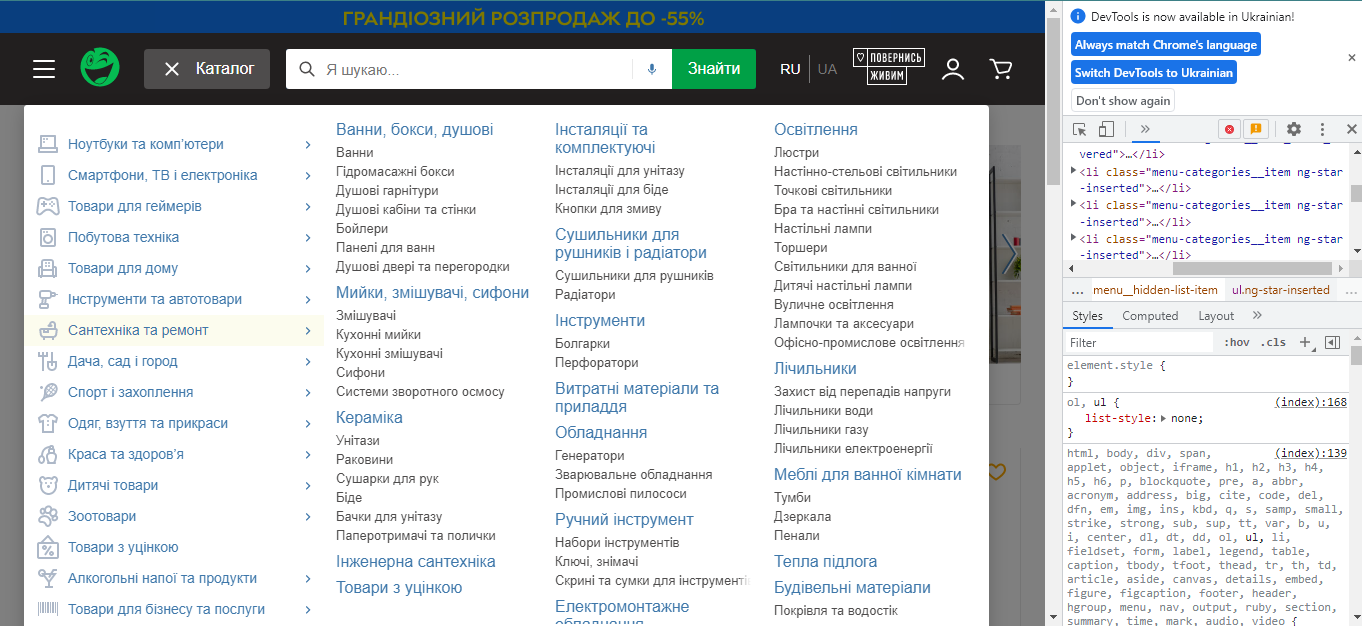
Списки

Что дают списки в HTML и зачем они нужны?

пример интернет магазин ROZETKA  

В HTML для создания списков используются **теги группы lists**.

К ним относятся:

* [<ul>](https://guruweba.com/html/teg-ul-html-markirovannyy-spisok/) - тег маркированного списка;
* [<ol>](https://guruweba.com/html/teg-ol-html-numerovannyy-spisok/) - тег нумерованного списка;
* [<li>](https://guruweba.com/html/teg-li-html-element-spiska/) - тег элемента списка;
* [<dl>](https://guruweba.com/html/teg-dl-html-spisok-opredeleniy/) - тег списка определений;
* [<dt>](https://guruweba.com/html/teg-dt-html-termin-spiska-opredeleniy/) - тег термина в списке определений;
* [<dd>](https://guruweba.com/html/teg-dd-html-opisanie-v-spiske-opredeleniy/) - тег описания термина в списке определений.

**Тег <ul> HTML маркированный список**

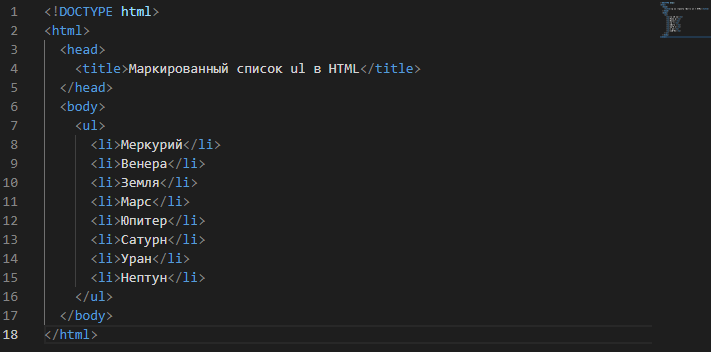
Тег <ul> в HTML определяет маркированный список (неупорядоченный список).

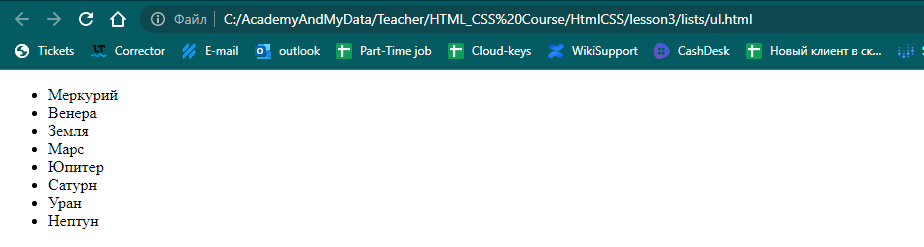
Маркированный список широко используется в коде HTML страниц. Помимо прямого применения тега <ul> для структурирования данных в виде маркированных списков, связку <ul> - [<li>](https://guruweba.com/html/teg-li-html-element-spiska/) часто используют для верстки меню сайта, как одноуровневого, так и многоуровневого. Как

Элемент списка определяется тегом [<li>](https://guruweba.com/html/teg-li-html-element-spiska/) и должен находиться внутри тега <ul>.

## Поддержка браузерами

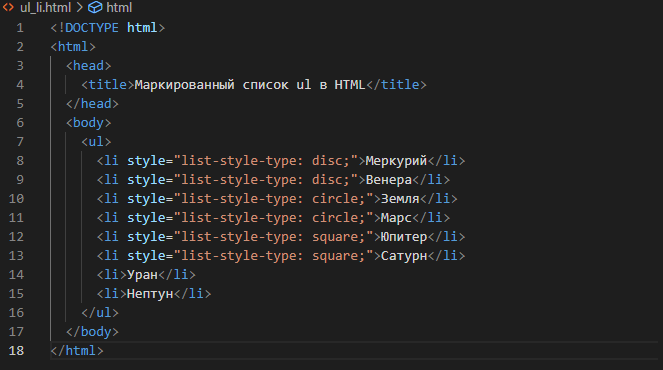






### Типы маркеров

Для маркированного списка доступны 3 типа маркеров по умолчанию: *disc*, *square* и *circle*. Задать тип маркера можно при помощи CSS свойства list-style-type. Тип маркера может быть задан, как для списка в целом (свойство применяется к [<ul>](https://guruweba.com/html/teg-ul-html-markirovannyy-spisok/)), так и для конкретного элемента (свойство применяется к [<li>](https://guruweba.com/html/teg-li-html-element-spiska/))

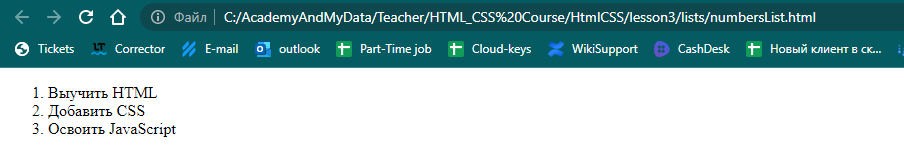




## Нумерованный список

Чтобы создать нумерованный список используйте теги [<ol>](https://guruweba.com/html/teg-ol-html-numerovannyy-spisok/) и [<li>](https://guruweba.com/html/teg-li-html-element-spiska/). Тег ol - это контейнер нумерованного списка. Тег li - элемент списка.





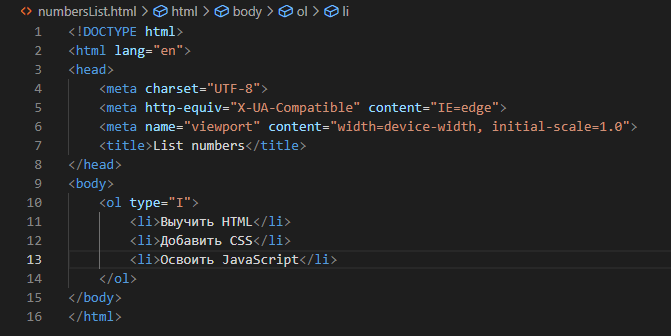
### Типы нумерации в HTML списке

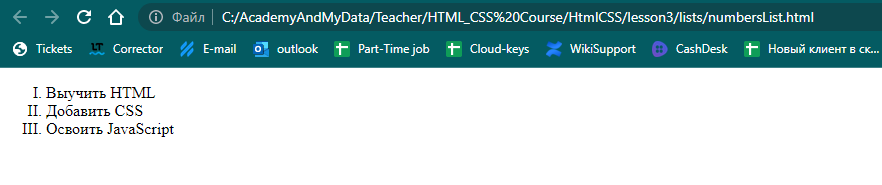
В HTML доступны 5 типов нумерации в списках. Чтобы изменить тип нумерации используйте атрибут type. Тип нумерации можно задавать как для списка в целом, так и для конкретных его элементов.

Разные типы нумерации в списках:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| type="I"   1. Первый 2. Второй 3. Третий | type="i"   1. Первый 2. Второй 3. Третий | type="1"   1. Первый 2. Второй 3. Третий | type="A"   1. Первый 2. Второй 3. Третий | type="a"   1. Первый 2. Второй 3. Третий |

Другие варианты нумерации, включая греческую, еврейскую, японскую, с ведущими нулями и т.д. можно получить используя CSS свойство list-style-type



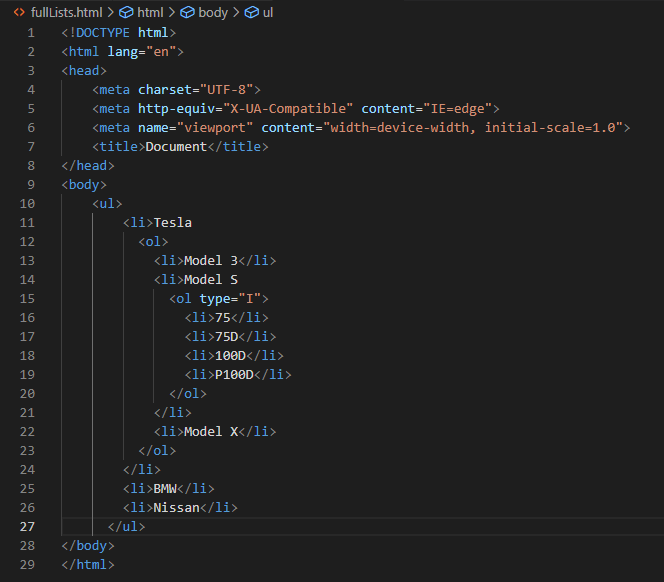


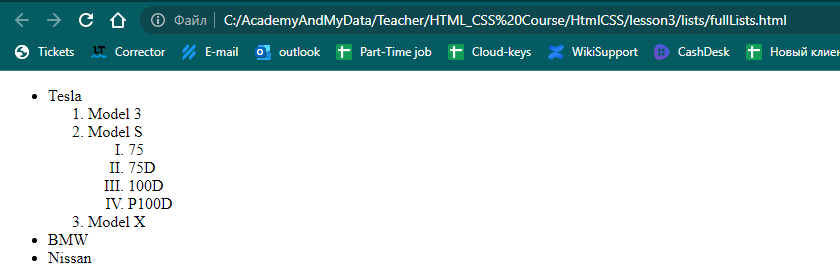
## Многоуровневый список HTML (список в списке)

HTML не имеет отдельного элемента для создания многоуровневых списков. Вместо этого в HTML используются вложенные списки - список в списке.

Чтобы создать новый уровень списка, нужно поместить внутрь элемента списка [<li>](https://guruweba.com/html/teg-li-html-element-spiska/) вложенный список. Вложенными могут быть маркированные, нумерованные, списки определений или другие многоуровневые списки.

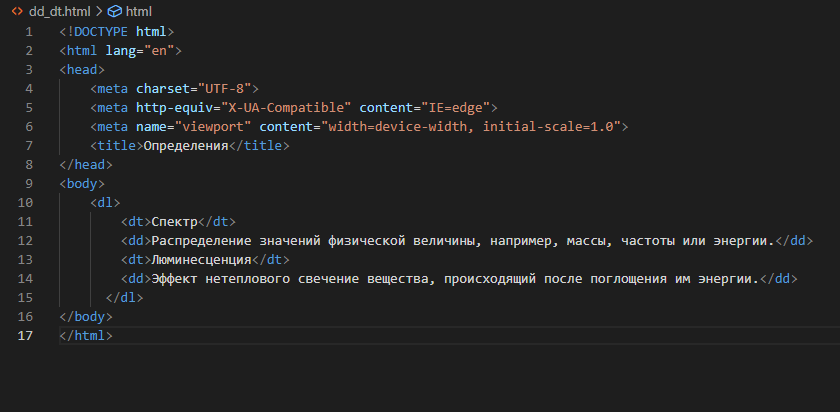
Ограничений по количеству уровней вложенности списков нет.

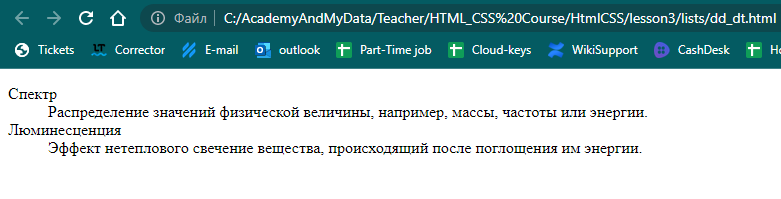




## Список определений

Для создания списка определений используются теги [<dl>](https://guruweba.com/html/teg-dl-html-spisok-opredeleniy/), [<dt>](https://guruweba.com/html/teg-dt-html-termin-spiska-opredeleniy/) и [<dd>](https://guruweba.com/html/teg-dd-html-opisanie-v-spiske-opredeleniy/). Тег dl - это контейнер списка определений, dt - термин, dd - описание термина.





Отступы

Внутренние и внешние отступы.

Свойство padding определяет расстояние от содержимого (контента) блочного элемента до его границы, а margin — расстояние между границами элементов, расположенных рядом. Внешние отступы являются прозрачными, а внутренние можно залить фоновым цветом.



Отступы можно назначать с 4-х сторон элемента:

верхней (англ. название — top),

правой (англ. название — right),

нижней (англ. название — bottom)

левой (англ. название — left):

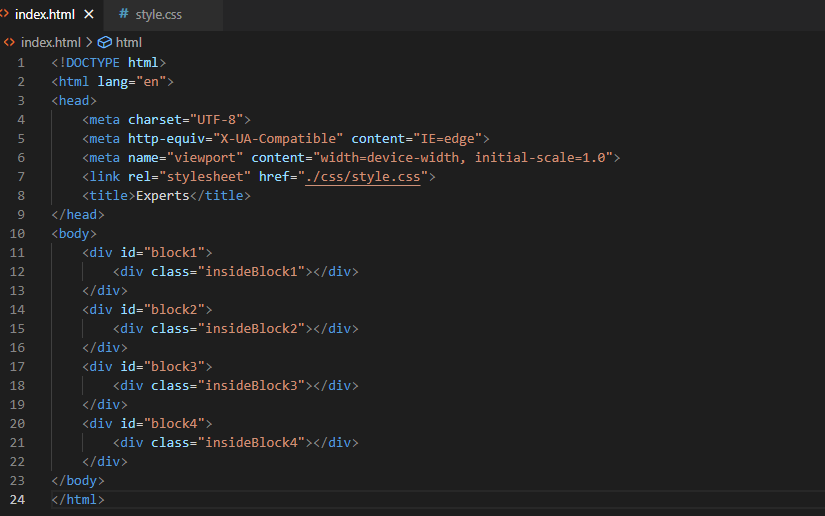
■ padding-top/margin-top — верхний отступ (внутренний/внешний);

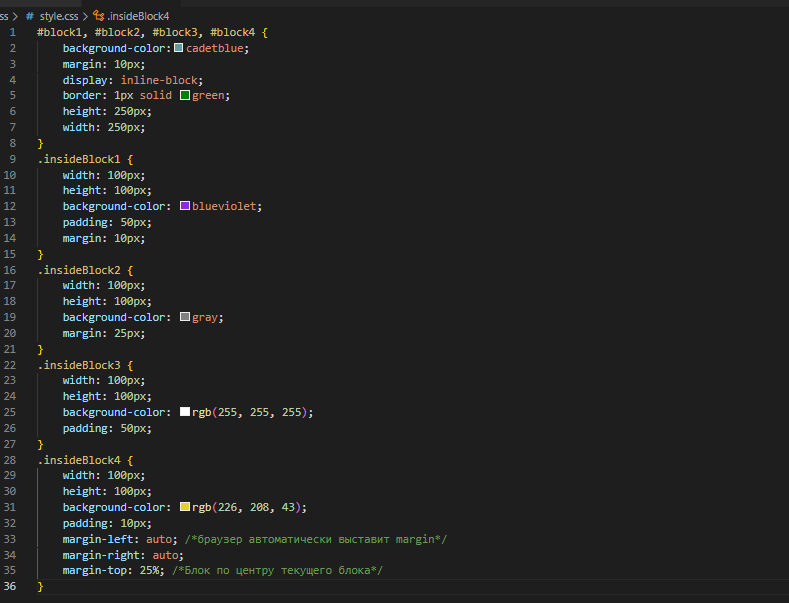
■ padding-right /margin-right — правый отступ (внутренний/внешний);

■ padding-bottom /margin-bottom — нижний отступ (внутренний/внешний);

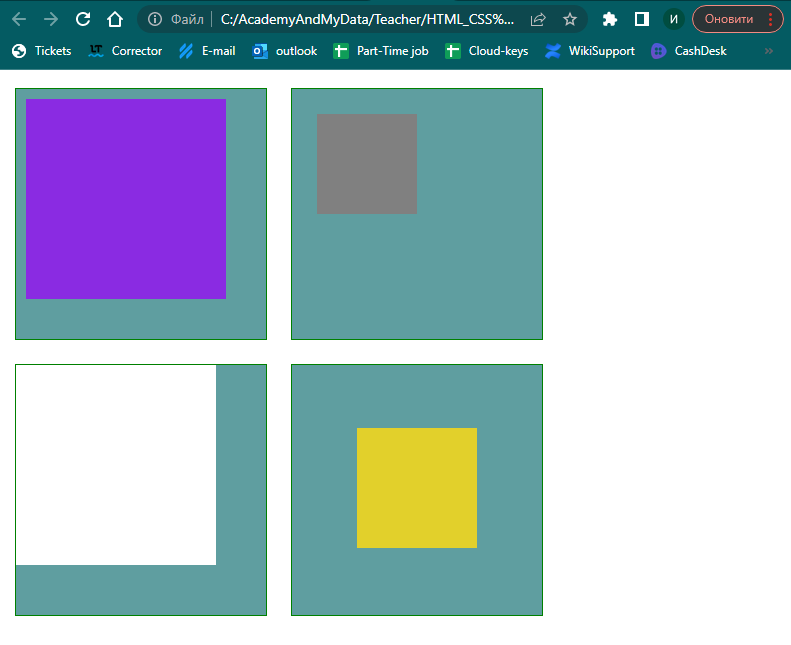
■ padding-left /margin-left — левый отступ (внутренний/ внешний);

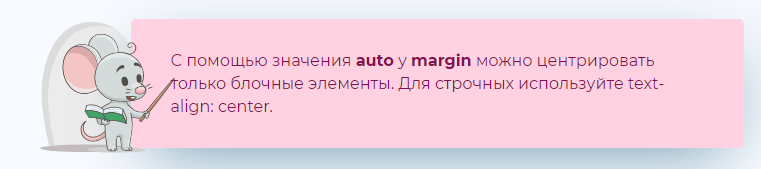
■ padding/margin — отступ (внутренний/внешний) (сокращённая форма).

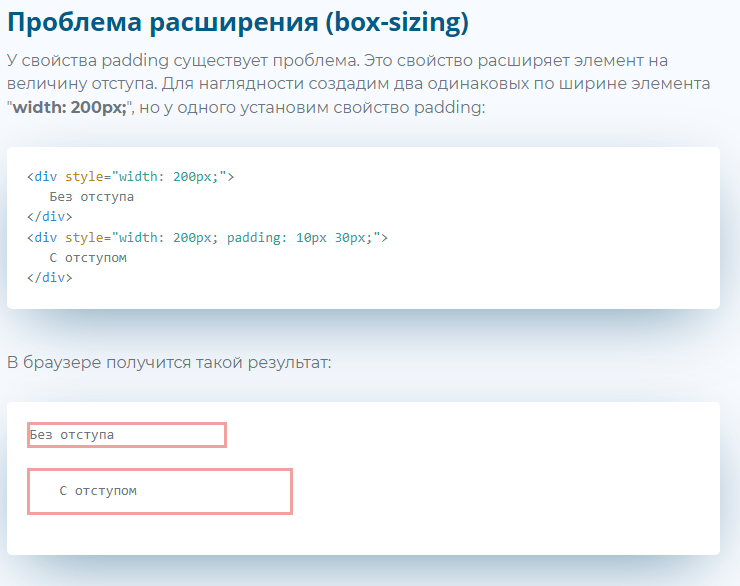




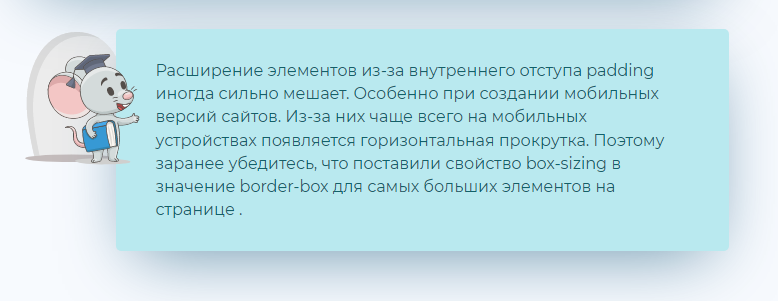
Можно убрать все стандартные отступы с помощью:  
\* { margin: 0; padding: 0; }  
и заменить нужно display: block-inline на float











Псевдоэлементы

**Псевдоэлемент** в CSS — это ключевое слово, добавляемое к селектору, которое позволяет стилизовать определённую часть выбранного элемента. Например, псевдоэлемент [::first-line](https://developer.mozilla.org/ru/docs/Web/CSS/::first-line) может быть использован для изменения шрифта первой строки абзаца.

## [Список стандартных псевдоэлементов](https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-elements#%D1%81%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D1%8B%D1%85_%D0%BF%D1%81%D0%B5%D0%B2%D0%B4%D0%BE%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82%D0%BE%D0%B2)

* [::after](https://developer.mozilla.org/ru/docs/Web/CSS/::after)
* [::before](https://developer.mozilla.org/ru/docs/Web/CSS/::before)
* [::cue](https://developer.mozilla.org/ru/docs/Web/CSS/::cue)
* [::first-letter](https://developer.mozilla.org/ru/docs/Web/CSS/::first-letter)
* [::first-line](https://developer.mozilla.org/ru/docs/Web/CSS/::first-line)
* [::selection](https://developer.mozilla.org/ru/docs/Web/CSS/::selection)
* [::slotted](https://developer.mozilla.org/ru/docs/Web/CSS/::slotted)
* [::backdrop](https://developer.mozilla.org/ru/docs/Web/CSS/::backdrop) Experimental
* [::placeholder](https://developer.mozilla.org/ru/docs/Web/CSS/::placeholder) Experimental
* [::marker](https://developer.mozilla.org/ru/docs/Web/CSS/::marker) Experimental
* [::spelling-error (en-US)](https://developer.mozilla.org/en-US/docs/Web/CSS/::spelling-error) Experimental
* [::grammar-error](https://developer.mozilla.org/ru/docs/Web/CSS/::grammar-error) Experimental

<https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-elements>

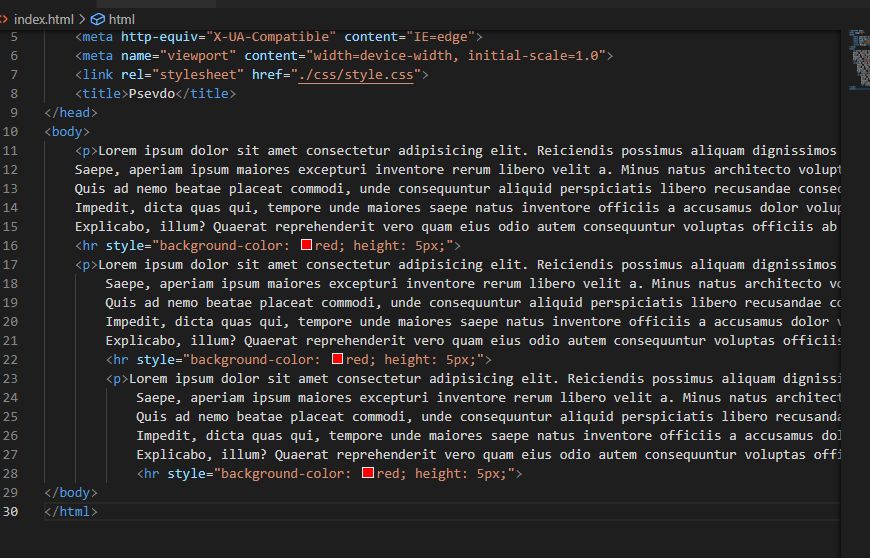
Псевдоклассы

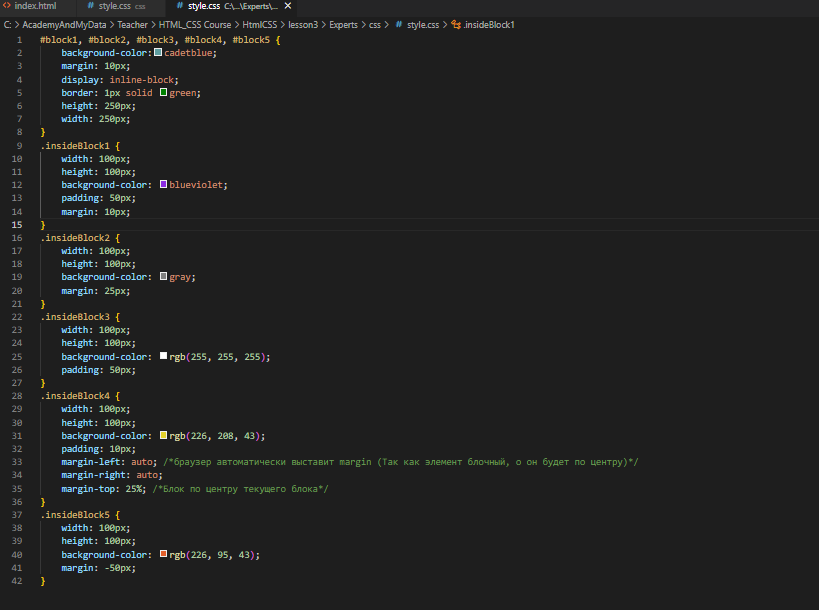
**Псевдокласс** в CSS — это ключевое слово, добавленное к селектору, которое определяет его особое состояние. Например, [:hover](https://developer.mozilla.org/ru/docs/Web/CSS/:hover) может быть использован для изменения цвета кнопки при наведении курсора на неё.

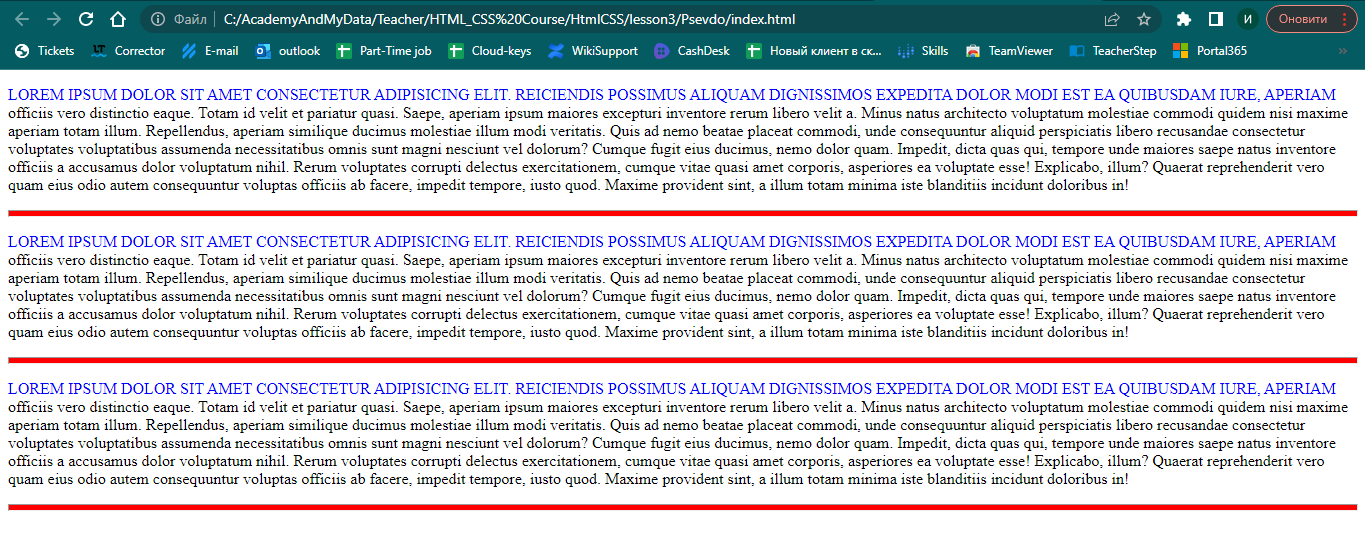
## [Список стандартных псевдоклассов](https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-classes#%D1%81%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%BD%D1%8B%D1%85_%D0%BF%D1%81%D0%B5%D0%B2%D0%B4%D0%BE%D0%BA%D0%BB%D0%B0%D1%81%D1%81%D0%BE%D0%B2)

* [:active](https://developer.mozilla.org/ru/docs/Web/CSS/:active)
* [:any (en-US)](https://developer.mozilla.org/en-US/docs/Web/CSS/:is)
* [:any-link](https://developer.mozilla.org/ru/docs/Web/CSS/:any-link)
* [:checked](https://developer.mozilla.org/ru/docs/Web/CSS/:checked)
* [:default](https://developer.mozilla.org/ru/docs/Web/CSS/:default)
* [:defined](https://developer.mozilla.org/ru/docs/Web/CSS/:defined)
* [:dir()](https://developer.mozilla.org/ru/docs/Web/CSS/:dir)
* [:disabled](https://developer.mozilla.org/ru/docs/Web/CSS/:disabled)
* [:empty](https://developer.mozilla.org/ru/docs/Web/CSS/:empty)
* [:enabled](https://developer.mozilla.org/ru/docs/Web/CSS/:enabled)
* [:first](https://developer.mozilla.org/ru/docs/Web/CSS/:first)
* [:first-child](https://developer.mozilla.org/ru/docs/Web/CSS/:first-child)
* [:first-of-type](https://developer.mozilla.org/ru/docs/Web/CSS/:first-of-type)
* [:fullscreen](https://developer.mozilla.org/ru/docs/Web/CSS/:fullscreen)
* [:focus](https://developer.mozilla.org/ru/docs/Web/CSS/:focus)
* [:hover](https://developer.mozilla.org/ru/docs/Web/CSS/:hover)
* [:indeterminate](https://developer.mozilla.org/ru/docs/Web/CSS/:indeterminate)
* [:in-range](https://developer.mozilla.org/ru/docs/Web/CSS/:in-range)
* [:invalid](https://developer.mozilla.org/ru/docs/Web/CSS/:invalid)
* [:lang()](https://developer.mozilla.org/ru/docs/Web/CSS/:lang)
* [:last-child](https://developer.mozilla.org/ru/docs/Web/CSS/:last-child)
* [:last-of-type](https://developer.mozilla.org/ru/docs/Web/CSS/:last-of-type)
* [:left](https://developer.mozilla.org/ru/docs/Web/CSS/:left)
* [:link](https://developer.mozilla.org/ru/docs/Web/CSS/:link)
* [:not()](https://developer.mozilla.org/ru/docs/Web/CSS/:not)
* [:nth-child()](https://developer.mozilla.org/ru/docs/Web/CSS/:nth-child)
* [:nth-last-child()](https://developer.mozilla.org/ru/docs/Web/CSS/:nth-last-child)
* [:nth-last-of-type()](https://developer.mozilla.org/ru/docs/Web/CSS/:nth-last-of-type)
* [:nth-of-type()](https://developer.mozilla.org/ru/docs/Web/CSS/:nth-of-type)
* [:only-child](https://developer.mozilla.org/ru/docs/Web/CSS/:only-child)
* [:only-of-type](https://developer.mozilla.org/ru/docs/Web/CSS/:only-of-type)
* [:optional](https://developer.mozilla.org/ru/docs/Web/CSS/:optional)
* [:out-of-range](https://developer.mozilla.org/ru/docs/Web/CSS/:out-of-range)
* [:read-only](https://developer.mozilla.org/ru/docs/Web/CSS/:read-only)
* [:read-write](https://developer.mozilla.org/ru/docs/Web/CSS/:read-write)
* [:required](https://developer.mozilla.org/ru/docs/Web/CSS/:required)
* [:right](https://developer.mozilla.org/ru/docs/Web/CSS/:right)
* [:root](https://developer.mozilla.org/ru/docs/Web/CSS/:root)
* [:scope (en-US)](https://developer.mozilla.org/en-US/docs/Web/CSS/:scope)
* [:target](https://developer.mozilla.org/ru/docs/Web/CSS/:target)
* [:valid](https://developer.mozilla.org/ru/docs/Web/CSS/:valid)
* [:visited](https://developer.mozilla.org/ru/docs/Web/CSS/:visited)

<https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-classes>







CSS Стили для списков

<https://html5book.ru/css-spiski/>

С помощью стандартных CSS-свойств можно **изменить внешний вид маркера списка**, **добавить изображение для маркера**, а также **изменить местоположение маркера**. Высоту блока маркера можно задать свойством line-height

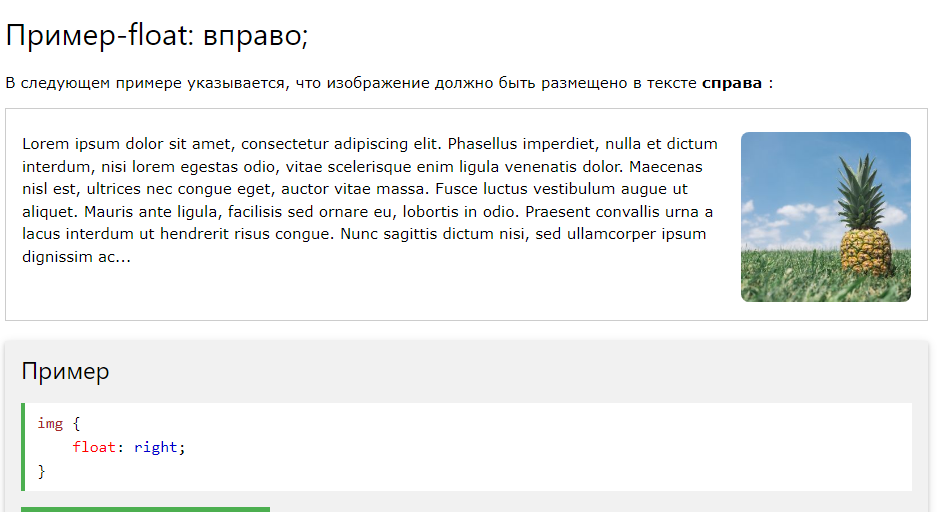
* 1. <https://itchief.ru/html-and-css/pseudo-elements>
  2. <https://developer.mozilla.org/ru/docs/Web/CSS/::before>
  3. <https://html5book.ru/examples/demo-lists.html>

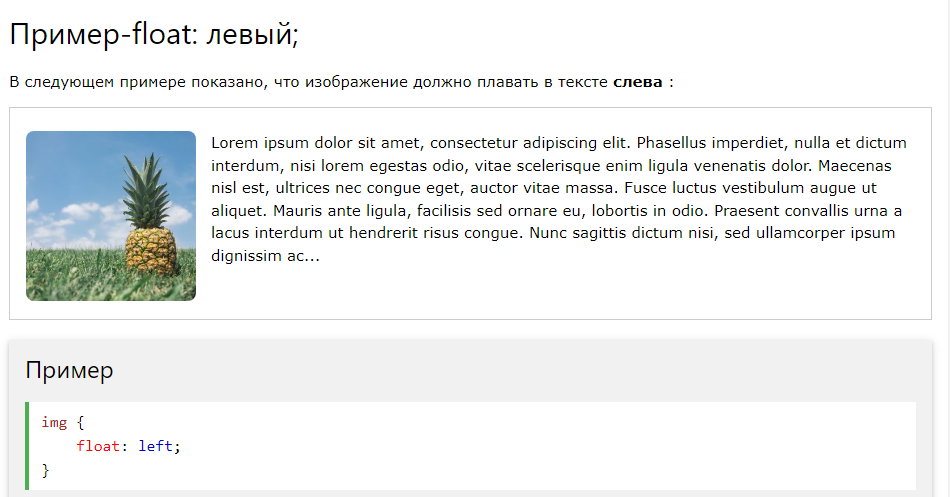
# Перед примером css style for lists пробежимся по некоторым часто используемым свойствам float

Свойство float CSS указывает, что элемент должен быть взят из нормального потока и помещён вдоль левой или правой стороны его контейнера, где текст и встроенные элементы будут обтекать его.

CSS-свойство float указывает, что текущий элемент должен быть изъят из обычного flow (потока) и прижат к левой или правой стороне

родительского элемента. Текст и inline элементы будут обтекать такой элемент.





<https://developer.mozilla.org/ru/docs/Web/CSS/float>

list-style

Универсальное свойство, позволяющее одновременно задать стиль маркера, его положение, а также изображение, которое будет использоваться в качестве маркера. Для подробного ознакомления смотрите информацию о каждом свойстве [list-style-type](http://htmlbook.ru/css/list-style-type), [list-style-position](http://htmlbook.ru/css/list-style-position) и [list-style-image](http://htmlbook.ru/css/list-style-image) отдельно.

<https://developer.mozilla.org/en-US/docs/Web/CSS/list-style>

text-shadow

CSS-свойство **text-shadow** добавляет тени к тексту. Свойство задаётся разделённым запятыми списком теней, которые будут применены к тексту и к любым его свойствам [decorations](https://developer.mozilla.org/en-US/docs/Web/CSS/text-decoration). Любая тень описывается комбинацией смещений по осям X и Y относительно элемента, радиусом размытия и цветом.

<https://developer.mozilla.org/ru/docs/Web/CSS/text-shadow>

position

Позиционирование позволяет вам изымать элементы из нормального потока макета документа и заставить их вести себя по-другому; например, располагаться друг на друге или всегда оставаться на одном и том же месте внутри окна просмотра браузера.

<https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning>

# Позиционирование

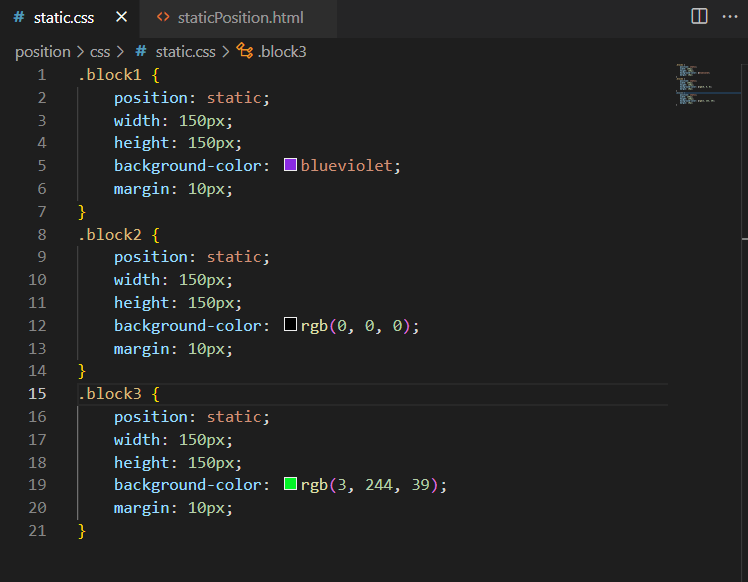
Позиционирование позволяет вам изымать элементы из нормального потока макета документа и заставить их вести себя по-другому; например, располагаться друг на друге или всегда оставаться на одном и том же месте внутри окна просмотра браузера.

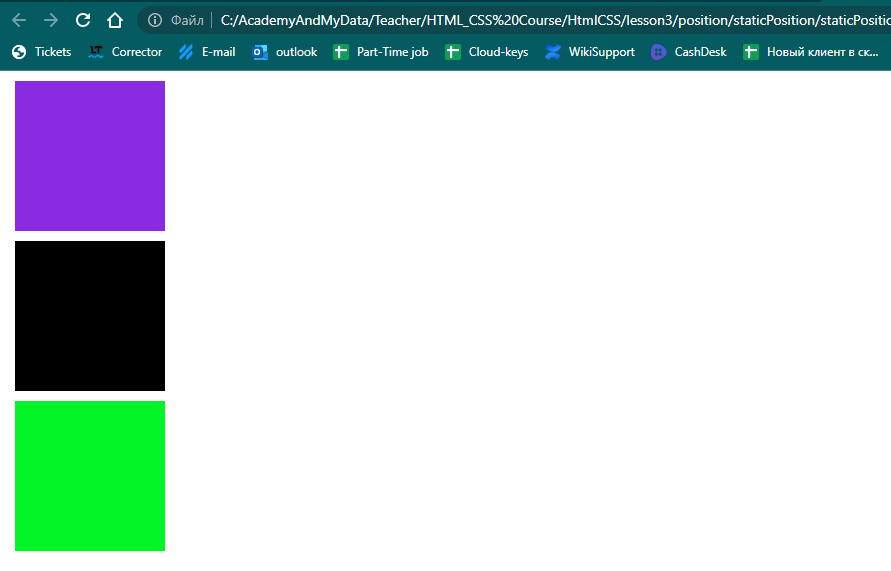
Существует несколько разных типов позиционирования, которые вы можете применить к элементам HTML. Для активации специфического типа позиционирования у элемента, мы используем свойство [position](https://developer.mozilla.org/ru/docs/Web/CSS/position).

### [Статическое позиционирование](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D1%81%D1%82%D0%B0%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%BE%D0%B5_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)

Статическое позиционирование — это умолчание, которое получает каждый элемент, что всего лишь значит "поставить элемент в его нормальное положение в потоке макета документа — ничего особенного для рассмотрения”.



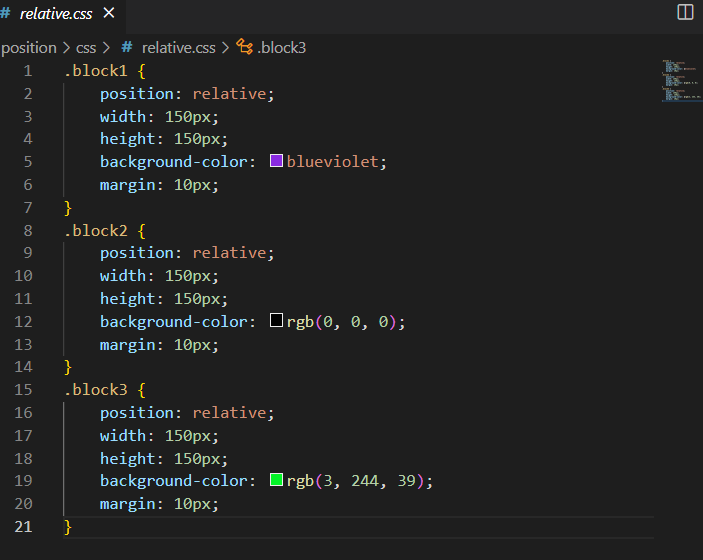


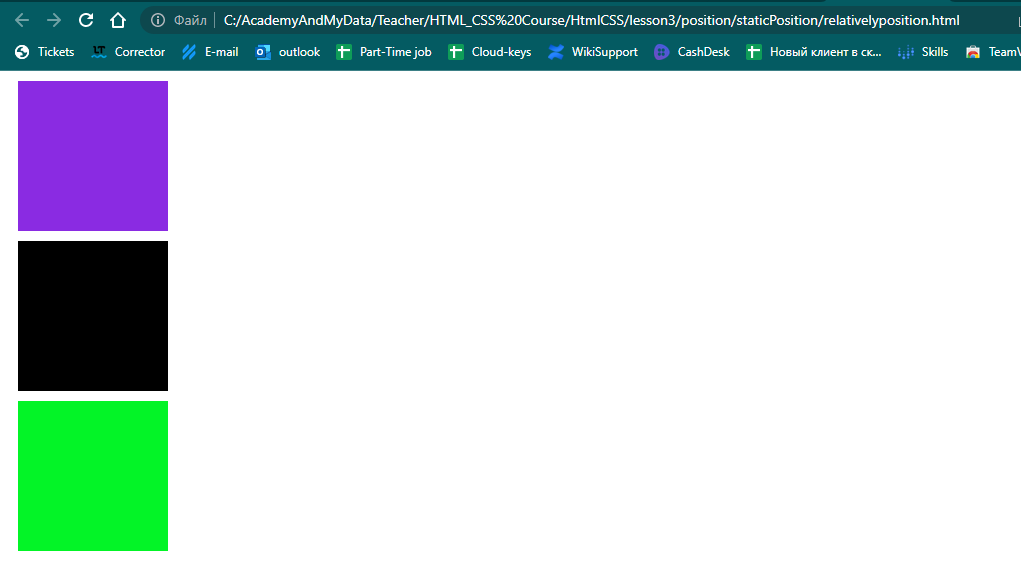


### [Относительное позиционирование](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D0%BE%D1%82%D0%BD%D0%BE%D1%81%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)

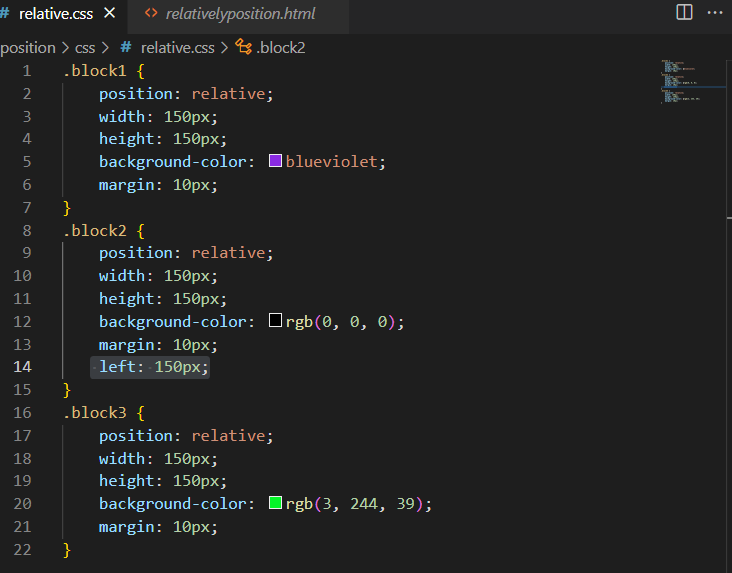
Относительное позиционирование первый тип позиции, который мы рассмотрим. Оно очень похоже на статическое позиционирование, за исключением того что вы можете модифицировать окончательное положение позиционируемого объекта занявшего своё место в макете нормального потока, в том числе заставлять его перекрывать другие элементы на странице.

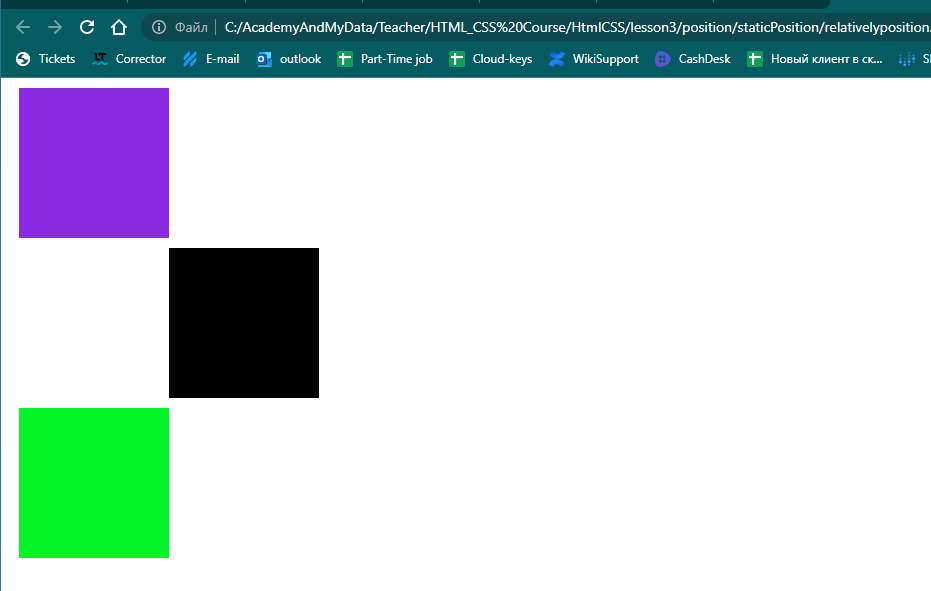






В результате вы совсем не увидите изменений. Так как же вам модифицировать положение? Вам необходимо использовать свойства [top](https://developer.mozilla.org/ru/docs/Web/CSS/top), [bottom](https://developer.mozilla.org/ru/docs/Web/CSS/bottom), [left (en-US)](https://developer.mozilla.org/en-US/docs/Web/CSS/left), и [right](https://developer.mozilla.org/ru/docs/Web/CSS/right) которые мы объясним в следующем разделе.

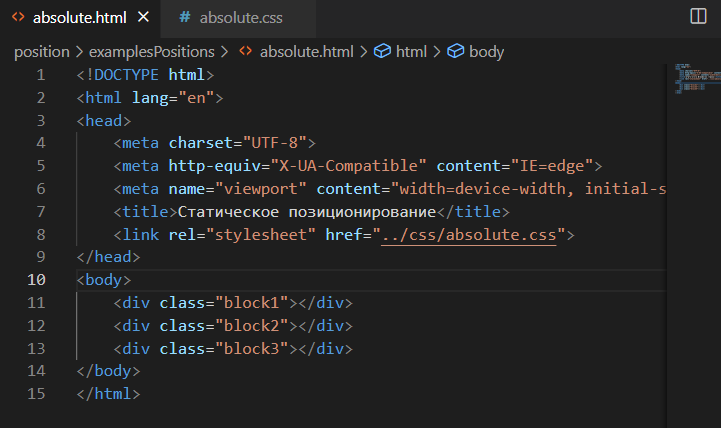


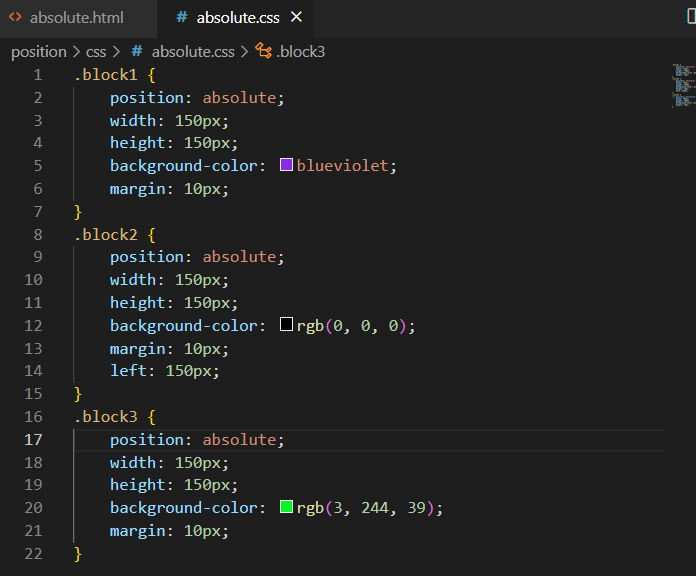


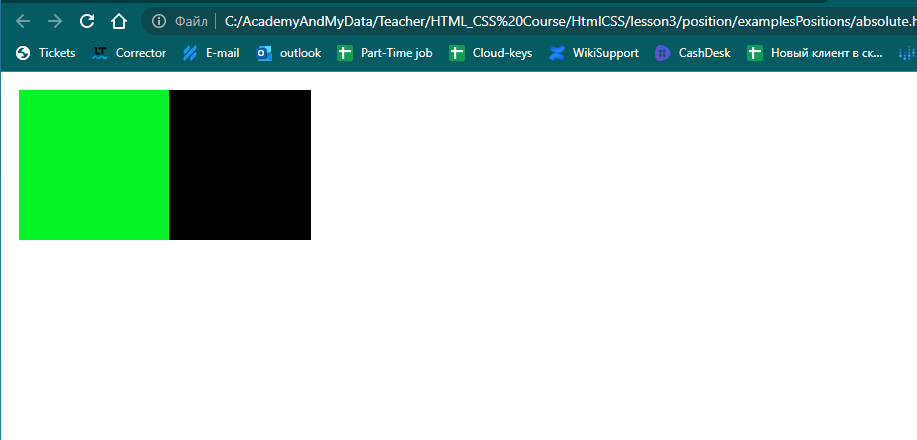
Вероятно это не то, чего вы ожидали — почему он переместился вниз и вправо, когда мы указали вверх и влево? Как бы нелогично это ни звучало это всего лишь способ того как работает позиционирование — вам надо думать о невидимой силе толкающей указанную сторону позиционируемого блока, перемещая его в противоположную сторону. Так, например, если вы указали top: 30px;, сила толкает блок, заставляя его перемещаться вниз на 30px.

### [Абсолютное позиционирование](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D0%B0%D0%B1%D1%81%D0%BE%D0%BB%D1%8E%D1%82%D0%BD%D0%BE%D0%B5_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)

Абсолютное позиционирование даёт совершенно другие результаты.





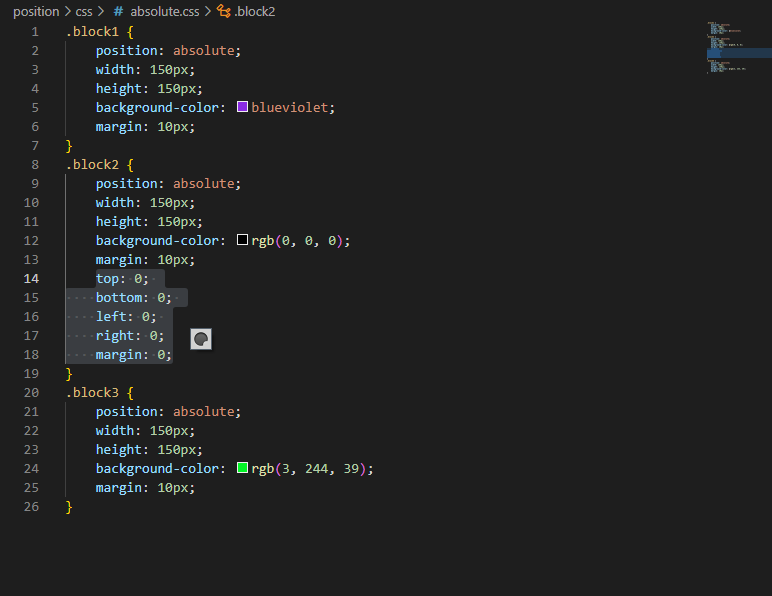


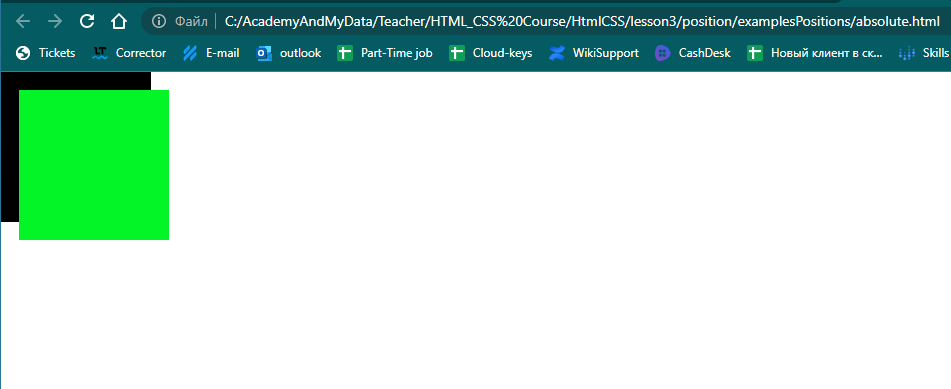
В первую очередь обратите внимание на то, что интервал там, где должен быть позиционируемый элемент в потоке документа теперь отсутствует — первый и третий элементы сблизились так будто, он больше не существует! Ну, в каком-то смысле это правда. Абсолютно позиционированный элемент больше не существует в нормальном потоке макета документа. Вместо этого он располагается на своём собственном слое отдельно от всего остального.

Это очень полезно: это значит, что мы можем создавать изолированные функции пользовательского интерфейса, которые не влияют на макет других элементов страницы. Например, всплывающие информационные блоки и меню управления; опрокидывающиеся панели; функции пользовательского интерфейса, которые можно перетаскивать в любом месте страницы; и так далее...

Во-вторых, обратите внимание, что позиция элемента изменилась — это потому, что [top](https://developer.mozilla.org/ru/docs/Web/CSS/top), [bottom](https://developer.mozilla.org/ru/docs/Web/CSS/bottom), left, и [right](https://developer.mozilla.org/ru/docs/Web/CSS/right) ведут себя по-другому с абсолютным позиционированием. Вместо того, чтобы позиционировать элемент на основе его относительного положения в обычном потоке макета документа, они определяют расстояние, на котором элемент должен находиться от каждой из сторон содержащего элемента.

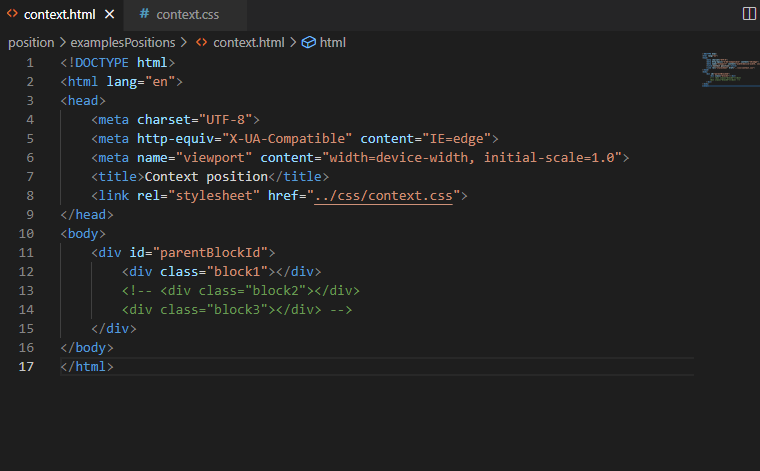
**Примечание**: вы можете использовать [top](https://developer.mozilla.org/ru/docs/Web/CSS/top), [bottom](https://developer.mozilla.org/ru/docs/Web/CSS/bottom), left, и [right](https://developer.mozilla.org/ru/docs/Web/CSS/right) для изменения размера элемента если вам надо. Попробуйте установить top: 0; bottom: 0; left: 0; right: 0; и margin: 0;

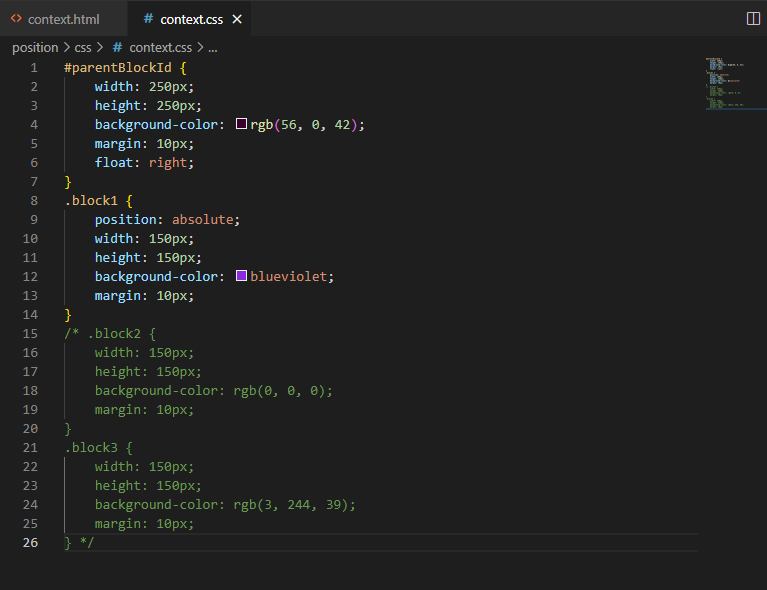


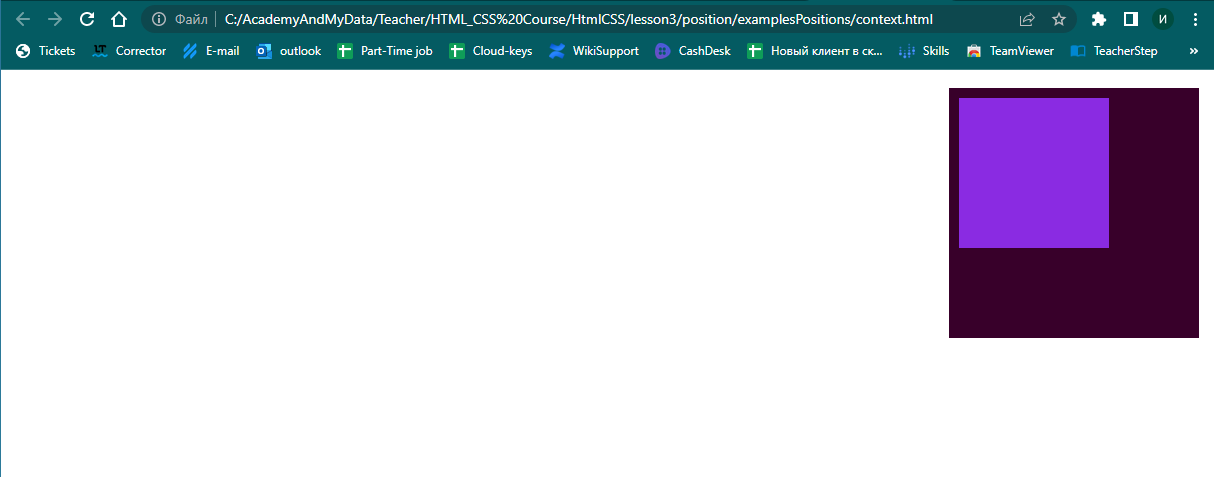


### [Контекст позиционирования](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D0%BA%D0%BE%D0%BD%D1%82%D0%B5%D0%BA%D1%81%D1%82_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)

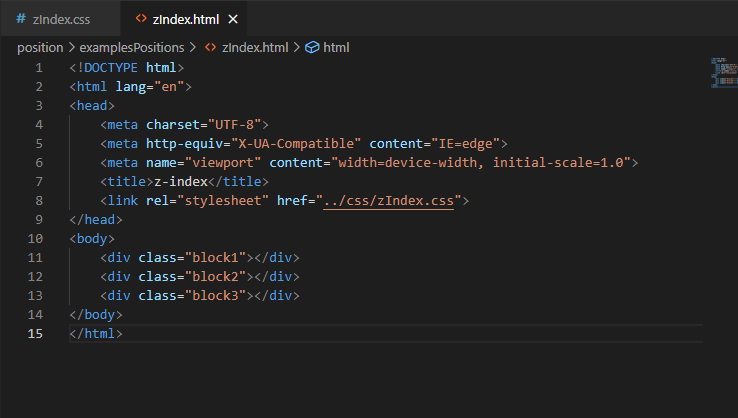
Если никакие из элементов предков не имеют конкретно заданного свойства позиции, то по умолчанию все элементы предков будут иметь статическую позицию. В результате этого абсолютно позиционируемый элемент будет содержаться в **исходным содержащем блоке**. Исходный содержащий блок имеет размеры области просмотра, а также является блоком, содержащим элемент [<html>](https://developer.mozilla.org/ru/docs/Web/HTML/Element/html). Проще говоря, абсолютно позиционируемый элемент будет отображаться за пределами элемента [<html>](https://developer.mozilla.org/ru/docs/Web/HTML/Element/html) и будет расположен относительно исходного окна просмотра.

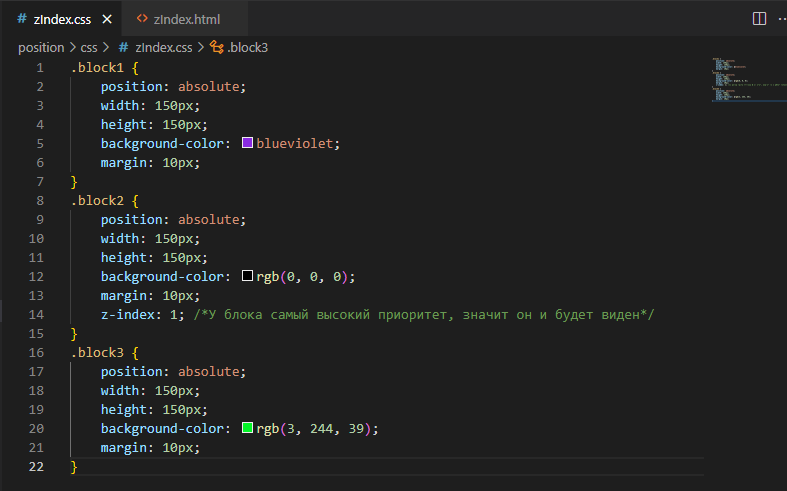


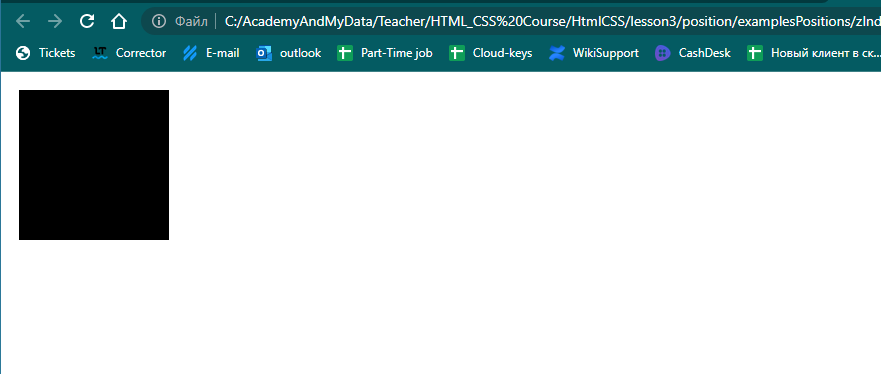




### [z-index](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D0%B2%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5_%D0%B2_z-index)







### [Фиксированное позиционированиее](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#%D1%84%D0%B8%D0%BA%D1%81%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D0%BE%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%D0%B5)

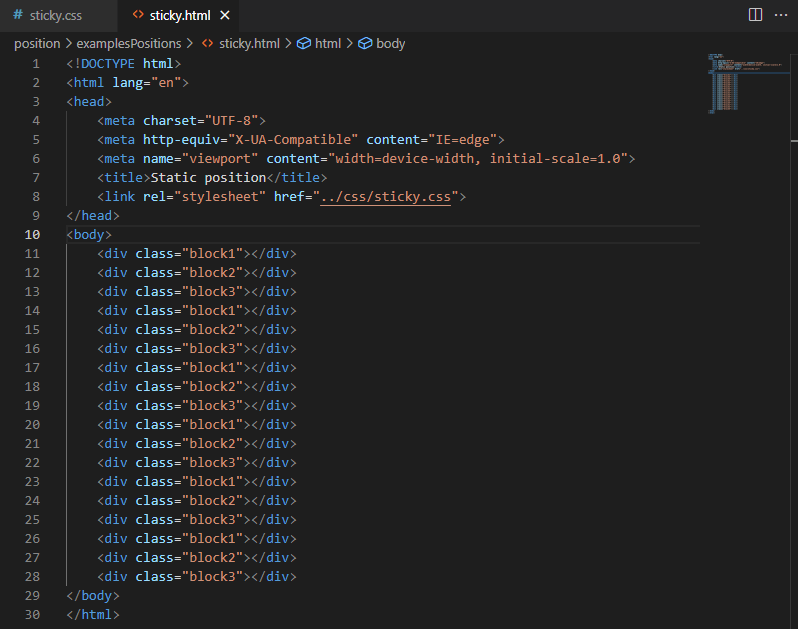
position: fixed;

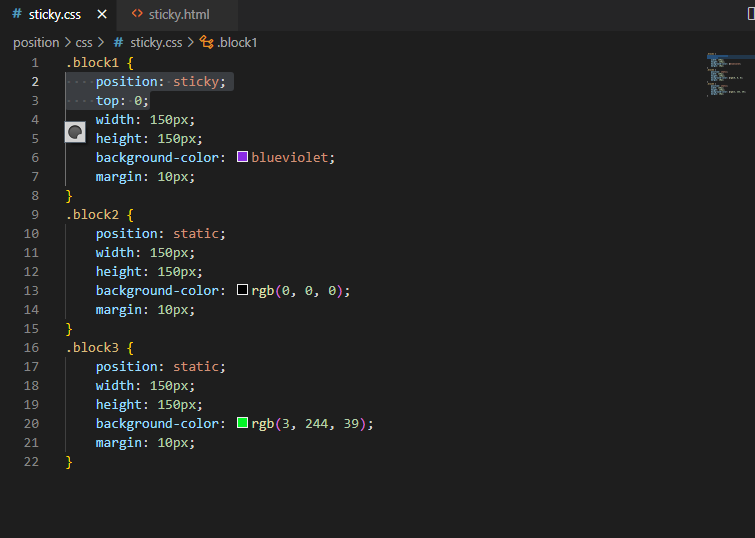
Это значит, что вы можете создать элементы пользовательского интерфейса, которые зафиксированы на месте, как постоянные меню навигации, которые всегда видимы вне зависимости от того сколько прокручивается страница.

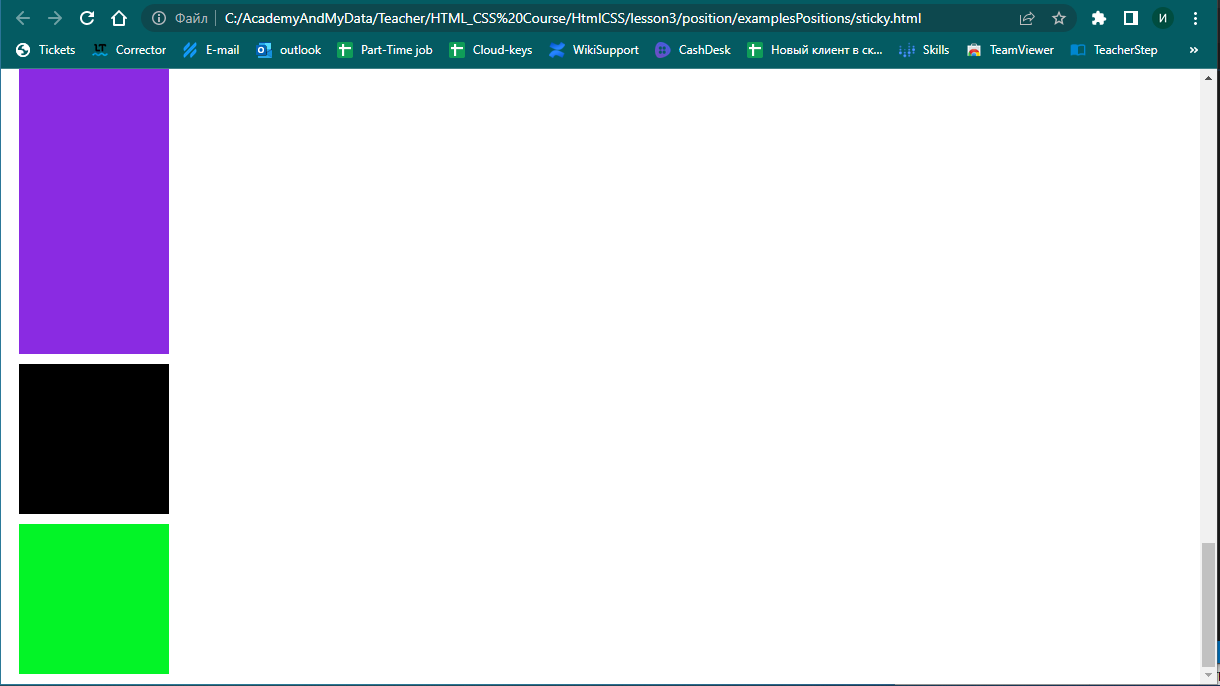
### [position: sticky](https://developer.mozilla.org/ru/docs/Learn/CSS/CSS_layout/Positioning#position_sticky);

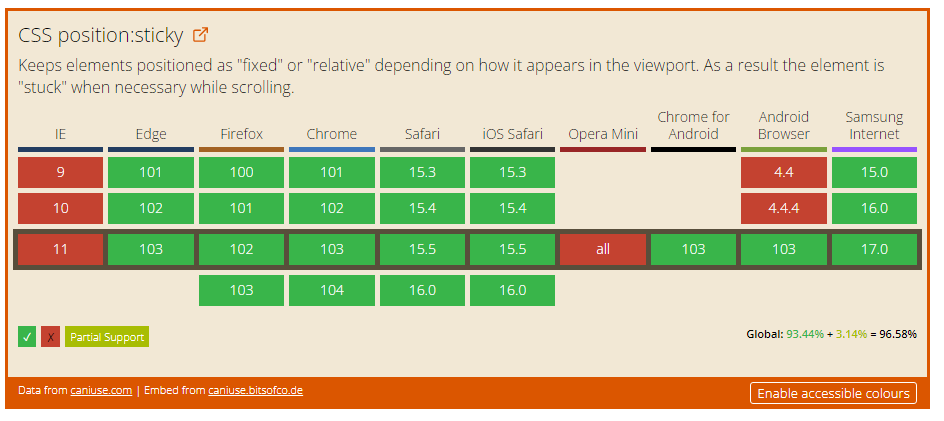
Доступно другое значение позиции называемое position: sticky, которое несколько новее чем другие. По сути, это гибрид относительной и фиксированной позиции, который позволяет позиционируемому элементу вести себя как будто он относительно

позиционирован, до тех пор пока он не будет прокручен до определённой пороговой точки



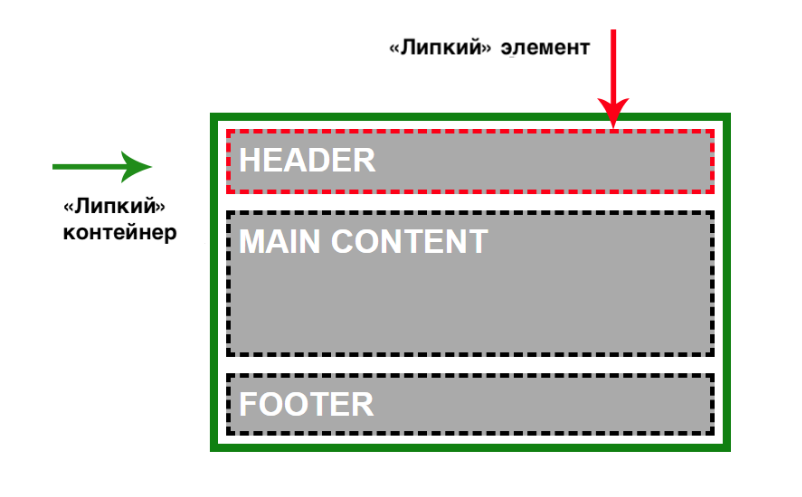






<https://medium.com/web-standards/sticky-bc7ff7088693>

sticky (ENG) - липкий (РУС)



Шрифты

### font-family

Семейство шрифта

В этом свойстве через запятую перечисляются шрифты в порядке приоритета. Браузер будет использовать первый из списка. Если первый не будет найден, будет использоваться следующий и так далее. Если ни один не будет найден, тогда браузер применит шрифт по умолчанию. Если в имени шрифта есть пробелы, его необходимо заключать в одинарные или двойные кавычки.

После всех шрифтов необходимо добавить семейство шрифтов, из которого будет выбран подходящий шрифт, если требуемых шрифтов не окажется на компьютере.

Пример:

**p** {

**font-family**: Verdana, Helvetica, Arial, **sans-serif**;

}

Существуют 5 семейств шрифтов:

1. serif — шрифты с засечками
2. sans-serif — шрифты без засечек
3. monospace — моноширинные
4. cursive — курсивные
5. fantasy — декоративные

Пример, как выглядят такие шрифты:

serif sans-serif monospace cursive fantasy

### font-style

Стиль шрифта

Существуют три значения этого свойства:

1. normal — обычный (значение по умолчанию)
2. italic — курсив
3. oblique — наклонный

Браузеры последний стиль обычно отображают как курсивный

### font-variant

Вариант шрифта

1. normal — обычный (значение по умолчанию)
2. small-caps — капитель, то есть все строчные символы становятся уменьшенными заглавными

### font-weight

Вес шрифта

Это свойство устанавливает вес (насыщенность) шрифта.

Возможные значения:

1. 100
2. 200
3. 300
4. 400 или normal

(значение по умолчанию)

1. 500
2. **600**
3. **700 или bold**
4. **800**
5. **900**

### font-size

Размер шрифта

Возможные значения этого свойства:

1. абсолютные константы: xx-small, x-small, small, medium (значение по умолчанию), large, x-large, xx-large
2. относительные константы: smaller, larger
3. абсолютные и относительные единицы измерения CSS

### line-height

Межстрочный интервал, часто применяется вместе с font-size

Возможные значения:

1. normal — нормальное значение (по умолчанию)
2. number — число (больше либо равно 0), на которое умножается текущий размер шрифта
3. length — фиксированное значение в единицах измерения CSS
4. % — проценты от текущего размера шрифта

### font

Это сокращённая форма записи свойств шрифта. Необходимо лишь указать font-size и font-family. Остальные свойства шрифта указываются при желании

Пример:

**p** {

**font**: **bold** **italic** **small-caps** 150%/1.4 Arial, Helvetica, **sans-serif**;

}

Также можно указывать ключевые слова: caption, icon, menu, message-box, small-caption, status-bar

### web-шрифты

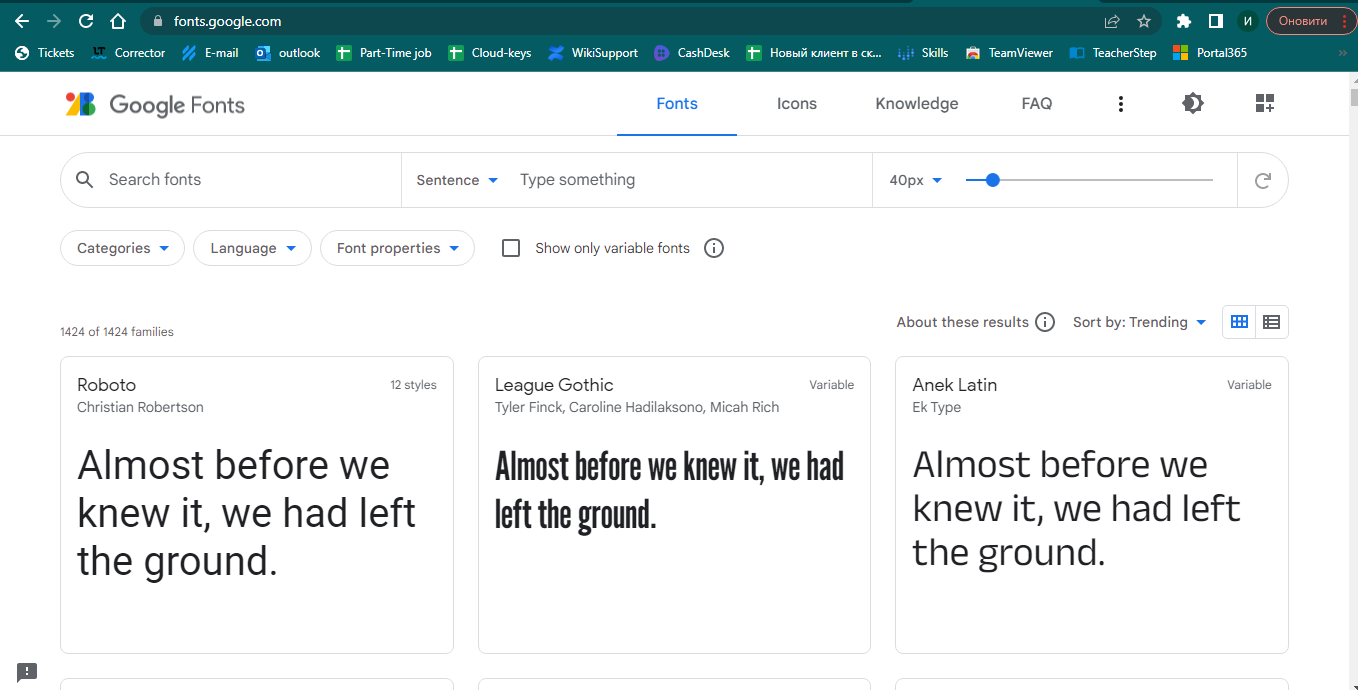
Свойство, которое позволяет отображать на экране компьютера любой шрифт. Впервые появилось в CSS2. Используется правило @font-face

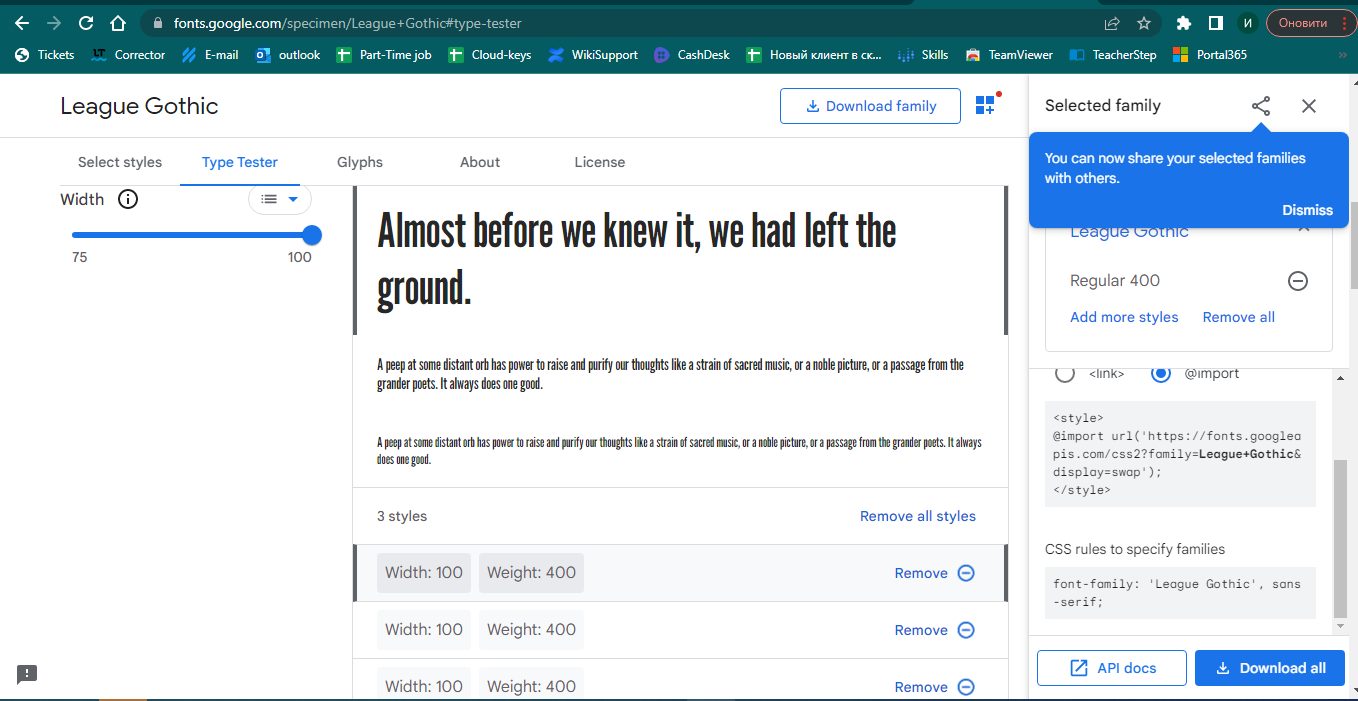
Шрифты бывают следующих типов:

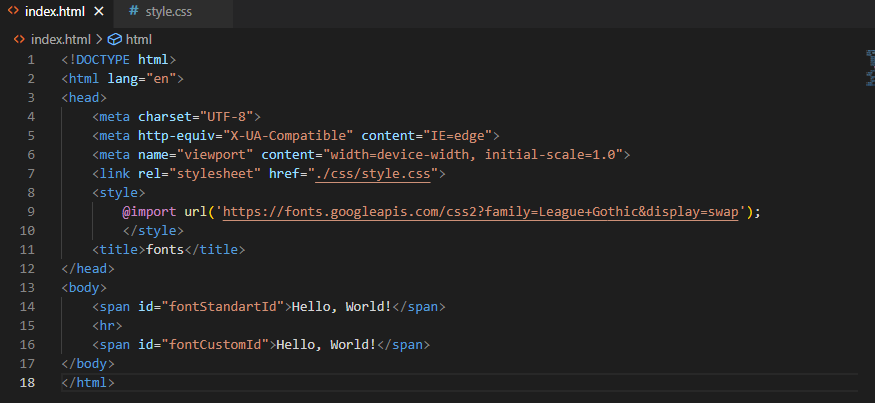
1. eot — embedded opentype (.eot)
2. ttf — truetype (.ttf)
3. otf — opentype (.ttf, .otf)
4. svg — svg-шрифты (.svg, .svgz)
5. woff — web open font format (.woff)

Google fonts:

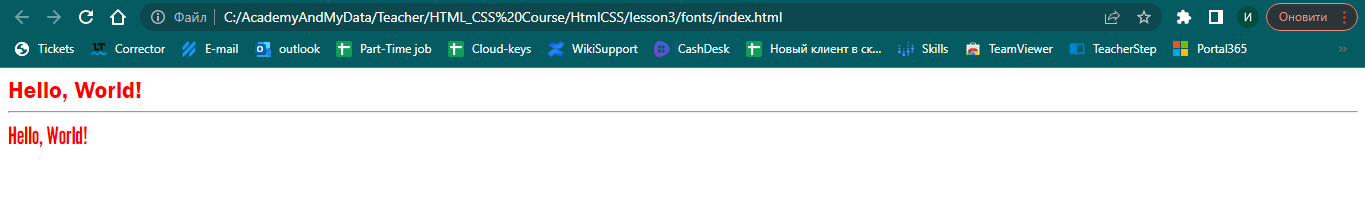
<https://fonts.google.com/>











<https://ru.wikibooks.org/wiki/CSS/%D0%A8%D1%80%D0%B8%D1%84%D1%82%D1%8B_%D0%B8_%D1%82%D0%B5%D0%BA%D1%81%D1%82>

Текст

### text-align

Выравнивание текста

Возможные значения:

1. left — по левому краю (значение по умолчанию)
2. right — по правому краю
3. center — по центру
4. justify — по ширине

### letter-spacing/word-spacing

Интервал между буквами/словами

1. normal — нормальное значение (по умолчанию)
2. length — значение в единицах измерения CSS (допускаются отрицательные значения)

### text-decoration

Оформление текста

Возможные значения:

1. underline — подчёркнутый текст
2. overline — линия над текстом
3. line-through — перечёркнутый текст
4. blink — мигающий текст
5. none — отмена всех эффектов (значение по умолчанию)

### text-indent

Отступ первой строки

1. length — фиксированное значение в единицах измерения CSS (значение по умолчанию равно 0)
2. % — проценты от ширины строки

### text-transform

Трансформация текста

Возможные значения:

1. none — нет трансформации (значение по умолчанию)
2. uppercase — преобразование всех строчных символов в заглавные
3. lowercase — преобразование всех заглавных символов в строчные
4. capitalize — преобразование первой буквы каждого слова в заглавную

### vertical-align

Выравнивание элемента по вертикали

Применяется только к строчным элементам и к ячейкам таблицы

Возможные значения для строчных элементов:

1. baseline — по базовой линии (значение по умолчанию)
2. sub — нижний индекс
3. super — верхний индекс
4. top — верх элемента выравнивается с самым высоким элементом строки
5. text-top — верх элемента выравнивается с верхом шрифта родительского элемента
6. middle — по середине
7. bottom — по нижней части
8. text-bottom — низ элемента выравнивается с низом шрифта родительского элемента
9. проценты
10. величины CSS

Для ячеек таблицы:

1. baseline — содержимое ячейки по базовой линии (значение по умолчанию)
2. top — по верхнему краю
3. bottom — по нижнему краю
4. middle — по середине ячейки таблицы

Для строчных элементов это свойство выравнивает сам элемент, а не его содержимое. Для ячеек таблицы — наоборот

### white-space

Пробелы между словами

Возможные значения:

1. normal — обычное поведение (значение по умолчанию)
2. pre — пробелы и переносы учитываются
3. nowrap — пробелы и переносы не учитываются
4. pre-wrap — пробелы учитываются, текст переносится
5. pre-line — пробелы не учитываются, текст переносится

### text-shadow

Добавление тени к тексту (CSS3)

Возможные значения:

1. h-shadow — смещение тени по горизонтали
2. v-shadow — смещение тени по вертикали
3. blur — радиус размытия тени
4. color — цвет тени
5. none — нет тени (значение по умолчанию)

### text-overflow

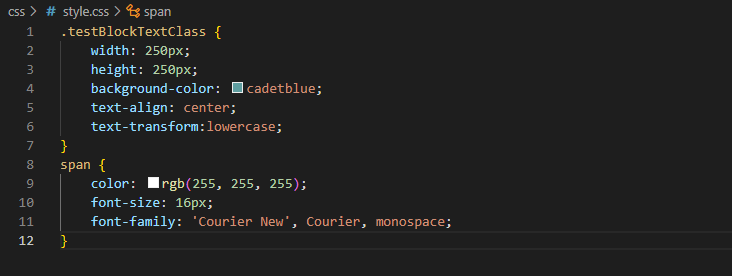
Видимость текста, если он не влезает в ширину блока (CSS3)

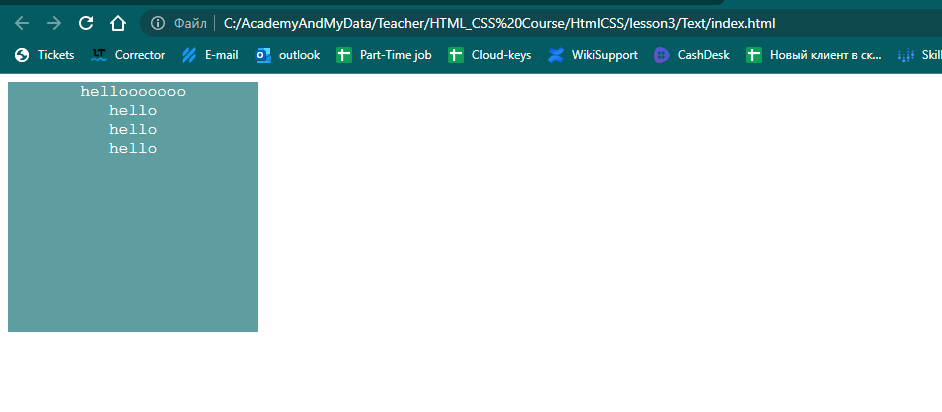
Возможные значения:

1. clip — текст обрезается по размеру блока (значение по умолчанию)
2. ellipsis — текст обрезается с добавлением многоточия

<https://ru.wikibooks.org/wiki/CSS/%D0%A8%D1%80%D0%B8%D1%84%D1%82%D1%8B_%D0%B8_%D1%82%D0%B5%D0%BA%D1%81%D1%82>







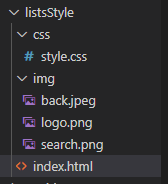
Transition

Transition позволяет определять переходное состояние между двумя состояниями элемента. Различные состояния могут быть определены с помощью [псевдоклассов](https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-classes), таких как [:hover](https://developer.mozilla.org/ru/docs/Web/CSS/:hover) или [:active](https://developer.mozilla.org/ru/docs/Web/CSS/:active) или установлены динамически с помощью JavaScript.

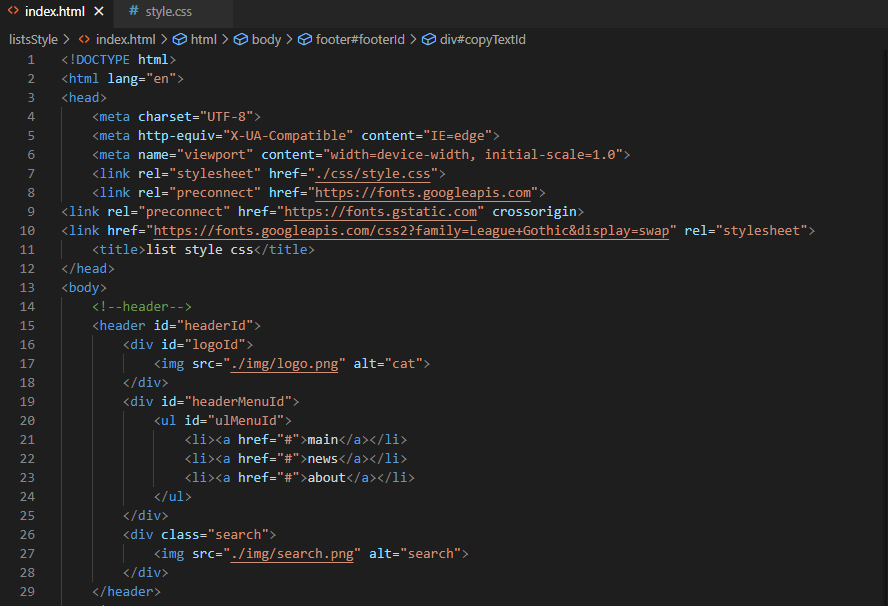


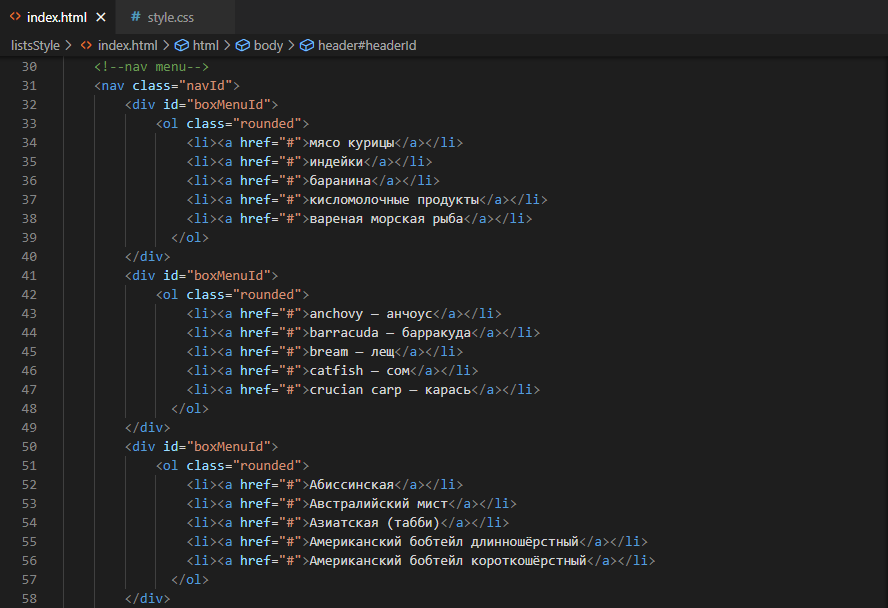
<https://developer.mozilla.org/ru/docs/Web/CSS/transition>

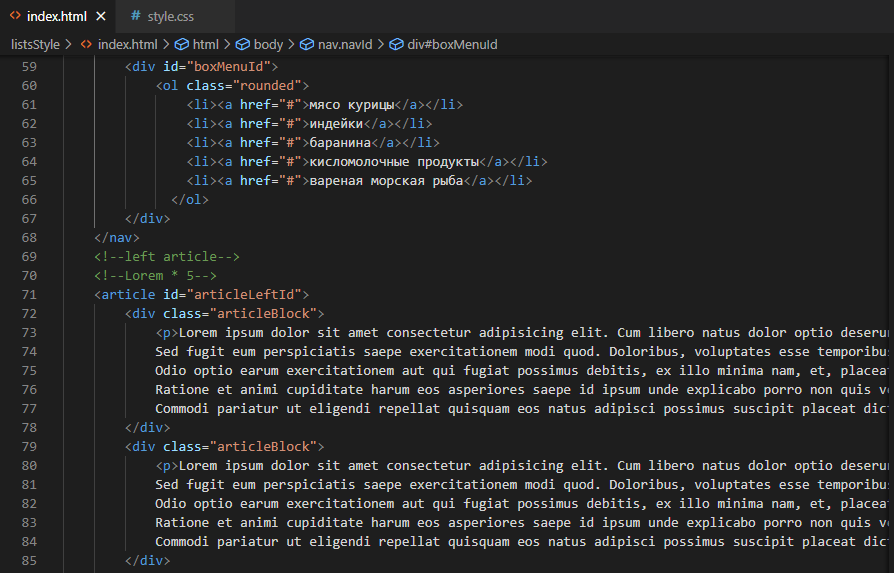
Практика:

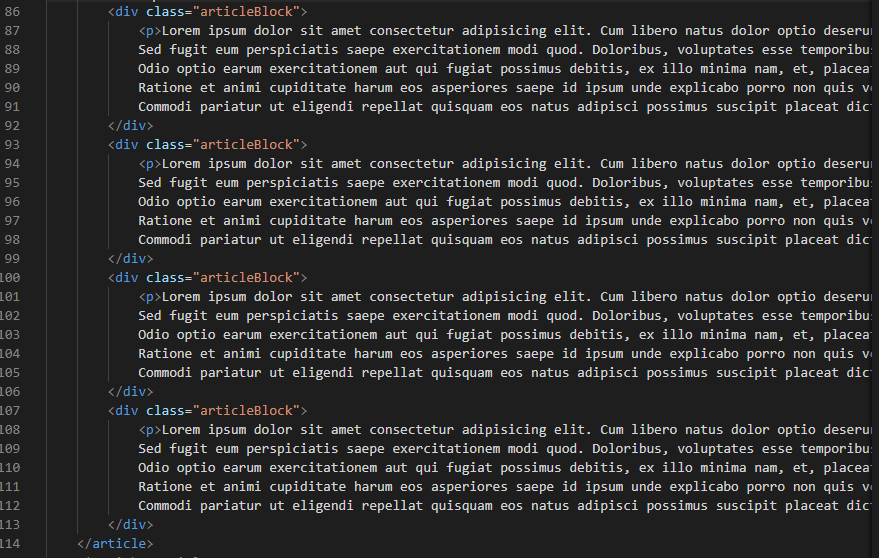


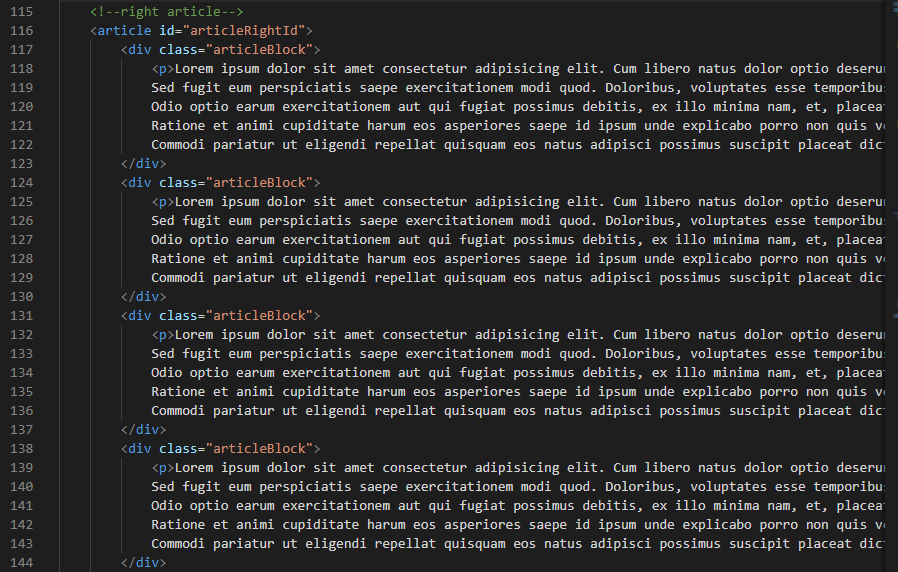
Html

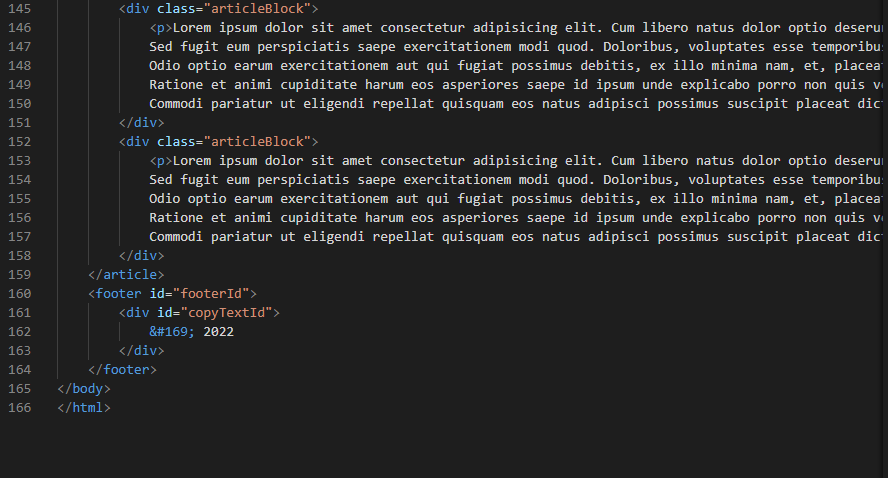




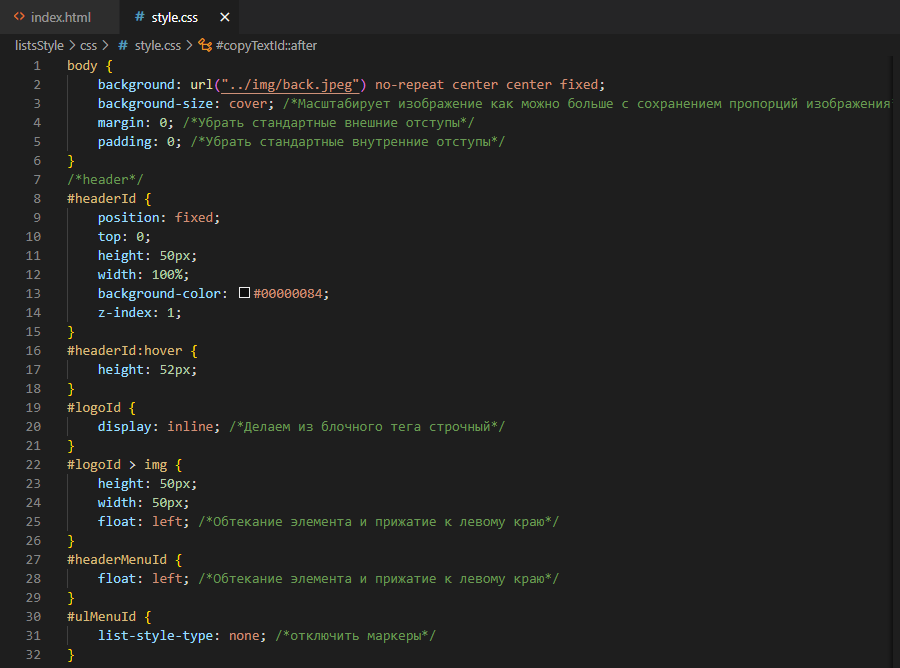


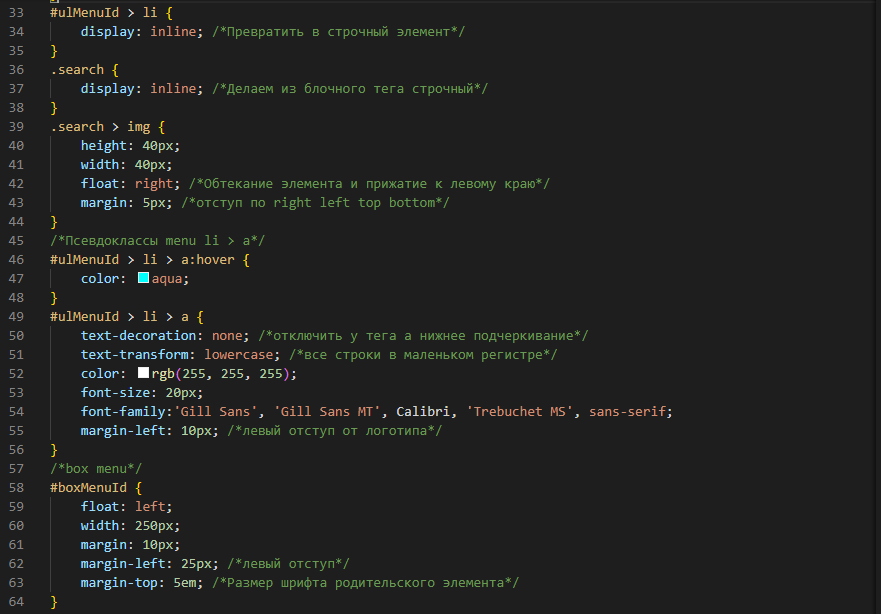


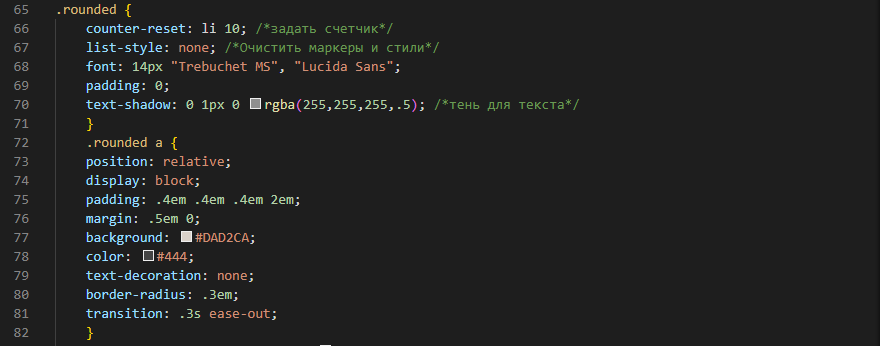


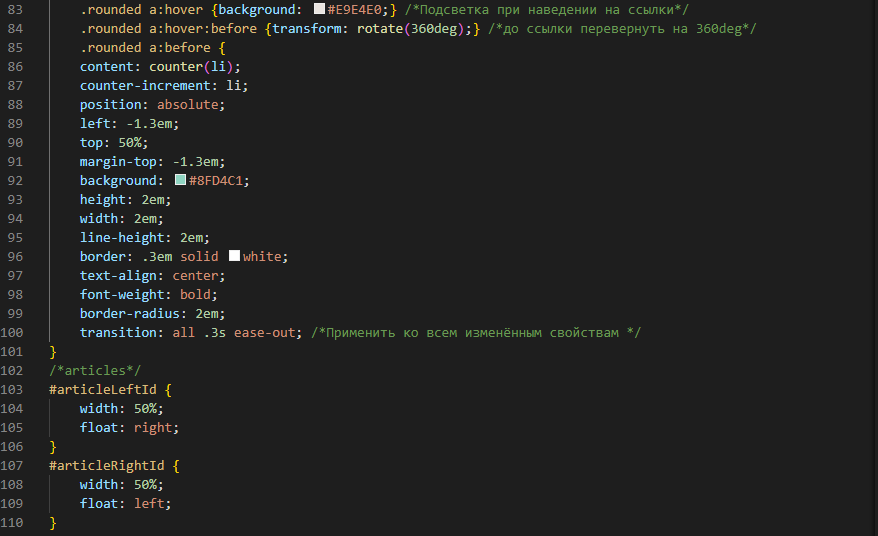


css



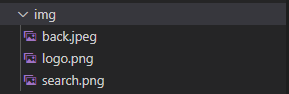


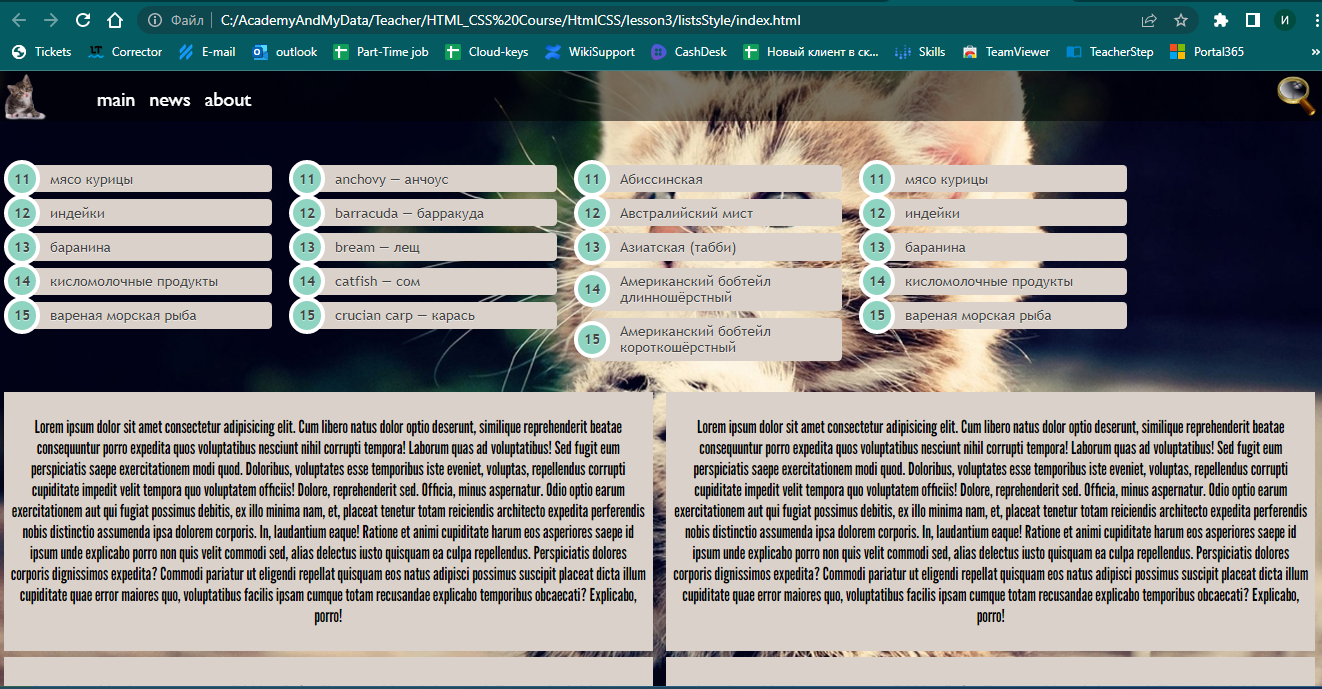


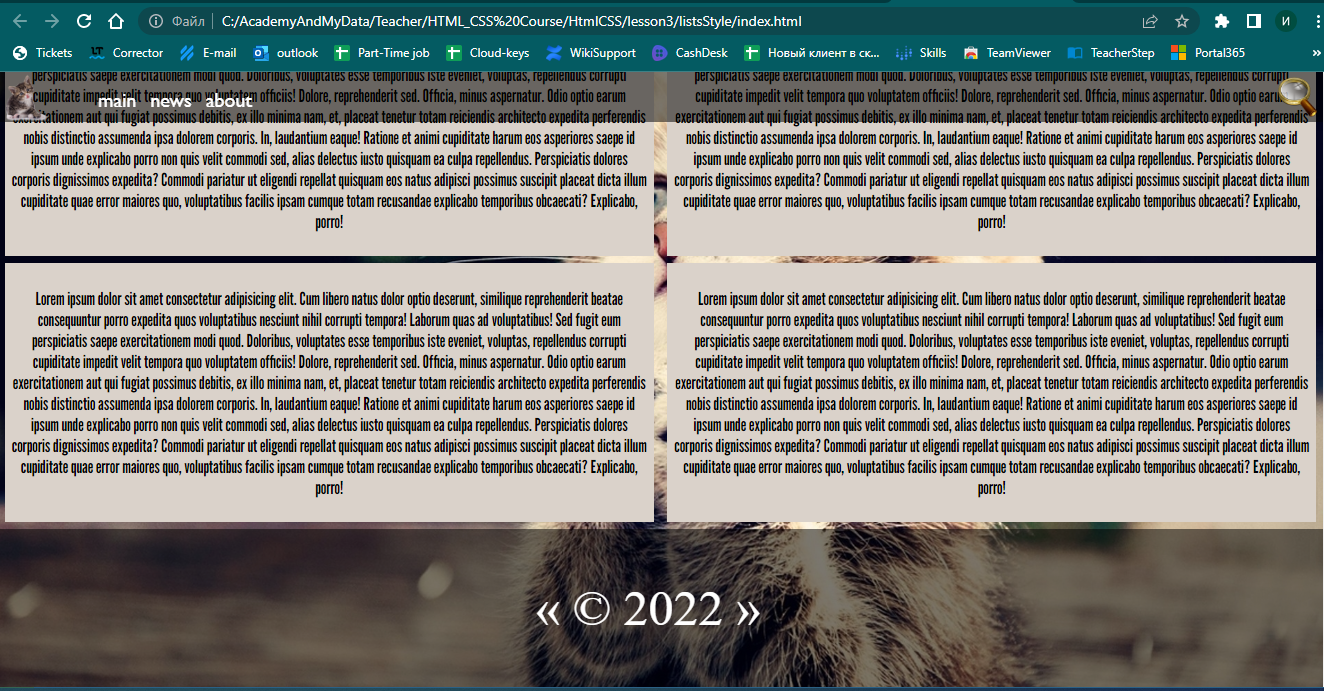




Img



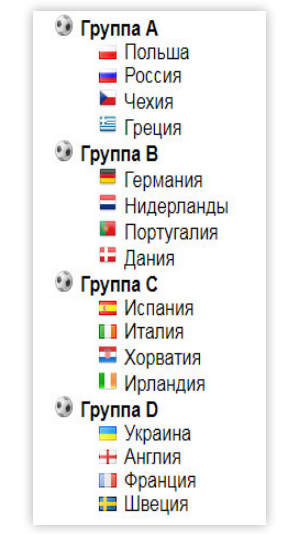
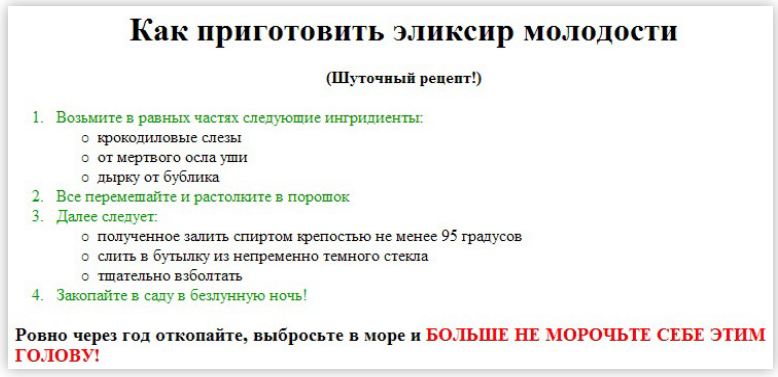
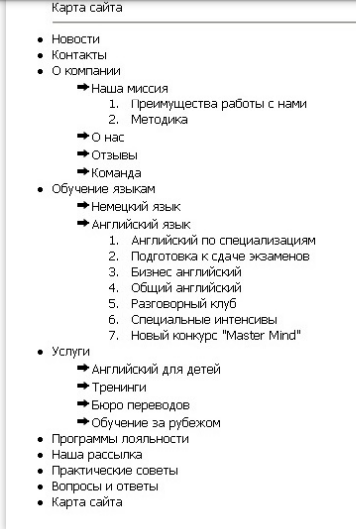




Материалы к уроку:

* 1. <https://guruweba.com/html/sozdanie-spiskov-vse-o-html-spiskakh/>
  2. <https://developer.mozilla.org/ru/docs/Learn/CSS/Styling_text/Styling_lists>
  3. <https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-elements>
  4. <https://developer.mozilla.org/ru/docs/Web/CSS/Pseudo-classes>
  5. <http://htmlbook.ru/css/cat/margin>
  6. <https://idg.net.ua/blog/uchebnik-css/ispolzovanie-css/margin-i-padding>
  7. <https://developer.mozilla.org/ru/docs/Web/CSS/margin>
  8. <https://developer.mozilla.org/ru/docs/Web/CSS/padding>
  9. <https://developer.mozilla.org/ru/docs/Learn/CSS/Styling_text/Fundamentals>
  10. <https://drive.google.com/file/d/11JdRKgw_GE6Tpc3a2RN6ctvqvAuQM7SY/view?usp=sharing> стр. 462 Позиционирование

Домашнее задание:

* + 1. 
    2. 
    3. 
    4. Написать практическое задание из раздела практика в методичке

Автор: Домбровский И.В. “Academy Step”   
Igor Dombrovsky telegram 0990356078

**dombrovskij\_i@itstep.academy**