt;option&gt;</code>	block
<code>&lt;option&gt;&lt;/option&gt;</code>	вариант (опция) в раскрывающемся списке	block
<code>&lt;fieldset&gt;&lt;/fieldset&gt;</code>	группирует связанные элементы в форме	block
<code>&lt;legend&gt;&lt;/legend&gt;</code>	заголовок элементов формы, сгруппированных с помощью <code>&lt;fieldset&gt;</code>	block
<code>&lt;button&gt;&lt;/button&gt;</code>	интерактивная кнопка	inline-block
<code>&lt;keygen&gt;</code>	генерирует закрытый и открытый ключи для формы	inline-block
<code>&lt;progress&gt;&lt;/progress&gt;</code>	индикатор выполнения задачи	inline-block
<code>&lt;meter&gt;&lt;/meter&gt;</code>	индикатор измерения в заданном диапазоне	inline-block
<code>&lt;output&gt;&lt;/output&gt;</code>	поле для вывода результата вычисления	inline

## Встраиваемый контент

<code>&lt;audio&gt;&lt;/audio&gt;</code>	добавляет аудио-файлы	inline-block
<code>&lt;video&gt;&lt;/video&gt;</code>	добавляет видео-файлы	inline-block
<code>&lt;source&gt;</code>	указывает местоположение и тип альтернативных файлов для <code>&lt;audio&gt;</code> и <code>&lt;video&gt;</code>	none
<code>&lt;track&gt;</code>	субтитры для элементов <code>&lt;audio&gt;</code> и <code>&lt;video&gt;</code>	none
<code>&lt;embed&gt;</code>	встраивает внешний интерактивный контент или плагин	inline-block
<code>&lt;object&gt;&lt;/object&gt;</code>	контейнер для встраивания мультимедиа	inline-block
<code>&lt;param&gt;</code>	задаёт параметры для плагинов, встраиваемых с помощью элемента <code>&lt;object&gt;</code>	none
<code>&lt;iframe&gt;&lt;/iframe&gt;</code>	создаёт встроенный фрейм	block

HTML атрибуты сообщают браузеру, каким образом должен отображаться тот или иной элемент страницы. Атрибуты позволяют сделать более разнообразными внешний вид информации, добавляемой с помощью одинаковых тегов.

Значение атрибута заключается в кавычки "". Названия и значения атрибутов не чувствительны к регистру, но, тем не менее, рекомендуется набирать их в нижнем регистре.

Глобальные атрибуты, приведенные в таблице ниже, могут быть использованы для любого HTML-элемента, хотя некоторые из них могут не оказывать на элементы никакого влияния.

HTML текст представлен в спецификации тегами для форматирования и группировки текста. Теги представляют собой контейнеры для текста и не имеют визуального отображения.

Теги для форматирования текста несут смысловую нагрузку и обычно задают для текста, заключенного внутрь, стилевое оформление, например, выделяют текст жирным начертанием или отображают его шрифтом другого семейства (свойство font-family).

Грамотно отформатированный текст дает понять поисковым системам, какие слова несут важную смысловую нагрузку, по каким из них предпочтительно ранжировать веб-страницу в поисковой выдаче. Вся текстовая информация, отображаемая на сайте, размещается внутри тега <body>.

## Теги для HTML текста

1. Теги заголовков: <h1...h6>

2. Теги для форматирования текста: <b>, <em>, <i>, <small>, <strong>, <sub>, <sup>, <ins>, <del>

3. Теги для ввода «компьютерного» текста: <code>, <kbd>, <samp>, <var>, <pre>

4. Теги для оформления цитат и определений: <abbr>, <bdo>, <blockquote>, <q>, <cite>, <dfn>

5. Абзацы, средства переноса текста: <p>, <br>, <hr>

# Заголовок первого уровня

## Заголовок второго уровня

### Заголовок третьего уровня

#### Заголовок четвертого уровня

##### Заголовок пятого уровня

###### Заголовок шестого уровня

Заголовки являются важными элементами веб-страницы, они упорядочивают текст, формируя его визуальную структуру. Теги `<h1>...<h6>` должны использоваться только для выделения заголовков нового раздела или подраздела. При использовании заголовков необходимо учитывать их иерархию, т.е. за `<h1>` должен следовать `<h2>` и т.д. Также не допускается вложение других тегов в теги `<h1>...<h6>`.

### 1.1. Тег <h1>

Заголовок самого верхнего уровня, на странице рекомендуется использовать только один раз, по возможности частично дублируя заглавие страницы. Тег <h1> должен быть уникальным для каждой страницы сайта. Рекомендуется прописывать тег в начале статьи, используя ключевое слово в тексте заголовка. Размер шрифта в браузере равен 2em, верхний и нижний отступ по умолчанию 0.67em.

### 1.2. Тег <h2>

Им обозначаются подзаголовки тега <h1>. Размер шрифта в браузере равен 1.5em, верхний и нижний отступ по умолчанию 0.83em.

### 1.3. Тег <h3>

Показывает подзаголовки тега <h2>. Размер шрифта в браузере равен 1.17em, верхний и нижний отступ по умолчанию 1em.

#### 1.4. Теги <h4>, <h5>, <h6>

Обозначают подзаголовки четвёртого, пятого и шестого уровня. Размер шрифта в браузере равен 1em / 0.83em / 0.67em, верхний и нижний отступ по умолчанию 1.33em / 1.67em / 2.33em соответственно.

Для всех тегов доступны [глобальные атрибуты](#).

## Теги для форматирования текста

### 2.1. Тег <b>

Задаёт полужирное начертания шрифта. Выделяет текст без акцента на его важность.

Для тега доступны [глобальные атрибуты](#).

### 2.2. Тег <em>

Отображает шрифт курсивом, придавая тексту значимость.

Для тега доступны [глобальные атрибуты](#).

### 2.3. Тег <i>

Отображает шрифт курсивом.

Для тега доступны [глобальные атрибуты](#).

### 2.4. Тег <small>

Уменьшает размер шрифта на единицу по отношению к обычному тексту.

Для тега доступны [глобальные атрибуты](#).

## 2.5. Тег <strong>

Задаёт полужирное начертание шрифта, относится к тегам логической разметки, указывая браузеру на важность текста.

Для тега доступны [глобальные атрибуты](#).

## 2.6. Тег <sub>

Используется для создания нижних индексов. Сдвигает текст ниже уровня строки, уменьшая его размер.

Для тега доступны [глобальные атрибуты](#).

## 2.7. Тег <sup>

Используется для создания степеней. Сдвигает текст выше уровня строки, уменьшая его размер.

Для тега доступны [глобальные атрибуты](#).

## 2.8. Тег <ins>

Выделяет текст в новой версии документа, подчёркивая его.

Для тега доступны следующие атрибуты: cite, datetime.

## 2.9. Тег <del>

Перечёркивает текст. Используется для выделения текста, удаленного из документа.

Для тега доступны следующие атрибуты: cite, datetime.

## Теги для ввода «компьютерного» текста

### 3.1. Тег <code>

Служит для выделения фрагментов программного кода. Отображается моноширинным шрифтом.

Для тега доступны [глобальные атрибуты](#).

### 3.2. Тег <kbd>

Отмечает фрагмент как вводимый пользователем с клавиатуры. Отображается моноширинным шрифтом.

Для тега доступны [глобальные атрибуты](#).

### 3.3. Тег <samp>

Применяется для выделения результата, полученного в ходе выполнения программы. Отображается моноширинным шрифтом.

Для тега доступны [глобальные атрибуты](#).

### 3.4. Тег <var>

Выделяет имена переменных, отображая курсивом.

Для тега доступны [глобальные атрибуты](#).

## Теги для оформления цитат и определений

### 4.1. Тег <abbr>

Применяется для форматирования аббревиатур. Браузером обычно подчеркивается пунктирной линией. Расшифровка сокращения осуществляется с помощью атрибута title, она появляется при наведении курсора мыши на текст.

Для тега доступны [глобальные атрибуты](#).

### 4.3. Тег <bdo>

Используется для замещения текущего направления текста, т.е. текст в теге отображается зеркально.

Для тега доступен атрибут dir.

### 4.4. Тег <blockquote>

Выделяет цитаты внутри документа, выделяя его отступами и переносами строк.

Для тега доступен атрибут cite.

### 4.5. Тег <q>

Используется для выделения коротких цитат. Браузерами заключается в кавычки.

Для тега доступен атрибут cite.

### 4.6. Тег <cite>

Применяется для выделения цитат, названий произведений, сносок на другие документы.

Для тега доступны [глобальные атрибуты](#).

### 4.7. Тег <dfn>

Позволяет выделить текст как определение. Несмотря на наличие специального тега, рекомендуется выделять текст силами CSS.

## **Абзацы, средства переноса текста**

### **5.1. Тег <p>**

Разбивает текст на отдельные абзацы, отделяя друг от друга пустой строкой. Браузер автоматически добавляет верхний и нижний отступ, равный 1em, при этом отступы соседних абзацев «схлопываются».

Для тега доступны [глобальные атрибуты](#).

### **5.2. Тег <br>**

Переносит текст на следующую строку, создавая разрыв строки.

Для тега доступны [глобальные атрибуты](#).

### **5.3. Тег <hr>**

Используется для разделения контента на веб-странице. Отображается в виде горизонтальной линии.

Для тега доступны [глобальные атрибуты](#).

HTML ссылки создаются с помощью элементов `<a>`, `<area>` и `<link>`. Ссылки представляют собой связь между двумя ресурсами, одним из которых является текущий документ.

Ссылки можно поделить на две категории:

ссылки на внешние ресурсы — создаются с помощью тега `<link>` и используются для расширения возможностей текущего документа при обработке браузером;

гиперссылки — ссылки на другие ресурсы, которые пользователь может посетить или загрузить.

## Структура ссылки

Гиперссылки создаются с помощью парного тега `<a></a>`. Внутри тега помещается текст, который будет отображаться на веб-странице. Текст ссылки отображается в браузере с подчёркиванием, цвет шрифта — синий, при наведении на ссылку курсор мыши меняет вид.

Обязательным параметром тега `<a>` является атрибут `href`, который задает URL-адрес веб-страницы.

```
<a href="http://site.ru">текст ссылки</a>
```

Ссылка состоит из двух частей — указателя и адресной части.

Указатель ссылки представляет собой фрагмент текста или изображение, видимые для пользователя, адресная часть ссылки для пользователя не видна, она представляет собой адрес ресурса, к которому необходимо перейти.

Адресная часть ссылки состоит из URL. URL — единый указатель ресурса, который определяет местонахождение ресурса. При создании адресов для разделения слов между собой рекомендуется использовать дефис, а не символ подчёркивания. В общем виде URL имеющий следующий формат:

```
метод доступа://имя сервера:порт/путь
```

Метод доступа, или протокол, осуществляет обмен данными между рабочими станциями в разных сетях.

file обеспечивает чтение файла с локального диска:

`file:/gallery/pictures/summer.html`

http предоставляет доступ к веб-странице по протоколу HTTP:

`http://www.site.ru/`

https — специальная реализация протокола HTTP, использующая шифрование (как правило, SSL или TLS)

`https://www.site.ru/`

ftp осуществляет запрос к FTP-серверу на получение файла:

`ftp://pgu/directory/library`

mailto запускает сеанс почтовой связи с указанным адресатом и хостом:

`mailto:nika@gmail.com`

Имя сервера описывает полное имя машины в сети, например, `www.site.ru`. Если имя сервера не указано, то ссылка считается локальной, т.е. она относится к той же машине, на которой находится html-документ, содержащий ссылку.

Номер порта TCP, на котором функционирует веб-сервер. Представляет собой число, которое необходимо указывать, если метод требует номер порта (отдельные сервера могут иметь свой отличительный номер порта). Если порт не указан, по умолчанию используется порт 80. Стандартными портами являются:

21 — FTP

23 — Telnet

70 — Gopher

80 — HTTP

Путь содержит имя папки, в которой находится файл.

## Абсолютные и относительные ссылки

Абсолютные ссылки содержат всю информацию, необходимую браузеру для нахождения файла. Абсолютный путь содержит следующие компоненты:

- 1) протокол (например, http)
- 2) домен (доменное имя или IP-адрес компьютера)
- 3) папка (имя папки, указывающей путь к файлу)
- 4) файл (имя файла)

<http://www.site.ru/папка/file.html>

Если файл находится в корневой папке, то путь к файлу будет отсутствовать:

<http://www.site.ru/file.html>

При отсутствии имени файла будет загружаться веб-страница, которая задана по умолчанию в настройках веб-сервера. Наличие завершающего слеша / означает, что обращение идет к папке, если его нет — напрямую к файлу.

<http://www.site.ru/http://www.site.ru/папка/>

Относительные ссылки описывают путь к указанному документу относительно текущего.

Путь определяется с учётом местоположения веб-страницы, на которой находится ссылка.

Относительные ссылки используются при создании ссылок на другие документы на одном и том же сайте. Когда браузер не находит в ссылке протокол `http://`, он выполняет поиск указанного документа на том же сервере.

Относительный путь содержит следующие компоненты:

- 1) папка (имя папки, указывающей путь к файлу)
- 2) файл (имя файла)

Путь для относительных ссылок имеет три специальных обозначения:

/ указывает на корневую директорию и говорит о том, что нужно начать путь от корневого каталога документов и идти вниз до следующей папки

./ указывает на текущую папку

../ подняться на одну папку (директорию) выше

Главное отличие относительного пути от абсолютного в том, что относительный путь не содержит имени корневой папки и родительских папок, что делает адрес короче, и в случае переезда с одного домена на другой не нужно прописывать новый абсолютный адрес. Но если сторонний ресурс будет ссылаться например, на ваши изображения с относительными адресами, то они не будут отображаться на другом сайте.

Якоря, или внутренние ссылки, создают переходы на различные разделы текущей веб-страницы, позволяя быстро перемещаться между разделами. Это оказывается очень удобным в случае, когда на странице слишком много текста. Внутренние ссылки также создаются при помощи тега `<a>` с разницей в том, что атрибут `href` содержит имя указателя — так называемый якорь, а не URL-адрес. Перед именем указателя всегда ставится знак `#`.

Следующая разметка создаст оглавление с быстрыми переходами на соответствующие разделы:

```
1 <h1>Времена года</h1>
2 <h2>Оглавление</h2>
3 <a href="#p1">Лето</a> <!--создаём якорь, указав #id элемента-->
4 <a href="#p2">Осень</a>
5 <a href="#p3">Зима</a>
6 <a href="#p4">Весна</a>
7 <p id="p1">...</p> <!--добавляем соответствующий id элементу-->
8 <p id="p2">...</p>
9 <p id="p3">...</p>
10 <p id="p4">...</p>
```

Если нужно сделать ссылку с **одной страницы** сайта на определенный раздел **другой страницы**, то необходимо задать `id` для этого раздела страницы, а затем добавить его к абсолютному адресу ссылки:

```
1 <th id="about-color">color</th>
2 <a href="http://html5book.ru/css-shrifty/#about-color" class="site" target="_blank">color</a>
```

## Как сделать изображение-ссылку

Чтобы сделать кликабельное изображение, необходимо поместить элемент `<img>` внутри тега `<a>`. Чтобы ссылка открывалась в другом окне, нужно добавить атрибут `target="_blank"` для ссылки.

```
1 <a href="http://www.fast-torrent.ru/film/gran-za-granyu-tv.html" target="_blank"></a>
```



## Как сделать ссылку на телефонный номер, скайп или адрес электронной почты

У ссылок появились новые возможности — по клику можно не только переходить на другие страницы и скачивать файлы, но и совершать звонки на телефоны, отправлять сообщения или звонить по скайпу.

```
1 ссылка на телефонный номер
2 <a href="tel:+74951234567">+7 (495) 123-45-67</a>
3
4 ссылка на адрес электронной почты
5 <a href="mailto:example@mail.ru">example@mail.ru</a>
6
7 ссылка на скайп (позвонить)
8 <a href="skype:имя-пользователя?call">Skype</a>
9
10 ссылка на скайп (открыть чат)
11 <a href="skype:имя-пользователя?chat">Skype</a>
12
13 ссылка на скайп (добавить в список контактов)
14 <a href="skype:имя-пользователя?add">Skype</a>
15
16 ссылка на скайп (отправить файл)
17 <a href="skype:имя-пользователя?sendfile">Skype</a>
```

## Тег <img>

Изображения добавляются на веб-страницы при помощи одинарного тега <img>, закрывающий тег не нужен. Так как элемент <img> является строчным, то рекомендуется располагать его внутри блочного элемента, например, <p> или <div>. Тег <img> имеет обязательный атрибут src, значением которого является адрес встраиваемого изображения, и рекомендуемый атрибут alt, например:



## Тег <map>

Тег <map> служит для представления графического изображения в виде карты с активными областями. Активные области определяются по изменению вида курсора мыши при наведении. Щелкая мышью на активных областях, пользователь может переходить к связанным документам.

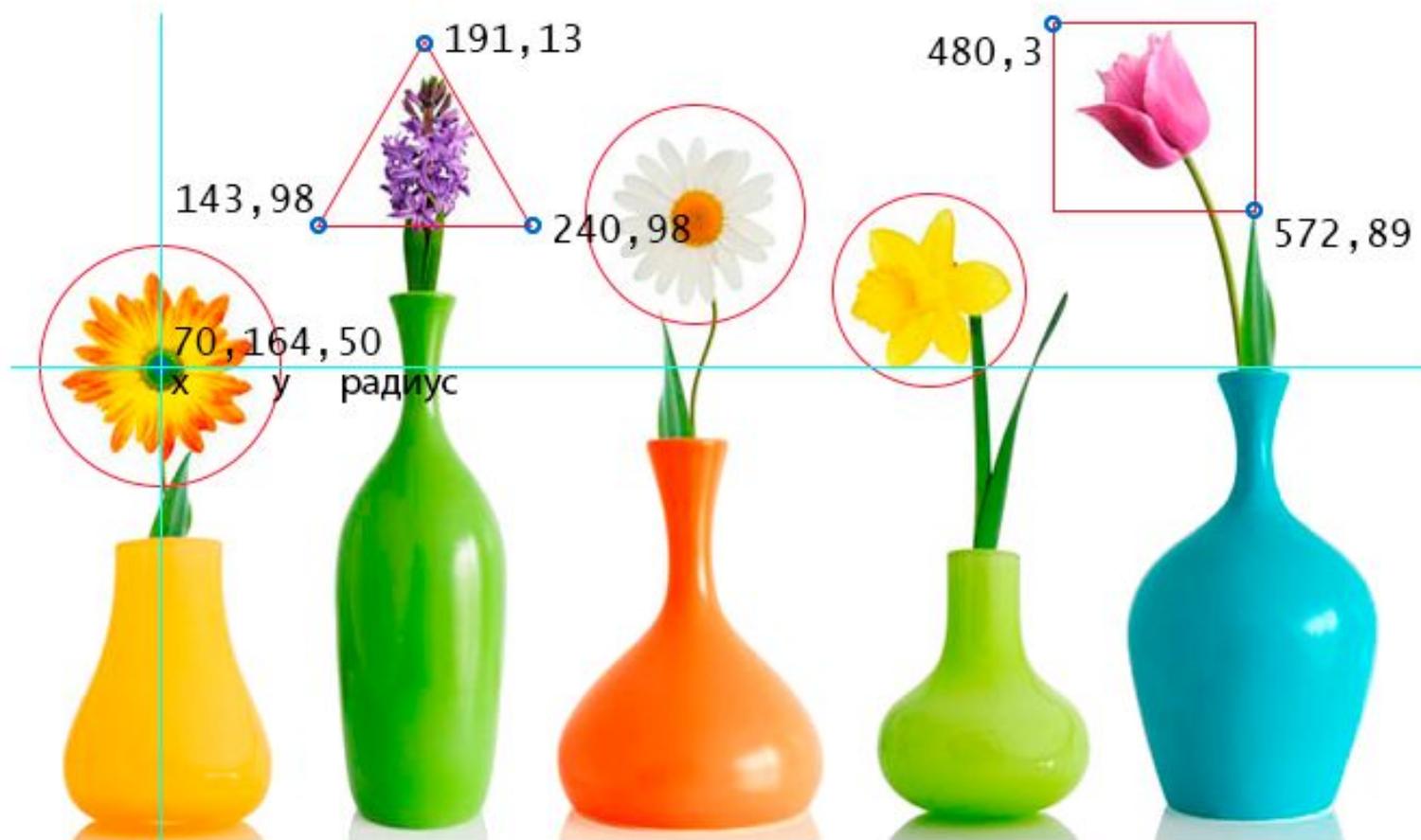
Для тега доступен атрибут name, который задает имя карты. Значение атрибута name для тега <map> должно соответствовать имени в атрибуте usemap элемента <img>:

```
1 
2 <map name="имя_карты">
3   ...
4 </map>
```

Элемент `<map>` содержит ряд элементов `<area>`, определяющих интерактивные области в изображении карты.

## Пример создания карты-изображения

1) Размечаем исходное изображение на активные области нужной формы. Координаты областей можно вычислить с помощью программы для обработки фотографий, например, **Adobe Photoshop** или **Paint**.



2) Задаем имя карты, добавив его в тег <map> с помощью атрибута name. Это же значение присваиваем атрибуту usemap тега <img>.

3) Добавляем ссылки на веб-страницы или части веб-документа для каждой активной области, по которым пользователь будет переходить при нажатии курсором мыши на активную область изображения.

```
1 
2 <map name="flowers">
3 <area shape="circle" coords="70,164,50" href="https://ru.wikipedia.org/wiki/Гербера" alt="gerbe
ra" target="_blank"> <!--задаем активную область в виде круга и указываем координаты ее
центра и радиус-->
4 <area shape="poly" coords="191,13,240,98,143,98,191,13" href="https://ru.wikipedia.org/wiki/%C3%E
8%E0%F6%E8%ED%F2" alt="hyacinth" target="_blank"> <!--задаем активную область в виде треугол
ьника и указываем координаты его вершин-->
5 <area shape="circle" coords="318,93,50" href="https://ru.wikipedia.org/wiki/Ромашка" alt="самом
илес" target="_blank">
6 <area shape="circle" coords="425,129,45" href="https://ru.wikipedia.org/wiki/Нарцисс_(растени
е)" alt="narcissus" target="_blank">
7 <area shape="rect" coords="480,3,572,89" href="https://ru.wikipedia.org/wiki/Тюльпан" alt="tulip"
target="_blank"> <!--задаем активную область в виде прямоугольника и указываем координ
аты верхнего левого и нижнего правого углов-->
8 </map>
```

# 1. Как создать таблицу

Таблица создаётся при помощи парного тега `<table></table>`. Данный тег является контейнером для элементов таблицы и все элементы должны находиться внутри него. Например, с помощью данной разметки можно создать таблицу, состоящую из **двух столбцов и двух строк**:

```
1 <table>
2 <tr><th>текст заголовка</th><th>текст заголовка</th></tr> <!--ряд с ячейками заголовков-->
3 <tr><td>данные</td><td>данные</td></tr> <!--ряд с ячейками тела таблицы-->
4 </table>
```

**текст заголовка текст заголовка**

данные

данные

По умолчанию таблица и ячейки не имеют видимых границ. **Границы** задаются с помощью свойства `border`:

```
1 /* внешние границы таблицы серого цвета толщиной 1px */
2 table {border: 1px solid grey;}
3 /* границы ячеек первого ряда таблицы */
4 th {border: 1px solid grey;}
5 /* границы ячеек тела таблицы */
6 td {border: 1px solid grey;}
```

**Промежутки между ячейками таблицы** убираются с помощью свойства `table` `{border-collapse: collapse;}`.

**Ширина** таблицы по умолчанию равна ширине её внутреннего содержимого. Чтобы установить ширину, нужно задать значение для свойства `width`:

```
1 /* сделает ширину таблицы равной ширине блока контейнера, в котором она находится */
2 table {width: 100%;}
3 /* задаст фиксированную ширину для таблицы */
4 table {width: 600px;}
```

Если для ячеек таблицы заданы внутренние отступы и границы, то ширина таблицы будет включать в себя следующие значения: padding-left и padding-right, ширина border-left плюс ширина border-right последней ячейки в ряду. Если заданы ширина и границы ячеек, то ширина таблицы будет складываться из ширины ячеек плюс ширина border-left и ширина border-right последней ячейки в ряду.

## **2. Как создать строки (ряды) таблицы**

Строки или ряды таблицы создаются с помощью тега `<tr>`. Количество горизонтальных строк таблицы определяется количеством парных тегов `<tr></tr>`.

### 3. Как сделать ячейку заголовка столбца таблицы

Элемент `<th>` создаёт заголовок столбца – специальную ячейку, текст в которой выделяется полужирным. Количество ячеек заголовка определяется количеством пар тегов `<th></th>`. Для элемента доступны атрибуты `colspan`, `rowspan`, `headers`.

```
1 <table>
2 <tr><th>ячейка заголовка</th><th>ячейка заголовка</th></tr>
3 </table>
```

## 4. Как сделать ячейку тела таблицы

Элемент `<td>` создаёт ячейки таблицы, внутрь которых помещаются данные таблицы. Парные теги `<td></td>`, расположенные в одном ряду, определяют количество ячеек в строке таблицы. Количество пар ячеек `<td>` должно быть равно количеству пар ячеек `<th>`. Для элемента доступны атрибуты `colspan`, `rowspan`, `headers`.

```
1 <table>
2 <tr><th>ячейка заголовка</th><th>ячейка заголовка</th></tr>
3 <tr><td>ячейка тела таблицы</td><td>ячейка тела таблицы</td></tr>
4 </table>
```

## 5. Как добавить подпись (заголовок) к таблице

Элемент `<caption>` создает подпись таблицы. Добавляется непосредственно после тега `<table>`, вне строки или ячейки.

```
1 <table>
2 <caption>Перечень продуктов</caption>
3   <tr>
4     <th>№ п/п</th>
5     <th>Наименование товара</th>
6     <th>Ед. изм.</th>
7     <th>Количество</th>
8     <th>Цена за ед. изм., руб.</th>
9     <th>Стоимость, руб.</th>
10  </tr>
11  <tr>
12    <td>1.</td>
13    <td>Томаты свежие</td><td>кг</td><td>15,20</td><td>69,00</td><td>1048,80</td>
14  </tr>
15  <tr>
16    <td>2.</td>
17    <td>Огурцы свежие</td><td>кг</td><td>2,50</td><td>48,00</td><td>120,00</td>
18  </tr>
19  <tr>
20    <td colspan="5" style="text-align:right">ИТОГО:</td><td>1168,80</td>
21  </tr>
22 </table>
```

## **6. Группирование строк и столбцов таблицы**

Элемент `<colgroup>` создает структурную группу столбцов, выделяя логически однородные ячейки. Группирует один или более столбцов для единого форматирования, позволяя применить стили к столбцам вместо того, чтобы повторять стили для каждой ячейки и для каждой строки. Добавляется непосредственно после тегов `<table>` и `<caption>`.

Элемент `<col>` формирует группы столбцов, которые делят таблицу на разделы, не относящиеся к общей структуре, т.е. не содержащие информацию одного типа. Позволяет задавать свойства столбцов для каждого столбца в пределах элемента `<colgroup>`. С помощью атрибута `<style>` можно изменить основной цвет фона ячеек. Для элемента `<col>` доступен атрибут `span`, задающий количество столбцов для объединения.

```

1 <table>
2   <colgroup>
3     <col span="2" style="background:Khaki"><!-- С помощью этой конструкции задаем цвет ф
она для первых двух столбцов таблицы-->
4     <col style="background-color:LightCyan"><!-- Задаем цвет фона для следующего (одного)
столбца таблицы-->
5   </colgroup>
6   <tr>
7     <th>№ п/п</th>
8     <th>Наименование</th>
9     <th>Цена, руб.</th>
10  </tr>
11  <tr>
12    <td>1</td>
13    <td>Карандаш цветной</td>
14    <td>20,00</td>
15  </tr>
16  <tr>
17    <td>2</td>
18    <td>Линейка 20 см</td>
19    <td>30,00</td>
20  </tr>
21 </table>

```

№ п/п	Наименование	Цена, руб.
1	Карандаш цветной	20,00
2	Линейка 20см	30,00

## 7. Группировка разделов таблицы

Элемент `<thead>` создает группу заголовков для строк таблицы с целью задания единого оформления. Используется в сочетании с элементами `<tbody>` и `<tfoot>` для указания каждой части таблицы.

Элемент должен быть использован в следующем порядке: как дочерний элемент `<table>`, после `<caption>` и `<colgroup>`, и перед `<tbody>`, `<tfoot>` и `<tr>` элементами. В пределах одной таблицы можно использовать один раз.

Элемент `<tbody>` группирует основное содержимое таблицы. Используется в сочетании с элементами `<thead>` и `<tfoot>`.

Элемент `<tfoot>` создает группу строк для представления информации о суммах или итогах, расположенную в нижней части таблицы. Используется в таблице один раз. Располагается после тега `<thead>`, перед тегам `<tbody>` и `<tr>`.

```

1 <table>
2 <thead>
3 <tr>
4 <th>№ п/п</th>
5 <th>Наименование товара</th>
6 <th>Ед. изм.</th>
7 <th>Количество</th>
8 <th>Цена за ед. изм., руб.</th>
9 <th>Стоимость, руб.</th>
10 </tr>
11 </thead>
12 <tfoot>
13 <tr>
14 <td colspan="5" style="text-align:right">ИТОГО:</td><td>1168,80</td>
15 </tr>
16 </tfoot>
17 <tbody>
18 <tr>
19 <td>1.</td>
20 <td>Томаты свежие</td><td>кг</td><td>15,20</td><td>69,00</td><td>1048,80</td>
21 </tr>
22 <tr>
23 <td>2.</td>
24 <td>Огурцы свежие</td><td>кг</td><td>2,50</td><td>48,00</td><td>120,00</td>
25 </tr>
26 </tbody>
27 </table>

```

## 8. Как объединить ячейки таблицы

Атрибуты `colspan` и `rowspan` объединяют ячейки таблицы. Атрибут `colspan` задает количество ячеек, объединенных по горизонтали, а `rowspan` — по вертикали.

```
1 <table>
2   <tr>
3     <th>№ п/п</th>
4     <th>Наименование товара</th>
5     <th>Ед. изм.</th>
6     <th>Количество</th>
7     <th>Цена за ед. изм., руб.</th>
8     <th>Стоимость, руб.</th>
9   </tr>
10  <tr>
11    <td>1.</td>
12    <td>Томаты свежие</td><td>кг</td><td>15,20</td><td>69,00</td><td>1048,80</td>
13  </tr>
14  <tr>
15    <td>2.</td>
16    <td>Огурцы свежие</td><td>кг</td><td>2,50</td><td>48,00</td><td>120,00</td>
17  </tr>
18  <tr>
19    <td colspan="5" style="text-align:right">ИТОГО:</td><td>1168,80</td><!-- Задаем количес
20    </tr>
21 </table>
```

№ п/п	Наименование товара	Ед. изм.	Количество	Цена за ед. изм., руб.	Стоимость, руб.
1.	Томаты свежие	кг	15,20	69,00	1048,80
2.	Огурцы свежие	кг	2,50	48,00	120,00
ИТОГО:					1168,80