

# Каскадные таблицы стилей

**CSS** (*Cascading Style Sheets*)



**Для оформления внешнего вида сайта  
используются стилевые правила языка CSS.**

**В соответствии с концепцией современной веб-разработки  
HTML-код должен содержать лишь теги и контент в них.**

# Каскадные таблицы стилей



## Преимущества CSS

- ✓ CSS дополняет язык HTML и значительно расширяет его возможности.
- ✓ Отделяя стиль представления документов от содержимого, CSS упрощает создание веб-страниц и обслуживание сайтов.
- ✓ За счет этого повышается и скорость разработки сайта
- ✓ Отладка сайта становится более быстрой и удобной.
- ✓ Появляется возможность одновременной работы с *одним* проектом сразу несколькими разработчиками: дизайнеру, верстальщику, программисту.



# Три способа вставить CSS



- ✓ Внешняя таблица стилей - стилевое описание html-элементов в отдельном файле, идеально подходит, когда стили должны применяться ко многим страницам

Внешние таблицы стилей хранятся в CSS файлах



- ✓ Внутренняя таблица стилей (в **head-секции**) - стилевое описание для всех идентичных элементов html-документа.

Задается с помощью парного тега **<style>**, который должен находиться в заголовке документа. Стили html-элементов внутри тега **<style>** задаются в соответствии с определенным синтаксисом.



- ✓ Встроенный стиль (**внутри HTML элемента**) - стилевое описание непосредственно в самом элементе (в открывающем теге). Действует только для этого тега.

# В чём заключается понятие каскадности?



том, что **стили**, подключенные к  
html-документу разными способами, **имеют**

## **приоритеты:**

1. сначала будут применены параметры форматирования, заданные во внешнем CSS-файле;
2. затем, прописанные в заголовке документа;
3. в последнюю очередь форматирование будет дополнено или изменено в соответствии с параметрами, объявленными непосредственно в тегах.



# Внешняя таблица стилей

## Как подключить?

- это файлы с расширением **.css**
- каждая html - страница должна ссылаться на таблицу стилей, используя тег **<link>** в **head-секции**

это в html - файле

```
<head>
```

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

```
</head>
```

отвечает за установку  
взаимосвязи html-документа и  
css-файла:

описывает тип данных в  
таблице стилей

имя подключаемого  
стилевого файла



## Подробнее о теге link

Теперь стилевой файл `test.css` нужно «подключить» к веб-странице с HTML-кодом. Для этого используется специальный тэг `<link>` (англ. *link* — связь), с помощью которого устанавливается связь с другими файлами. Этот тэг нужно расположить внутри контейнера `<head>` (заголовок страницы):

```
<head>
  <title>Страница с файлом стилей</title>
  <link rel="stylesheet" type="text/css"
    href="test.css">
</head>
```

У тэга `<link>` указаны три атрибута, причем сразу понятно, что `href` обозначает имя подключаемого стилевого файла.

Атрибут `rel` — это сокращение от английского *relation* — отношение; он определяет, какую роль играет файл `test.css`. В данном случае значение `stylesheet` говорит о том, что это таблица стилей (англ. *style sheet*). Атрибут `type` — это тип данных, значение `text/css` говорит о том, что это файл с каскадной таблицей стилей (CSS).

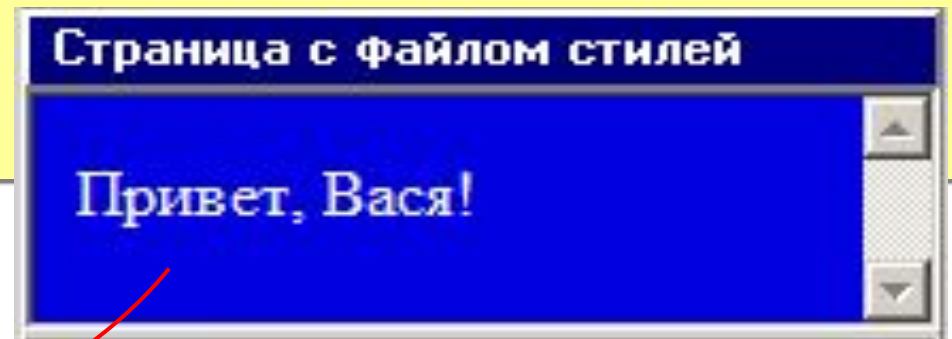
# Пример подключения css- файла

имя подключаемого  
стилевого файла

```
<html>
<head>
<title> Страница с файлом стилей</title>
<link rel="stylesheet" type="text/css href="mystyle.css">
</head>
  <body>
    Привет, Вася!
  </body>
</html>
```

**mystyle.css**

```
body {
  color: white;
  background: #0000E0;
}
```







# Синтаксис CSS

- Основное правило CSS состоит из двух главных частей: **селектор**, и один или более **объявлений**:



**Селектор** – это HTML-элемент, которому вы хотите придать определенный стиль.

**Каждое объявление** состоит из свойства и значения.

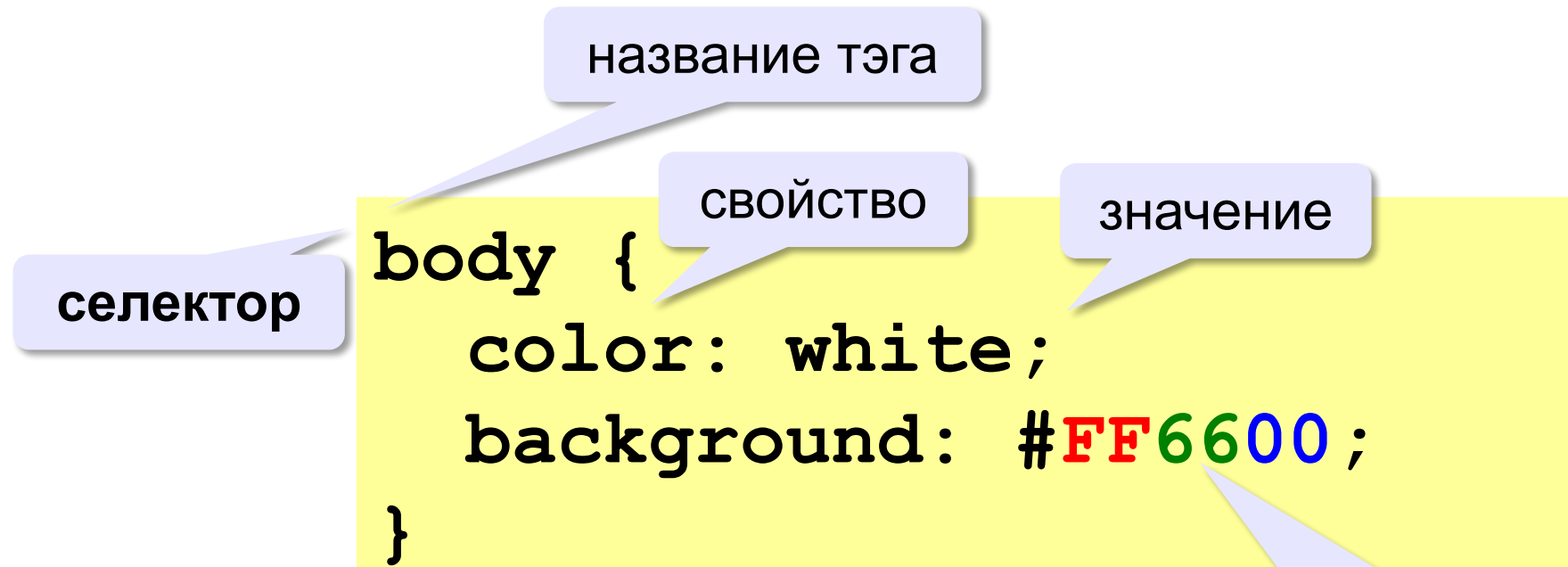
**Свойство** - это атрибут стиля, который вы хотите изменить.  
**Каждое свойство имеет значение.**

Комментарий начинается с "/\*", и заканчивается "\*/"



# Пример синтаксиса CSS для body

mystyle.css



`color` – цвет символов

`background` – цвет фона

$R = FF_{16} = 255$   
 $G = 66_{16} = 102$   
 $B = 0$

# Пример синтаксиса CSS для абзаца

Объявление CSS всегда оканчивается точкой с запятой,

селекто

p

```
p {color:red; text-align:center;}
```

свойств

о

значени

е

свойств

о

значени

е

Привет Мир!

Этот параграф оформлен с помощью CSS.

Группы объявлений заключаются в фигурные скобки:

# Стили: размеры, выравнивание

```
p {  
  background: #E6E6FF;  
  width: 80%;  
  height: 100px;  
  text-align: left;  
  text-indent: 20px;  
}
```

цвет фона

ширина

высота

абзацный  
отступ

выравнивание:  
**left**  
**center**  
**right**  
**justify**

20px

Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Curabitur felis velit, tempus sit amet cursus at, porta ut  
turpis. Sed augue tellus, tincidunt id ultricies at, tempor quis  
sem.

100px

80%

# Рамка, поля, отступы

толщина рамки

**solid** – сплошная  
**dashed** – штриховая  
**dotted** – точечная

рамка

отступы  
снаружи

отступы  
внутри

```
p {  
  background: #ccffcc;  
  border: 1px solid green;  
  margin: 0 40px 20px 40px;  
  padding: 5px;  
}
```

цвет

со всех сторон

сверху, справа,  
снизу, слева

40px



























40px

5px

20px

Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Curabitur felis velit, tempus sit amet cursus at, porta ut turpis.  
Sed augue tellus, tincidunt id ultricies at, tempor quis sem.

# border-style

1 пиксел	3 пиксела	5 пикселей	7 пикселей
			
			
			
			
			
			
			
			

**inset** Псевдотрехмерная рамка, при которой правая и нижняя граница осветляется, а левая и верхняя линии затемняются.

**outset** Псевдотрехмерная рамка, при которой левая и верхняя граница имеют более светлый оттенок, чем заданный цвет, а правая и нижняя линии затемняются.

## outline-style

Задаёт стиль внешней границы элемента. В отличие от линии, задаваемой через `border`, линия через `outline` отображается вокруг элемента, не влияя на ширину блока или его положение.

## outline-offset

Устанавливает расстояние между рамкой, созданной с помощью свойства [outline](#), и краем или границей элемента добавленной через [border](#).

```
border: 9px solid green; outline: 5px dotted; outline-offset: 3px; outline-color: #f92525;
```

# Значения для **margin** и **padding**

Значения могут быть указаны в любых единицах CSS – px, em, % и т. д.

Для **margin** и **padding** можно указывать сразу 1, 2, 3 или 4 значения.

От количества значений зависит, как распределяются настройки:

- **4 значения:** задаются отступы для всех сторон элемента в такой последовательности: сверху, справа, снизу, слева:

**padding:** 2px 4px 5px 10px;

4 1 2 3

- **3 значения:** задается отступ сначала для верхней стороны, потом одновременно для левой и правой, а затем – для нижней:

**padding:** 3px 6px 9px;

1 3 2 3

- **2 значения:** задаются отступы сначала одновременно от верхней и нижней стороны, а затем – одновременно для левой и правой

**padding:** 6px 12px;

1 1 2 2

- **1 значение:** задаются одинаковые отступы для всех сторон элемента

Те же правила касаются и свойства **margin** CSS.

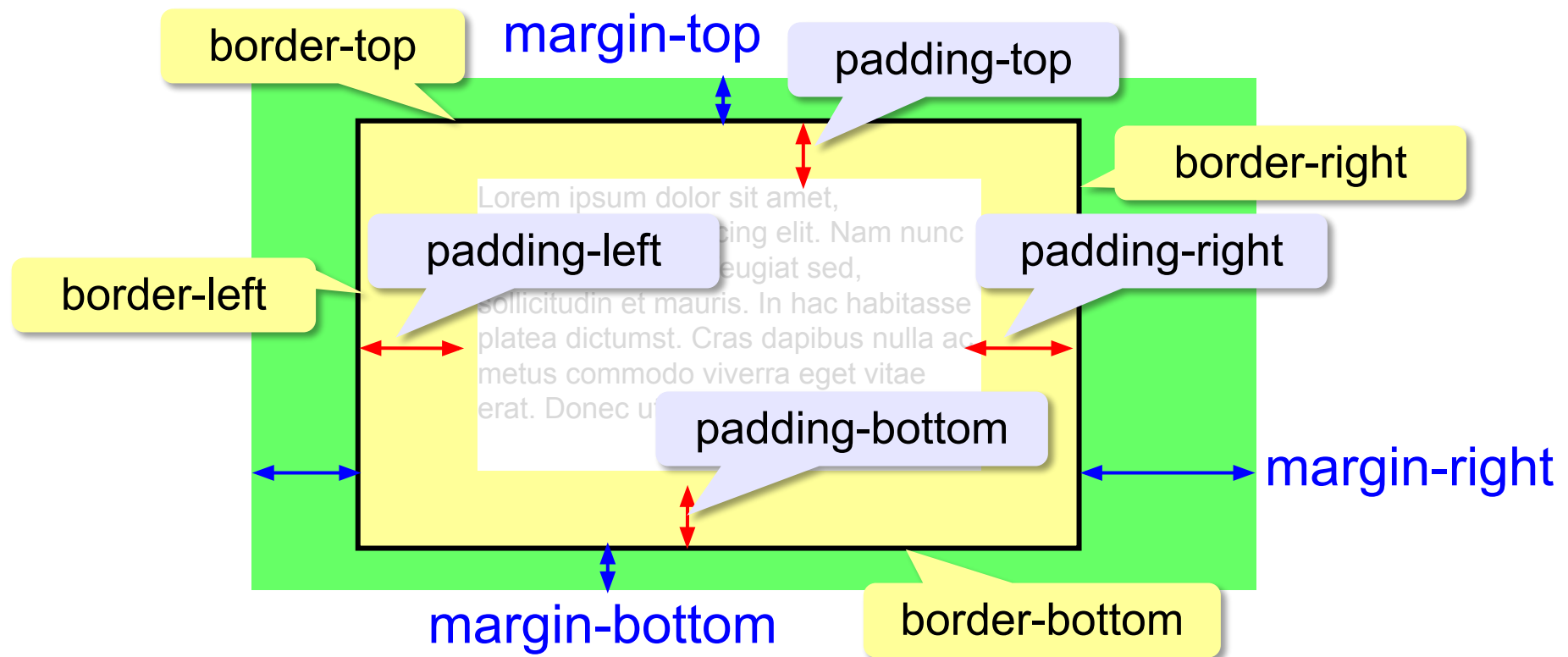
1 1 1 1

Отступы:

- **padding-top:** значение;
- **padding-right:** значение;
- **padding-bottom:** значение;
- **padding-left:** значение;

Отступы:

- **padding-top:** значение;
- **padding-right:** значение;
- **padding-bottom:** значение;
- **padding-left:** значение;





# Стили для элементов: шрифты

для всех  
абзацев

семейство  
шрифтов

**serif** – с засечками  
**sans-serif** – без засечек  
**monospace** – моноширинный

```
p {  
  font-family: Arial, sans-serif;  
  font-size: 20px;  
  font-style: italic;  
  font-weight: bold;  
}
```

размер в пикселях

**normal** – обычный  
**italic** – курсив

**normal** – обычный  
**bold** – жирный

# Внутренняя таблица стилей

## Как подключить?

Внутренняя таблица стилей используется когда отдельный документ имеет уникальный стиль.

Внутренние стили - в **head**-секции

```
<head>
```

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

```
  hr {color:sienna;}
```

```
  p {margin-left:20px;}
```

```
  body {background-image:url("images/back40.gif");}
```

```
</style>
```

```
</head>
```



# Встроенные стили.

# Как подключить?

Используется непосредственно в самом элементе (в открывающем теге).  
Действует только для этого тега.

Используете атрибут **style** в соответствующем теге.  
Атрибут style может содержать любое свойство CSS.

```
<p style="color:sienna; margin-left:20px">  
    Например, для какого - либо абзаца.  
</p>
```

Пример показывает как  
изменить цвет и левый отступ  
абзаца

# Текстовые свойства

Свойство	Описание
<b><u>color</u></b>	Устанавливает цвет текста
<b><u>direction</u></b>	Устанавливает направление текста/направление при вводе текста
<b><u>letter-spacing</u></b>	Увеличивает или уменьшает расстояние между символами в тексте
<b><u>line-height</u></b>	Устанавливает высоту строки
<b><u>text-align</u></b>	Указывает горизонтальное выравнивание текста
<b><u>text-decoration</u></b>	Указывает декорации, добавляемые к тексту
<b><u>text-indent</u></b>	Указывает отступ первой линии в текстовом блоке
<b><u>text-shadow</u></b>	Устанавливает эффект тени, добавляемой к тексту
<b><u>text-transform</u></b>	Контролирует регистр текста
<b><u>vertical-align</u></b>	Устанавливает вертикальное выравнивание элемента
<b><u>white-space</u></b>	Указывает как будет обрабатываться пустое пространство в элементе
<b><u>word-spacing</u></b>	Увеличивает или уменьшает расстояние между словами в тексте

# Декорация Текста

```
a { text-decoration: none; }
```

Свойство text-decoration в основном используется для того, чтобы удалить подчеркивание ссылок для целей дизайна:

```
h1 { text-decoration: wavy overline lime; }
```

Замки Беларуси

```
h1 { text-decoration: line-through; }
```

~~Замки Беларуси~~

```
h1 { text-decoration: underline red; }
```

Замки Беларусь

```
h1 { text-decoration: dashed underline overline; }
```

~~Замки Беларусь~~

```
h1 { text-decoration: solid underline purple 4px; }
```

Замки Беларусь

# Трансформация текста

Свойство text-transform используется, чтобы изменять регистр букв в тексте.

Все буквы  
заглавные

```
p.uppercase {text-transform:uppercase;}  
p.lowercase {text-transform:lowercase;}  
p.capitalize {text-transform:capitalize;}
```

Заглавные только первые буквы  
в каждом слове

Все буквы малые

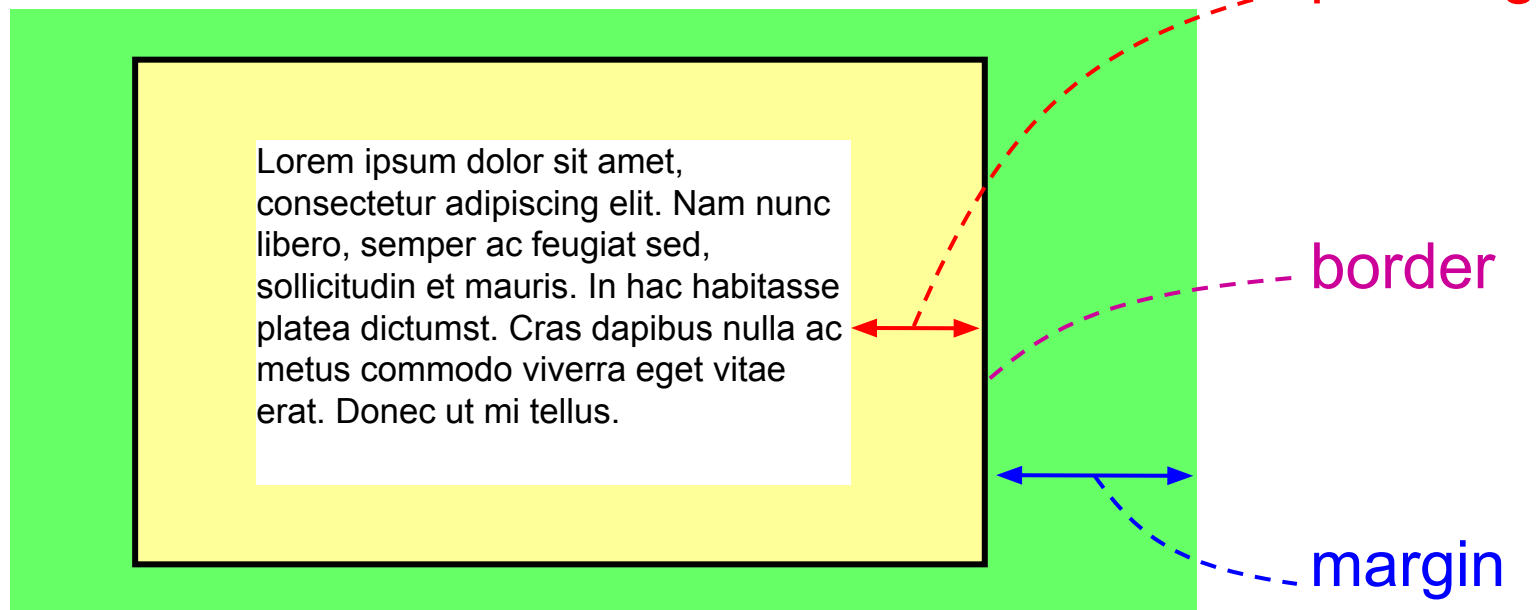
# Что такое блок (`div`)?

Тег **div** для группировки  
тегов

```
<div>  
...  
</div>
```

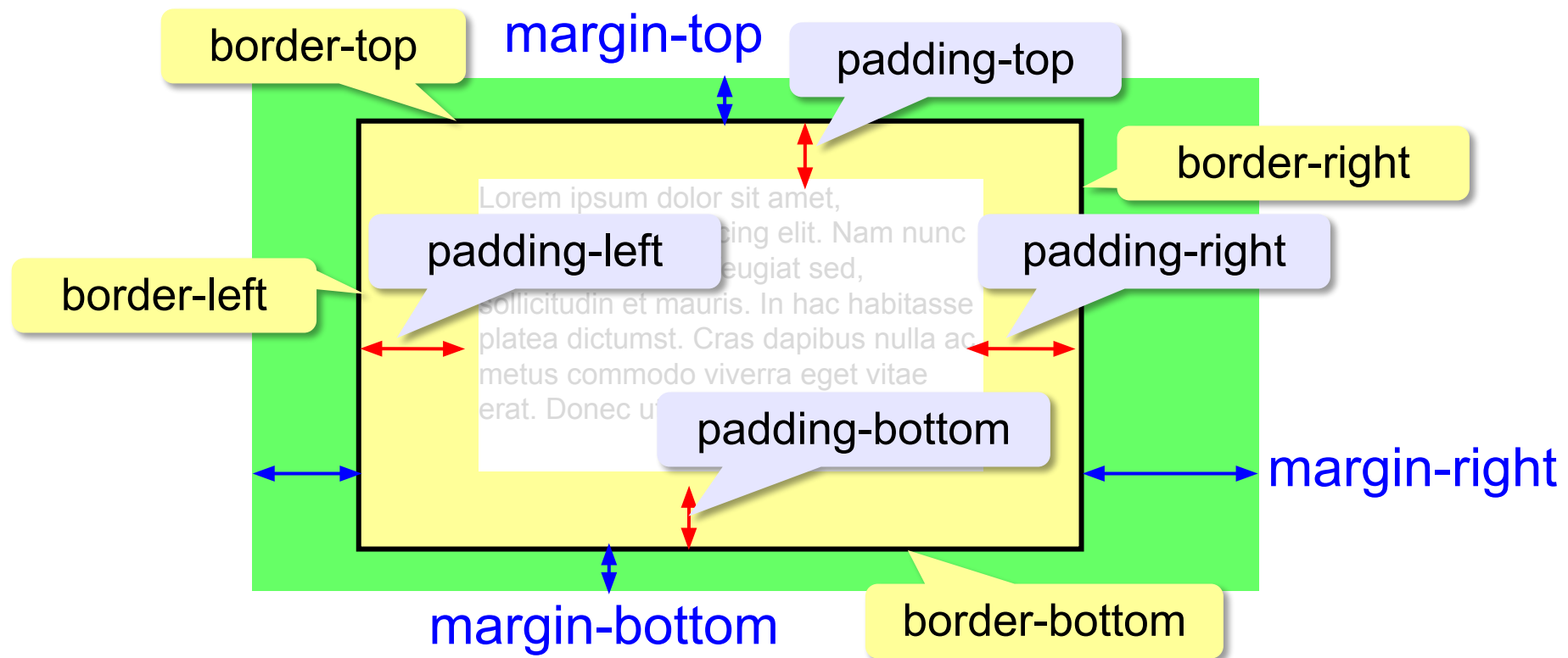
```
<p>  
<table>  
<a>  
<img>  
...
```

## свойства блока:





# Свойства CSS, устанавливаемые для блоков



```
.box {width: 350px; border: 10px solid black;}
```

# Свойства CSS, устанавливаемые для блоков

## Настройка ширины контейнера

По умолчанию `width: 100%;`

```
div{  
    width: 1170px;  
}
```

На самом деле, свойство **width** для настройки ширины контейнеров использовать **не рекомендуется**. Т.к. это приводит к необходимости **горизонтального скроллинга страницы** в том случае, если ширина области окна браузера, доступной для вывода страницы, меньше 1170px.

**Решить эту проблему можно, воспользовавшись свойством **max-width****

```
div{  
    max-width: 1170px;  
}
```

В этом случае – не будет горизонтальной полосы прокрутки

## Выравнивание контейнера по центру страницы

```
.div {  
    max-width: 1170px;  
    margin: 0 auto;  
}
```

# Свойства CSS, устанавливаемые для блоков

## Настройка левого и правого внутренних отступов

```
.wrapper { max-width: 1170px;  
            margin-left: auto;  
            margin-right: auto;  
            padding-left: 16px;  
            padding-right: 16px;  
        }
```

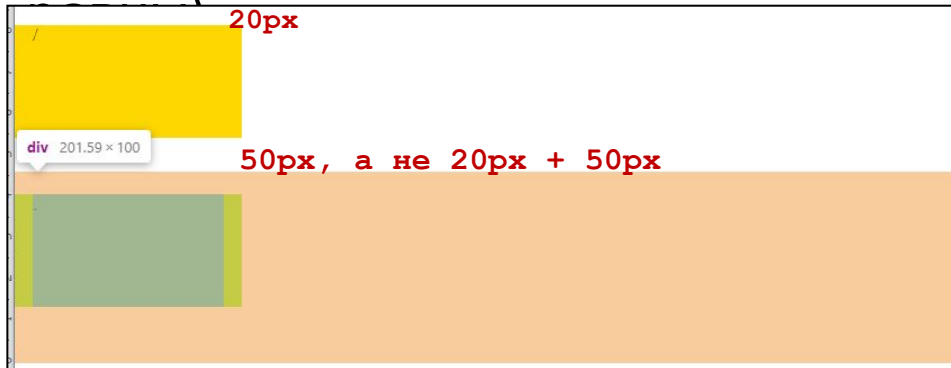
Внутренние отступы — это нечто вроде защитного механизма, который не даёт границам контейнера прижиматься к границам области просмотра даже в том случае, когда область просмотра меньше, чем максимальная ширина контейнера.

# Свойства CSS, устанавливаемые для блоков

## Схлопывание внешних

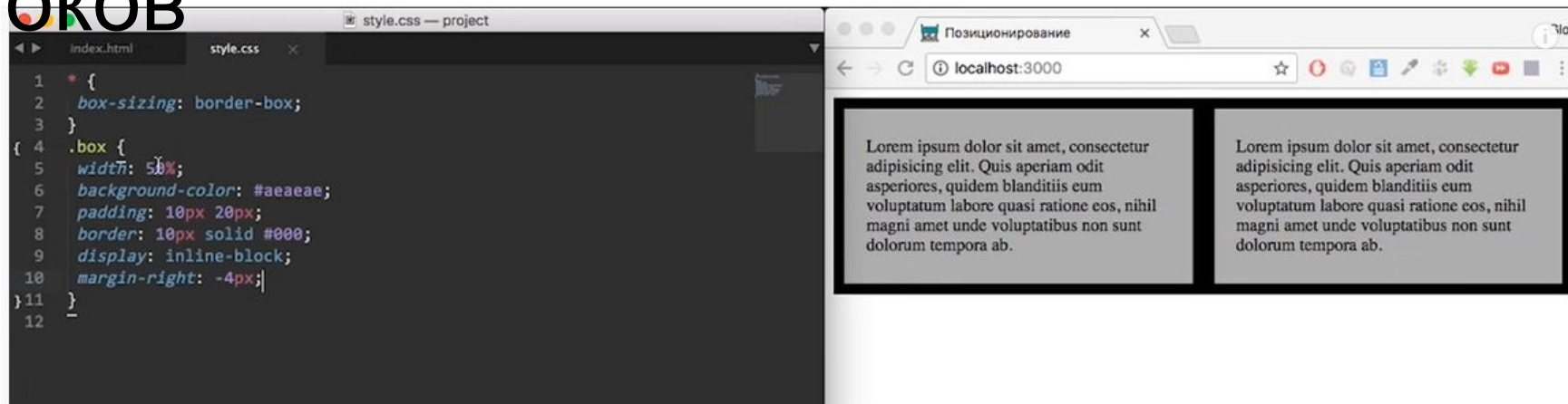
### отступов

Отступы [margin-top](#) и [margin-bottom](#) иногда объединяются в один, с размером **равным наибольшему из них** (или размеру одного, если они



```
div{
    background-color: gold;
    width:20%;
    padding-left: 16px;
    padding-right: 16px;
    margin-bottom: 50px;
    margin-top:20px;
}
```

# Свойства CSS, устанавливаемые для блоков



Если сделать два блока, то они расположатся друг под другом. Но если сделать **display: block-inline**, то они станут в линию. Но между ними будет – отступ. И если эти блока будут – по 50%, то они не станут в линию, и тогда можно этот отступ убрать так:

1 способ – **margin-right: -4px;**

2 способ-

Убрать **все** пробелы между блоками

```
<div>  
  <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Laudantium rerum  
  sapiente facilis debitis minus ut consequatur, aliquid quam quisquam delectus culpa  
  exeretur adipisicing elit. Ipsam maioresima aliquid, odit voluptates, exercitationem  
  necessitatibus quo, voluptatum aliquam dolor alias minima aliquid, citationem hic  
  aliquam eveniet soluta repellendus voluptas molestias sint.</p>  
  
</div><div>  
  <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Laudantium rerum  
  sapiente facilis debitis minus ut consequatur, aliquid quam quisquam delectus culpa  
  exeretur adipisicing elit. Ipsam maioresima aliquid, odit voluptates, exercitationem  
  necessitatibus quo, voluptatum aliquam dolor alias minima aliquid, citationem hic  
  aliquam eveniet soluta repellendus voluptas molestias sint.</p>  
  
</div>
```