

Мастер-класс по созданию сайта-визитки.

Первый сайт в мире появился в интернете и был доступен онлайн **6 августа 1991 года**. Его создатель – **Тим Бернерс-Ли**, британский ученый, сотрудник Европейского Центра по ядерным исследованиям (CERN). Первый браузер, который также создал Тим Бернерс-Ли поддерживал только текстовую информацию. Прошло много лет, стандарты и возможности языка HTML поменялись.

Начнем с того, что сайт любой состоит из двух частей:
то, что мы видим перед собой, например, текст, картинки, анимацию и так далее; и то, что скрыто от наших глаз. По-другому это называется: Front-end и Back-end



Фронтенд — это все, что браузер может читать, выводить на экран и запускать.

Бэкенд — это набор средств, с помощью которых происходит реализация логики веб-сайта. Начинка сайта, скрытая от пользователя.

Пример: как только вы введете в поисковой строке какой-то запрос и нажмете enter, то работа переходит из фронтенда в бэкенд и там уже поисковые алгоритмы выбирают нужную вам информацию и как только найдут, снова роль за фронтендом, ведь эту информацию нужно вывести на экран.

Для создания сайтов у нас есть два помощника: HTML и CSS.

HTML (HyperText Markup Language — «язык гипертекстовой разметки») – нужен для структурирования и отображения веб-страницы и ее контента.


! HTML не является языком программирования.

CSS (Cascading Style Sheets – «каскадные таблицы стилей») – используется для стилизации веб-страницы.

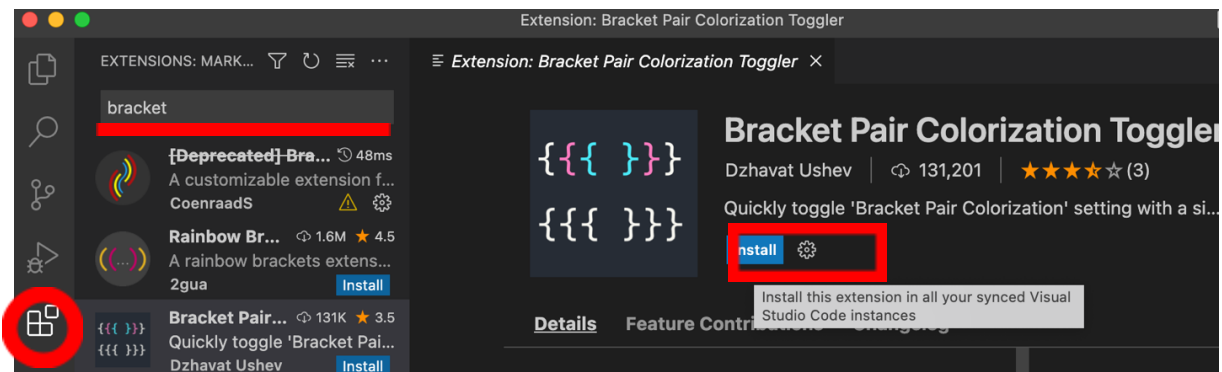
Часть 1. Работа в Visual Studio Code.

Работать мы будем в Visual Studio Code, скачать который можно по ссылке:
<https://code.visualstudio.com/>

Так же для удобства работы можно установить некоторые плагины:

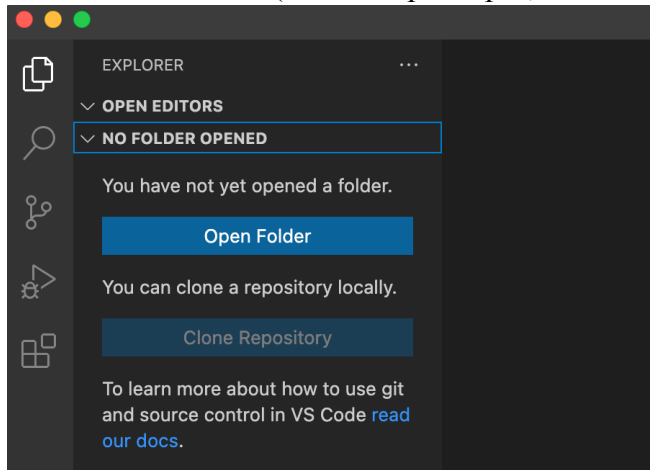
Нажмите на  значок (Extensions). В строке поиска введите название расширений (указаны ниже) и нажмите Install. (Для удобства выделено красным на рисунке).

- Auto Complete Tag
- Live Server
- Bracket pair colorization Toggler
- Indent-rainbow



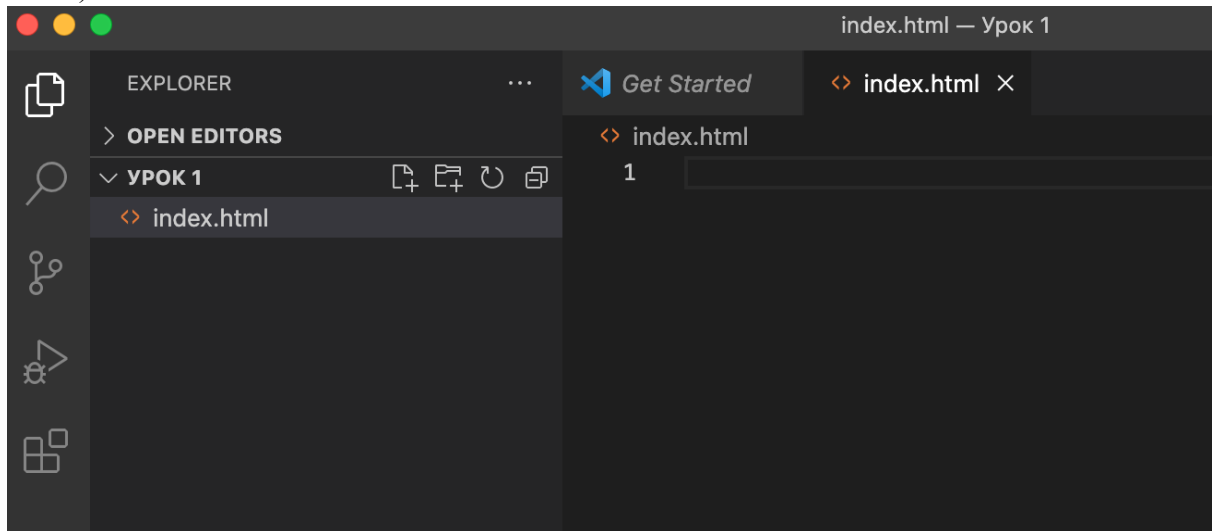
Как создать HTML-файл?

- 1) Создайте на рабочем столе папку, в которой будет храниться все файлы с нашего курса. Назовите ее «Мастер-класс по созданию сайта».
- 2) Заходим в VS Code (если в первый раз, то нажимаем «Open Folder».)



- 3) Далее выбираем папку «Мастер-класс по созданию сайта», в которой и создадим файл «index.html» (нажмем на значок , напишем имя файла и нажмем

Enter.)



- 4) Справа видим поле для ввода кода. Вводим «!» и нажимаем TAB. У вас должен появиться базовый код html-страницы.
- 5) Перейдя в папку с уроком откройте в браузере ваш файл index.html.

Задание 1. Придумайте тему и примерный контент для вашего сайта. (Учитывайте, что ваш сайт будет содержать не менее 5-ти страниц, которые должны быть связаны между собой общим меню.)

Задание 2. Создайте еще 4 html-файла, дайте им осмысленные названия.

Часть 2. Работа со структурой HTML-документа.

Любой HTML-документ состоит из тегов.

Тег – это HTML-элемент, состоящий из имени элемента, окруженного < >.



Существуют теги двойные и одинарные.

Структура HTML документа - скелет, на основе которого строится вся страница, выглядит вот так:

```
<!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>
```

Вот тут-то и будет то, что мы хотим видеть в браузере!

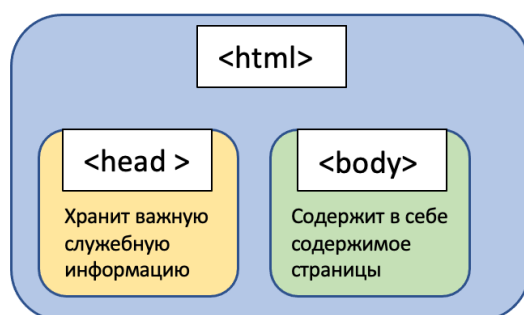
```
</body>  
</html>
```

! Обратите внимание на отступы (они позволяют видеть структуру документа)

Каждый HTML-документ начинается с декларации типа документа, чтобы браузер мог определить версию HTML и мог правильно отобразить страницу. (`<!DOCTYPE html>`)

Простейшая HTML-страница состоит, как минимум из 3 тегов: `<html>`, `<head>`, `<body>`.
Тег `<html>` идет сразу после доктайпа и содержит в себе теги `<head>` и `<body>`.

!В документе теги `<head>` и `<body>` могут быть использованы только один раз.



Тег `<head>` отвечает за важную служебную информацию. И содержит в себе тег `<title>`.

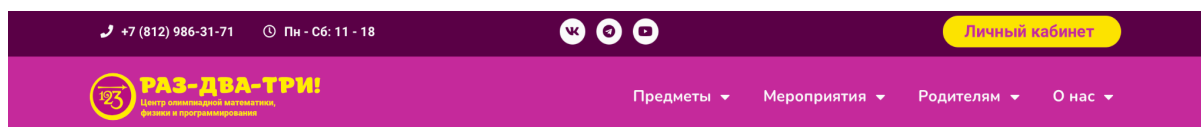
Тег `<title>` определяет заголовок документа (`<title>Document</title>`).

Тег `<body>` хранит в себе содержимое страницы, которое отображается в окне браузера. Все тексты и картинки мы будем добавлять внутрь `<body>`.

Тело страницы обычно разбито на некоторые логические блоки: шапка сайта, уникальное содержимое страницы, подвал сайта. Для того, чтобы это указать, нужно использовать специальные теги:

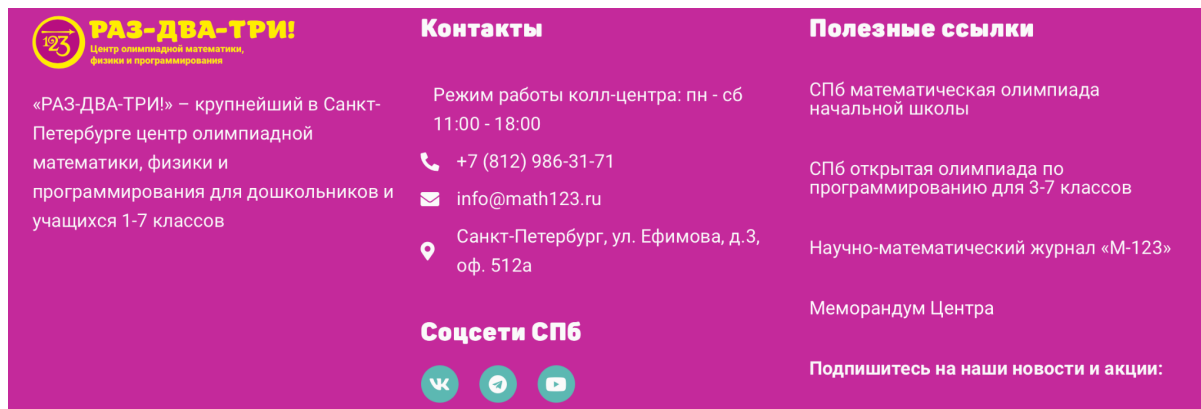
`<main>` - выделяет основное содержание страницы, которое не повторяется на других страницах.

`<header>` - вводная часть страницы, которую чаще называют “шапкой”. Предназначена для облегчения навигации по странице, первый элемент, который привлекает внимание пользователя. С помощью хедера посетитель сайта понимает, где находится, как перейти в следующий раздел и найти нужную информацию. В интернет-магазинах в шапке сайта находится один из главных элементов - форма для оформления заказа, вход в личный кабинет и корзина. Пример::



<footer> - заключительная часть страницы, которую чаще называют “подвалом”.

Предназначена для логического завершения страницы, облегчения навигации и размещения дополнительных страниц. Пример:



Задание 1. Поменяйте заголовок вашей страницы на тему вашего сайта.

Задание 2. Добавьте логические разделы на ваш сайт. (header, main, footer)

Часть 3. Содержимое страницы.

С помощью HTML него мы может задать структуру и смысл всему контенту на нашей странице.

Заголовки и абзацы:

Для создания основной структуры текста используют заголовки. В HTML существует целое семейство заголовочных тегов: от <h1> до <h6>. Тег <h1> обозначает самый важный заголовок (заголовок верхнего уровня, его также добавляют в шапку на главных страницах), а тег <h6> обозначает подзаголовок самого нижнего уровня.

Более мелкую структуру выстраивают с помощью параграфов.

По умолчанию абзацы начинаются с новой строки и отделяются от остального контента отступами сверху и снизу. За это отвечает тег <p></p>.

Тег
 отвечает за перенос строки

Теги для форматирования текста:

 - используется для выделения текста жирным шрифтом, тег физической разметки.

 - используется для выделения текста жирным шрифтом, следует использовать для выделения важных фрагментов текста по смыслу, тег логической разметки.

<small> - используется для того, чтобы напечатать текст меньшего размера.

<s> - используется для того, чтобы выделить текст, как неактуальный (зачеркивает его).
<i> - используется для выделения текста курсивом, тег физической разметки.
 - используется для выделения текста курсивом, акцентируя внимание на тексте, тег логической разметки.
<sup> - используется для написания текста в виде верхнего индекса.
<sub> - используется для написания текста в виде нижнего индекса.
<pre> - используется для вывода текста с сохранением формата, в котором текст был написан в текстовом редакторе.
<mark> - используется для подсветки фрагмента.
<dfn> - указывает, что текст внутри тега является термином.
<blockquote> - указывает, что текст внутри тега является цитатой.

HTML-элементы могут быть вложены друг в друга.

Элементы делятся на два вида:

- Блочные
- Строчные

Элементы блочного типа: <p>, <h1>....

Элементы строчного типа: , , <small>, <a>

Задание 1. Создайте несколько блоков текстовой информации на вашей главной странице.

Задание 2. Примените теги для форматирования текста (не меньше 5).

Часть 4. Добавление ссылок и изображений.

Атрибут тега - необходим для того, чтобы расширить возможности отдельных тегов.

Атрибуты указываются внутри тегов. Они состоят из имени и значения.

Пример:

<p align="left"> Выровнено по левому краю.</p>

align - это имя атрибута, a left - его значение! Обратите внимание, что значение атрибута берется в кавычки.

Добавление ссылок:

Абсолютная ссылка — это ссылка, в которой прописан адрес целиком. (Обычно данные ссылки ведут на сторонний ресурс.) Для создания такой ссылки используется тег <a> с атрибутом href="абсолютный адрес"

Выглядит это так: Подробнее тут

Добавление изображений

Добавление изображений делается с помощью одиночного тега и атрибута src. Между кавычками вставляется адрес изображения.

Для того, чтобы скорректировать размеры, нужно использовать атрибуты width и height

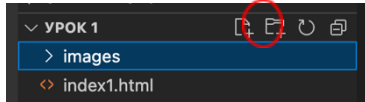
```

```

Файлы (изображения) можно размещать:

- в одной папке с html-файлом
- если на удаленном сервере, то обращение происходит по ссылке

Для картинок принято создавать отдельную папку в проекте с названием `images`. И сохранять там все картинки, которые используются в вашем проекте.



Если картинка находится в папке, то обращение к ней происходит следующим образом:

```

```

`img` – имя папки, в которой лежат картинки

`image` – название картинки

Для создания картинки ссылки тег `<a>` окружает тег ``



Задание 1. Вставьте в ваш документ на главную страницу ссылки на другие ваши страницы. А на других страницах вставьте ссылку на главную.

Задание 2. Вставьте по 1-ому изображению на каждую вашу страницу.

Часть 5. Добавление списков.

Списки - упорядоченный набор значений, который помогает нам систематизировать информацию, делая ее более доступной для понимания.

Списки:

- Маркированные
- Нумерованные

За создание маркированного списка отвечает тег ``, за создание пунктов в этом списке тег ``

```
<ul>
  <li>Елки</li>
  <li>Пихты</li>
  <li>Ели</li>
</ul>
```

- Елки
- Пихты
- Ели

За создание нумерованного списка отвечает тег ``, за создание пунктов в этом списке тег ``

```
<ol>
  <li>Елки</li>
  <li>Пихты</li>
  <li>Ели</li>
</ol>
```

1. Елки
2. Пихты
3. Ели

Задание 1. Создайте минимум 1 список, в котором вы укажете структуру вашего сайта, его разделы.

Часть 6. Создание меню.

Меню создается в шапке сайта. С использованием тега `<nav>`.

Тег `<nav>` - применяется для связей, которые поддерживают начальную навигацию по сайту. Список ссылок для перехода создается при помощи тегов `<a>` и ``.

Пример:

```
<nav>
  <ul>
    <li>Главная</li>
    <li><a href="film1.html">Фильм 1</a></li>
    <li><a href="film2.html">Фильм 2</a></li>
    <li><a href="film3.html">Фильм 3</a></li>
    <li><a href="film4.html">Фильм 4</a></li>
    <li><a href="film5.html">Фильм 5</a></li>
  </ul>
</nav>
```

Задание 1. Создайте меню вашего сайта. (объедините ссылки с созданным списком на главной странице). Создание меню выполняется в шапке вашего сайта.

Часть 7. Подключение стилей.

Подключение стилей

CSS-правило состоит из селектора и перечня свойств и их значений:

```
Селектор
{
  свойство: значение;
  свойство: значение;
}
```


Селектор находится в начале CSS-правила, до фигурных скобок, и определяет, к каким HTML-элементам применяется свойства и значения из правила.

Простейшие (и самые популярные) селекторы — это селекторы по тегам, они содержат имя тега без символов и применяются ко всем подходящим тегам.

```
h1 { color: red; }
```


На странице может быть несколько списков, и стили применяются ко всем спискам, даже к тем, которые вы менять не хотели.

Как подключить стили из файла?

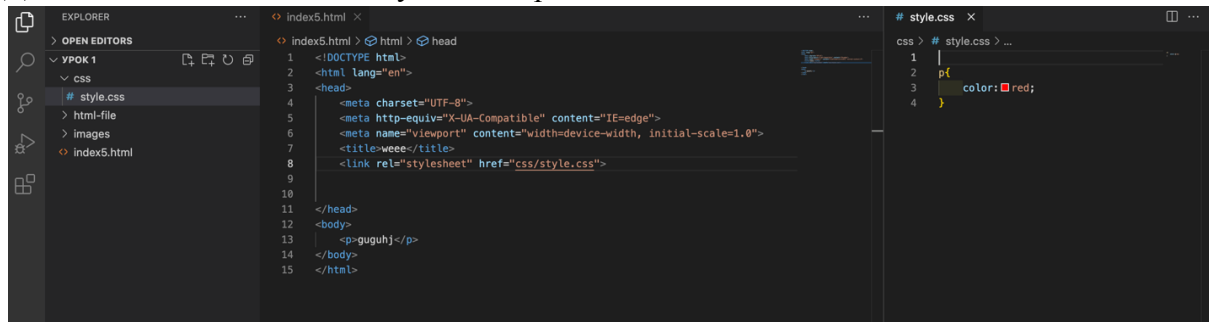
В папке с проектом создается подкаталог «css», где хранятся файлы с расширением .css , вы можете поступить таким же образом, либо в папке с вашим проектом создать файл «style.css». Осталось связать его с нашим файлом.

Указываем путь следующим образом:

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

Так же для удобства можно разделить экран на css и html файл (в верхнем правом углу )

Данный метод часто используется на практике.



Типы селекторов:

Универсальный селектор – универсальный селектор соответствует элементам любого типа

```
* {  
  padding: 0;  
  margin: 0;  
}
```

Обычно в начале файла вставляют такой код, чтобы избавиться от лишних отступов.

Селектор элемента:

Селектор элемента соответствует всем элементам указанного тега:

```
p {  
  color: red;  
}
```

Селектор ID (идентификатора)

Определяет уникальное имя элемента, которое используется для изменения его стиля и обращение к нему.

! Идентификатор может встретиться в коде только один раз

Селектор ID выбирает те элементы, у которых указан соответствующий атрибут ID

Чтобы обратиться к элементу по ID нужно добавить в html элементу `id="название_id"` в CSS используется обращение по хештегу `#название_id` (мы уже встречали это в якорных ссылках, для этого он в основном и используется)

Каждый элемент в документе (блочный или строчный) - генерирует прямоугольный блок элемента. У любого элемента имеются внешние и внутренние отступы.

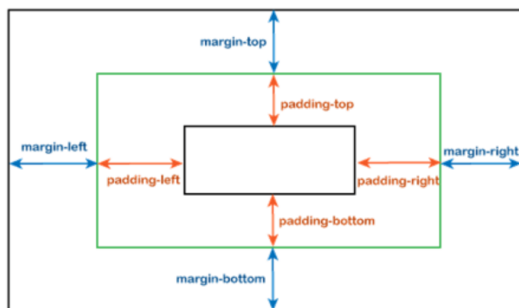
Регулировать длину и ширину контента можно с помощью свойств:

`width` - ширина

`height` - высота

Внешние и внутренние отступы для элемента можно регулировать:

- свойство `margin` - определяет внешние отступы элемента
- свойство `padding` - определяет внутренние отступы элемента



Для того, чтобы регулировать длину и ширину блока, которая включает в себя отступы, контент и границы блока, нужно использовать свойство:

`box-sizing: border-box;`

Для того, чтобы применить свойства `width` и `height` к строчным элементам, нужно применить свойство:

`display: inline-block;`

Свойство `float` - определяет, по какой стороне будет выровнен элемент. Значения: `right`, `left`. (Используется, например, для выравнивания изображений).

Свойства для стилизации текста, для работы с цветом:

| | |
|--------------------------|--|
| <code>font-family</code> | <p>Устанавливает семейство шрифта, которое будет использоваться для оформления текста содержимого.</p> <p>Значения:</p> <ul style="list-style-type: none">• <code>serif</code> — шрифты с засечками (антиквенные), типа Times;• <code>sans-serif</code> — рубленые шрифты (шрифты без засечек или гротески), типичный представитель — Arial;• <code>cursive</code> — курсивные шрифты;• <code>fantasy</code> — декоративные шрифты;• <code>monospace</code> — моноширинные шрифты, ширина каждого символа в таком семействе одинакова (шрифт Courier). |
|--------------------------|--|

| | |
|-----------------|--|
| | <p>Например,</p> <pre>p { font-family: Courier, monospace; }</pre> |
| font-size | <p>Определяет размер шрифта элемента. Размер можно указывать в px.</p> <p>Значения:</p> <ul style="list-style-type: none"> • xx-small • x-small • small • medium • large • x-large • xx-large • smaller • larger <p>Например,</p> <pre>p { font-size: larger; } h1 { font-size: 20px; }</pre> |
| font-weight | <p>Устанавливает насыщенность шрифта (жирность).</p> <p>Значения:</p> <ul style="list-style-type: none"> • bold • bolder • lighter • 100 • 200 • 900 <p>Например,</p> <pre>p { font-weight: bold; }</pre> |
| text-decoration | <p>Добавляет оформление текста в виде его подчеркивания, перечеркивания, линии над текстом и мигания. Одновременно можно применить более одного стиля, перечисляя значения через пробел.</p> <p>Значения:</p> <ul style="list-style-type: none"> • blink – мигающий текст • line-through – перечеркнутый текст • overline – сплошная линия над текстом • underline – сплошная линия под текстом • none – отменяет все эффекты (полезно у ссылок) <p>Например,</p> <pre>h1</pre> |

	<pre>{ font-decoration: underline; }</pre>
color	<p>Определяет цвет текста элемента.</p> <p>Например,</p> <pre>h1 { color: red; }</pre>
text-transform	<p>Управляет преобразованием текста элемента в заглавные или прописные символы</p> <p>Значения:</p> <p>none</p> <p>capitalize - делает заглавной первую букву каждого слова.</p> <p>lowercase - все буквы становятся строчными.</p> <p>uppercase - все символы текста становятся прописными</p> <p>Например,</p> <pre>h1 { text-transform: lowercase; }</pre>
text-shadow	<p>Добавляет тень к тексту, а также устанавливает её параметры: цвет тени, смещение относительно надписи и радиус размытия.</p> <p>Значения: смещение по горизонтали, смещение по вертикали, радиус размытия, цвет.</p> <p>Например,</p> <pre>h1 { color: red; text-shadow: 2px 2px 3px black; }</pre>
text – align	<p>Определяет горизонтальное выравнивание текста в пределах элемента.</p> <p>Значения:</p> <p>left - выравнивание по левому краю</p> <p>right - выравнивание по правому краю</p> <p>center - выравнивание по центру</p> <p>justify - выравнивание по ширине</p> <p>Например,</p> <pre>h1 { text – align: left; }</pre>
background-color	<p>Определяет цвет фона элемента.</p> <p>Например,</p>

	<pre> p { background-color: red; } </pre>
line-height	<p>Устанавливает интерлиньяж (межстрочный интервал) текста.</p> <p>Например,</p> <pre> h1 { line-height: 1.5; } </pre>
background-image	<p>Устанавливает фоновое изображение для элемента.</p> <p>Например,</p> <pre> h1 { background-image: url(путь к файлу); } </pre>

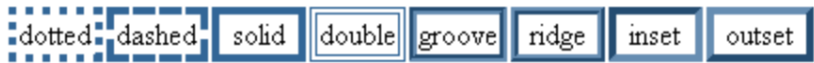
Свойства списков:

List-style-type	<p>Позволяет задать стиль маркера.</p> <p>Значения:</p> <ul style="list-style-type: none"> • disc - закрашенный круг • circle - окружность • square - квадрат • decimal - 1,2,3... • lower-greek - строчные символы греческого алфавита • lower-alpha - a,b,c... • lower-roman - i,ii,iii,iv.... • upper-roman - I, II, III... • upper-latin - A,B,C....
List-style-position	<p>Определяет, как будет размещаться маркер относительно текста.</p> <ul style="list-style-type: none"> • outside - маркер располагается вне блока текста • inside - маркер в одном блоке с текстом

Стили границ

Стили границ можно применять к каждой стороне поочередно:

border-top-style, border-right-style, border-bottom-style, border-left-style

border	<p>Задаёт границы таблицы: толщину, стиль и цвет.</p> <p>Стили рамок:</p>  <p>Пример: border: 2px solid red;</p>
border-collapse	<p>Устанавливает, как отображать границы вокруг ячеек таблицы:</p> <p>collapse - линия между ячейками отображается только одна</p> <p>separate - рамка отображается вокруг каждой ячейки отдельно</p>
border-color	Устанавливает цвет границы на разных сторонах элемента.
border-radius	Радиус скругления углов.

Задание 1. Создайте сайт, по вашему вкусу.

Сайты, которые могут помочь:

<https://colorscheme.ru/html-colors.html> - таблица цветов.

<https://mybrandnewlogo.com/ru/generator-cvetovoi-palitry> - цветовые палитры

<https://www.w3.org/Style/Examples/007/fonts.ru.html> - шрифты