

< Teach  
Me  
Skills />

# Lesson 2

Знакомство с CSS



# Знакомство с CSS

- Работа с размерами элементов: width, height
- Работа с цветом в CSS: числовые значения, модель RGB, RGBA
- Работа с градиентом
- Работа с физическими свойствами отступов
- Работа с физическими свойствами полей
- Изменение размеров блочной модели
- Изучение свойства переполнения блока обертки
- Изучение сброса стилей в CSS
- Стилизация курсора
- Работа со шрифтами
- Использование БЭМ-методологии в проекте
- Изучить правила формирования имен БЭМ-сущностей

# Размеры элементов

Ширина (width):

```
.item { width: 400px; } .element { width: 50%; } .block { width: calc(70%/3) }
```



# Размеры элементов

Ширина (max-width), (min-width):

```
.item { max-width: 1170px; } .element { max-width: 50%; } .block { max-width: calc(70%/3) }  
.item { min-width: 350px; } .element { min-width: 25%; } .block { min-width: calc(70%/3) }
```



# Размеры элементов

Высота (height):

```
.item { height: 300px; } .element { height: 50%; } .block { height: calc(70%/3) }
```



# Размеры элементов

Высота (max-height), (min-height):

```
.item { max-height: 900px; } .element { max-height: 50%; } .block { max-height: calc(70%/3) }  
.item { min-height: 200px; } .element { min-height: 25%; } .block { min-width: calc(70%/3) }
```



# Размеры элементов

## Важно!

Как правило, высота элемента складывается из высоты контентной части. Т.е. размера шрифта, отступов, полей, размеров изображений, кнопок и прочих элементов родительского блока.

Таким образом, так как высота является динамической величиной и может изменяться за счет количества контента, принудительно, родительским блокам, высота не задается.



# Работа с цветом в CSS

По названию цвета:

```
.item { background-color: red; color: yellow; border: 1px solid coral; }
```





# Работа с цветом в CSS

HEX кодировка:

```
.item { background-color: #ff0000; color: #ffff00; border: 1px solid #00FFFF; }
```



# Работа с цветом в CSS

## RGB кодировка:

Аббревиатура английских слов **red**, **green**, **blue** — красный, зелёный, синий.

Если смешивать эти цвета в разных сочетаниях, то можно получить все цвета радуги.



```
.item {  
  background-color: rgb(255, 0, 0);  
  color: rgb(255, 255, 0);  
  border: 1px solid rgb(0, 255, 255);  
}
```

# Работа с цветом в CSS

## RGBA кодировка:

Цветовая модель RGB включает составляющую альфа, позволяющую задать непрозрачность цвета.

Это значит, что можно добавить четвертое значение (от 1 до 0), чтобы задать уровень непрозрачности данного RGB-цвета.



```
.item {  
  background-color: rgba(255, 0, 0, 0.8);  
  color: rgba(255, 255, 0, 0.78);  
  border: 1px solid rgba(0, 255, 255, 0.479);  
}
```

# Работа с цветом в CSS

## HSL (HSLA) кодировка:

HSL (от англ. hue, saturation, lightness (intensity)) — цветовая модель, в которой цветовыми координатами являются тон, насыщенность и светлота.



```
.item {  
  background-color: hsl(0, 100%, 50%);  
  color: hsla(60, 100%, 50%, 0.733);  
  border: 1px solid hsla(180, 100%, 50%, 0.479);  
}
```

HSL (HSLA)

# Градиент

**CSS** - градиенты - последовательный переход между двумя и более цветами.  
Типы градиентов: линейный `linear-gradient`, радиальный `radial-gradient`.

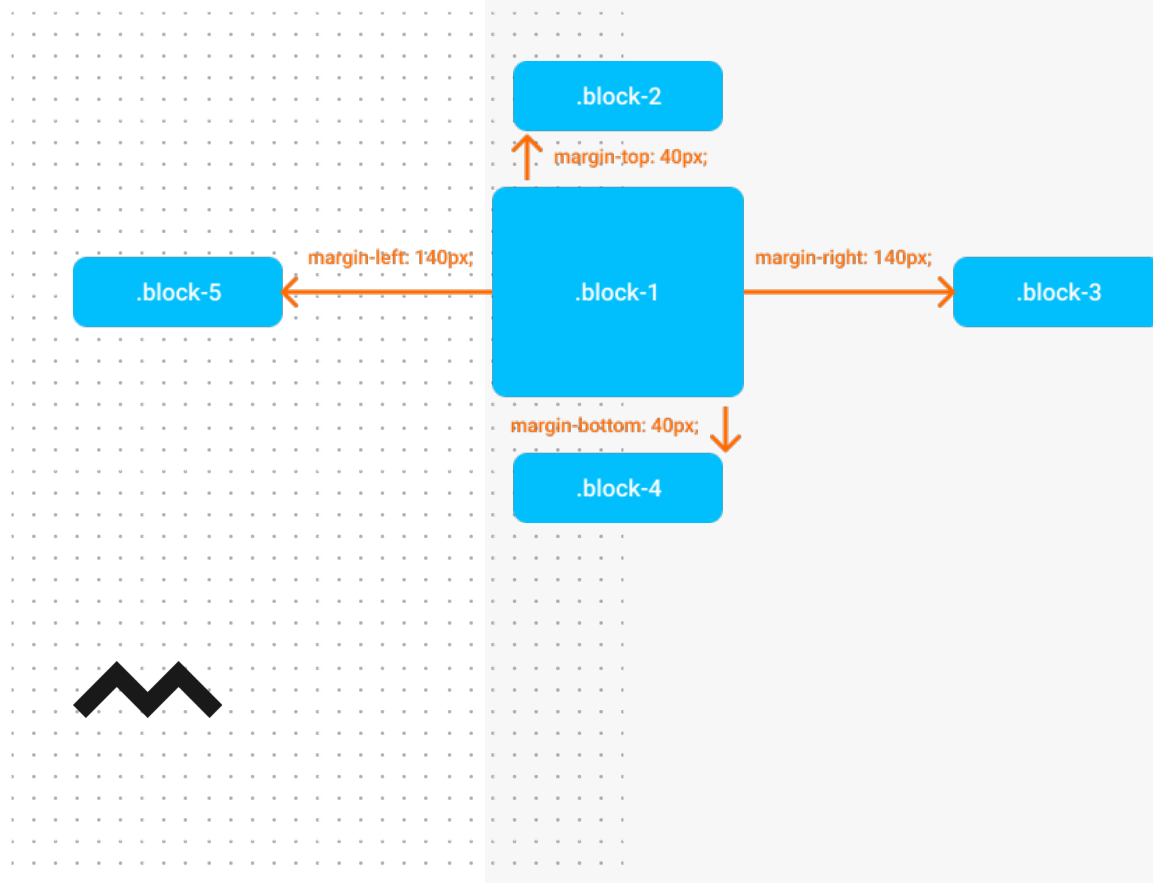
Больше о градиентах [тут](#)

```
.simple-linear { background: linear-gradient(coral, yellow); }  
.simple-radial { background: radial-gradient(purple, blue); }
```



# Работа с физическими свойствами отступов

**margin** - устанавливает величину отступа от каждого края элемента.



```
.elem {  
  margin-top: 10px;  
  margin-right: 20px;  
  margin-bottom: 15px;  
  margin-left: 20px;  
}
```

margin

# Работа с физическими свойствами отступов

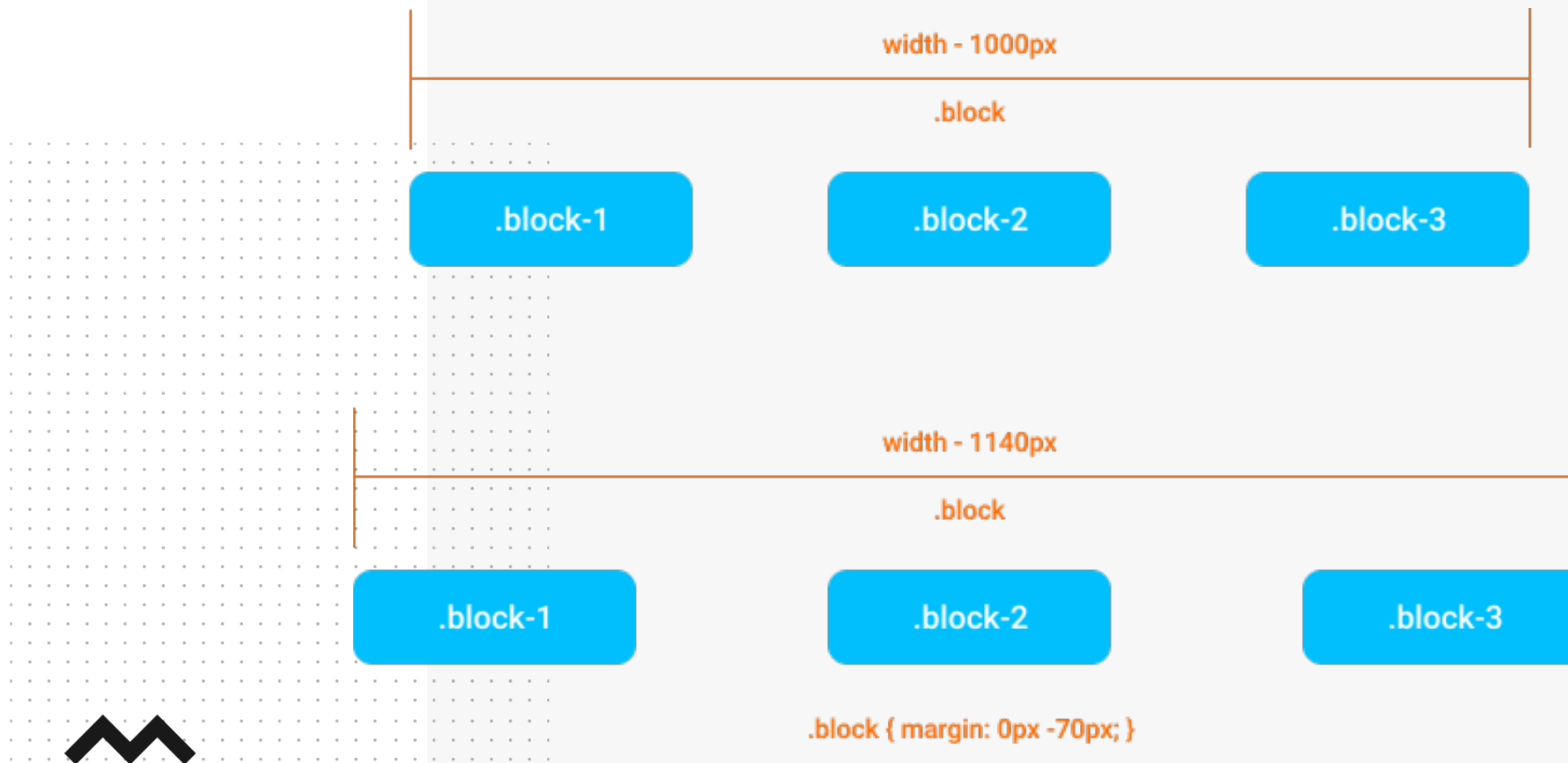
## Группировка значений

```
.block {  
  margin: 10px 20px 15px 25px; /*(top) (right) (bottom) (left)*/  
  margin: 10px 20px 15px; /*(top) (right/left) (bottom)*/  
  margin: 10px 20px; /*(top/bottom) (right/left)*/  
  margin: 10px; /*(top/bottom/right/left)*/  
}
```

margin

# Работа с физическими свойствами отступов

margin с отрицательным значением

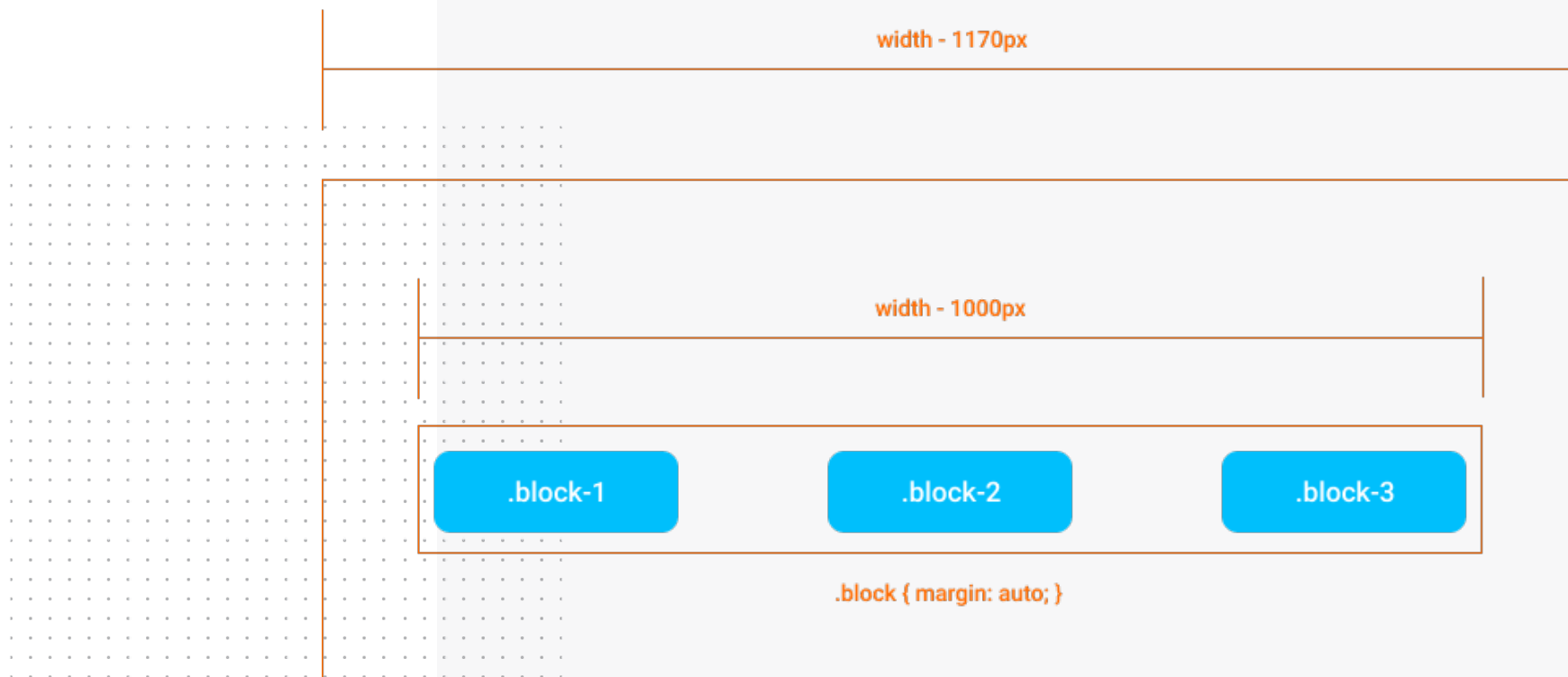


margin



# Работа с физическими свойствами отступов

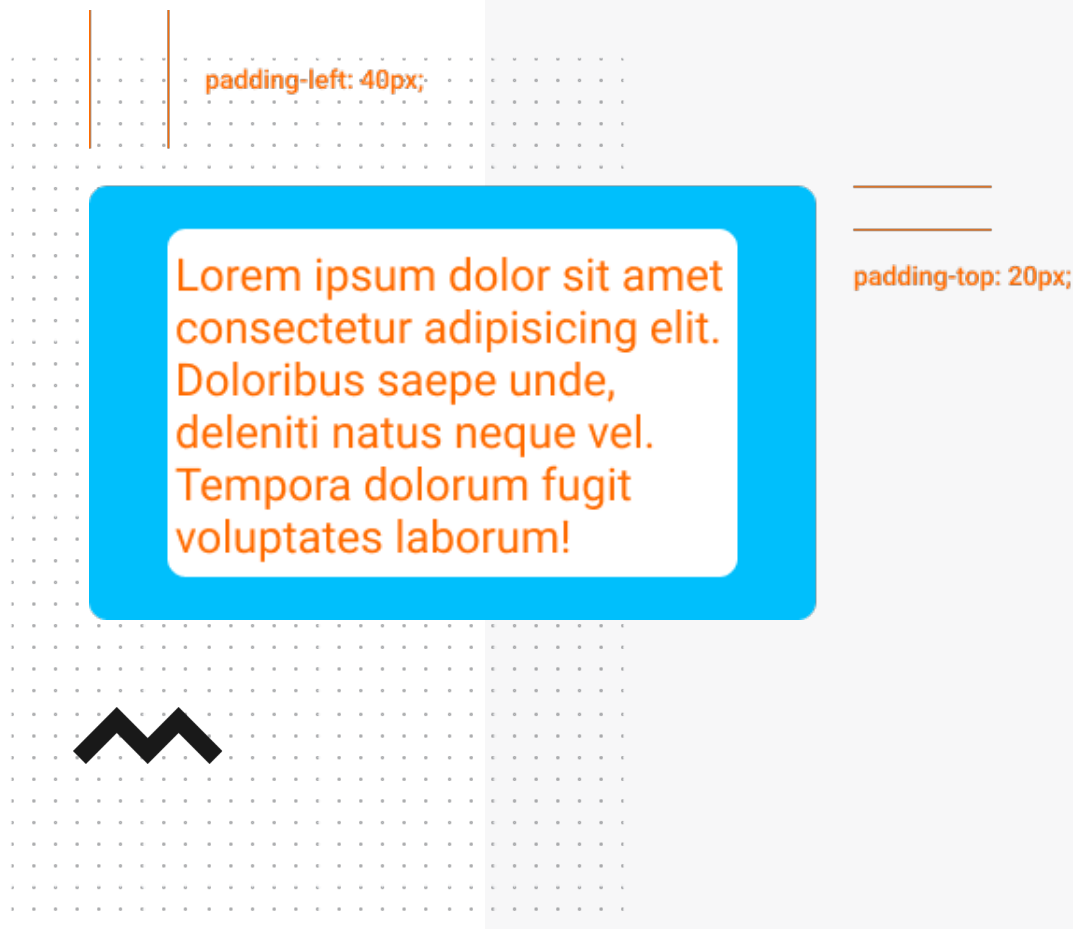
margin: auto;



margin

# Работа с физическими свойствами полей

**padding** - для полей (padding) применимы такие же правила группировки значений. Разница между ними в том, что margin - это отступ (дистанция), а padding и его border является его частью.

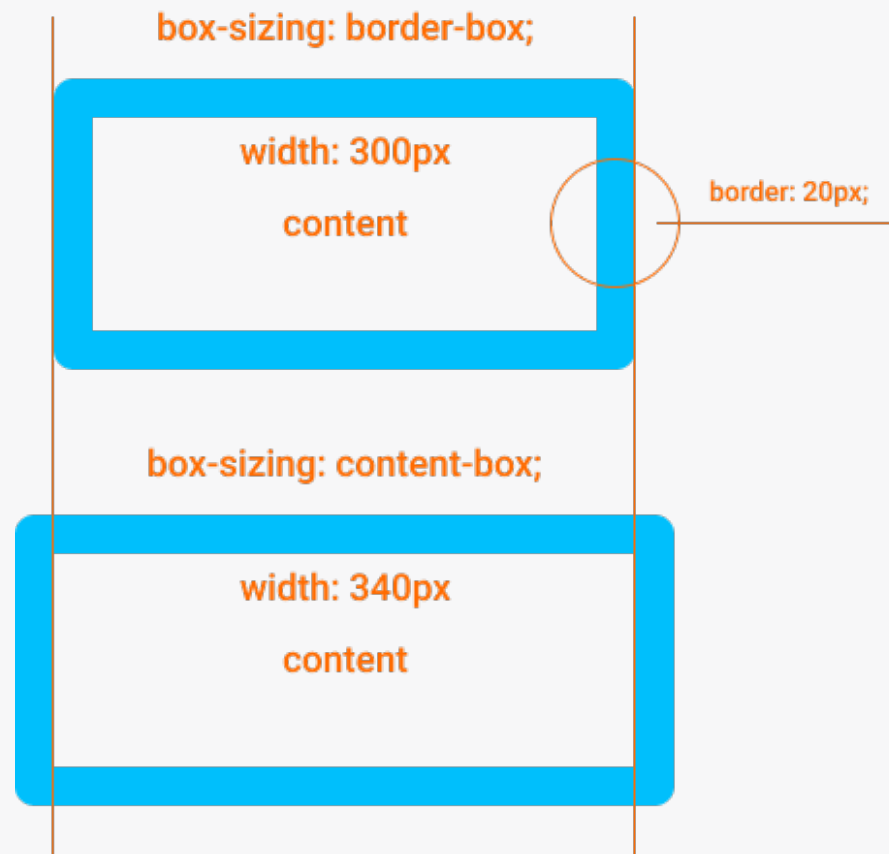


padding

# Изменения размеров блочной модели

**box-sizing** - применяется для изменения алгоритма расчёта ширины и высоты элемента.

Свойство **box-sizing** позволяет изменить этот алгоритм, чтобы свойства **width** и **height** задавали размеры не содержимого, а размеры блока.



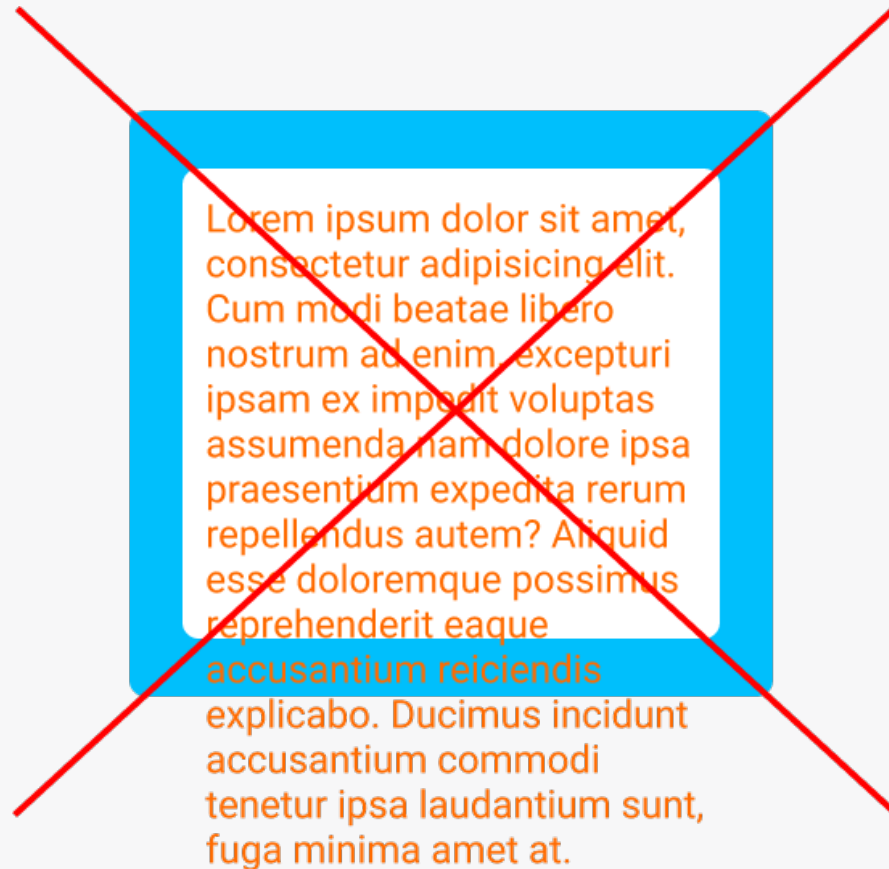
# Overflow

**overflow** - управление контентом блока, позволяет обрезать видимую часть, или добавляет скролл, если контента больше, чем величина блока.

Значения:

`overflow: auto | hidden | scroll | visible;`

**Помним!** Высота элемента складывается из высоты контента.



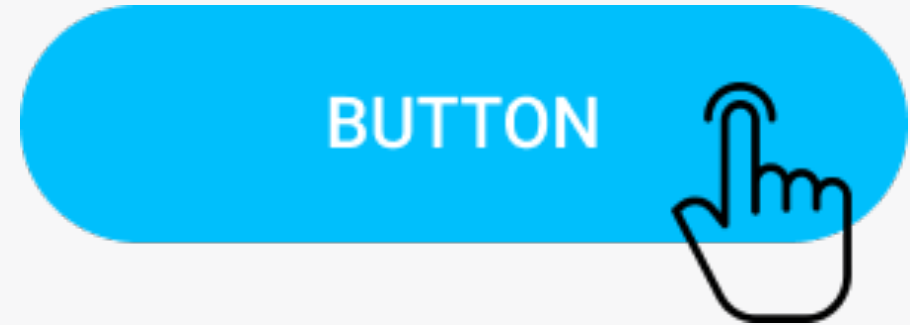
# Стилизация курсора

**cursor pointer** - устанавливает форму курсора, когда он находится в пределах элемента.

Вид курсора зависит от операционной системы и установленных параметров.

Значения:

`cursor: default / pointer;`



cursor

# Работа со шрифтами

**GoogleFonts** - после выбора шрифта нужно определиться с необходимыми **типами** шрифтов. От этого зависит размер файлов шрифта и скорость загрузки страницы.

Значение – название:

100	- Thin (Hairline)	- Тонкий
200	- Extra Light (Ultra Light)	- Дополнительный светлый
300	- Light	- Светлый
400	- Normal	- Нормальный
500	- Medium	- Средний
600	- Semi Bold (Demi Bold)	- Полужирный
700	- Bold	- Жирный
800	- Extra Bold (Ultra Bold)	- Дополнительный жирный
900	- Black (Heavy)	- Черный



# Варианты подключения шрифта

В тег <head>

```
<link rel="preconnect" href="https://fonts.googleapis.com">  
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>  
<link href="https://fonts.googleapis.com/css2?family=Roboto:wght@100;400;700;&display=swap"  
rel="stylesheet">
```

# Варианты подключения шрифта

Подключение в общий файл стилей. В начало файла css нужно добавить код:

```
@import url(  
  'https://fonts.googleapis.com/css2?family=Roboto:wght@100;300;400;500;700;900&display=swap'  
);
```





# Варианты подключения шрифта

GoogleFonts - предлагает нам использовать шрифт так:

```
.title {  
  font-family: "Roboto", sans-serif;  
  font-size: 36px;  
  font-weight: 700;  
}
```

# Локальное подключение шрифтов

Правило @font-face

```
@font-face {  
  font-family: "MyWebFont";  
  src: url("../fonts/WebFont.eot");  
  src: url("../fonts/WebFont.eot?iefix") format("eot"),  
        url("../fonts/WebFont.woff") format("woff"),  
        url("../fonts/WebFont.ttf") format("truetype"),  
        url("../fonts/WebFont.svg#webfont") format("svg");  
  font-weight: normal;  
  font-style: normal;  
}
```

fonts



# Работа со шрифтами

Используются следующие форматы:

TTF/OTF – работают в большинстве браузеров, кроме IE.

EOT – представляет сжатую копию шрифта TTF, только в IE.

WOFF – формат представляет собой сжатый шрифт в формате TTF/OTF.

WOFF2 – имеет улучшенное сжатие, по сравнению с первой версией.



# Работа со шрифтами

Рекомендуется подключать файлы шрифтов по приоритету:

WOFF2 – современные браузеры.

WOFF – браузеры, которые не поддерживают WOFF2.

TTF – устаревшие браузеры.

EOT – поддержка IE.



# Варианты подключения шрифта

Внимание! Шрифт должен быть подключен до его использования в коде.



# Методология БЭМ

**БЭМ-методология** — это набор правил и рекомендаций по организации работы над проектом.

- Разработчик должен понимать свой код и код любого программиста в команде БЭМ-проекта.
- Любой блок кода может быть использован повторно.
- Работая в одной команде, разработчики, менеджеры, дизайнеры и верстальщики должны называть одни и те же вещи одинаково.

Аббревиатура от трех слов Блок, Элемент и Модификатор стала названием методологии — БЭМ.

## Блок

Логически и функционально независимый компонент страницы. Блок полностью самодостаточен: у него может быть свое поведение, шаблоны, стили, документация и не только. Блоки могут использоваться в любом месте страницы. Одни блоки можно вкладывать в другие.

## Элемент

Часть блока, которая не может использоваться в отрыве от него и имеет смысл только в рамках своего родителя. Работая с элементами, важно помнить правило: не рекомендуется создавать элементы элементов.

## Модификатор

Свойство блока или элемента, которое меняет их внешний вид, состояние или поведение. Модификатор имеет имя и может иметь значение. У блока/элемента может быть несколько разных модификаторов одновременно.

[Ссылка на полную документацию](#)



# Методология БЭМ

```
<header class="header">
  
  <form action="#" class="header__search-form">
    <input type="text" class="header__search-form__input">
    <button class="header__search-form__button">send</button>
  </form>
  <div class="header__lang-switcher">content</div>
</header>
```

# Ссылки на дополнительные материалы

1. [WebReference](#). Руководства по веб-технологиям.
2. [HTML5BOOK.RU](#)
3. [Can I Use](#). Справочник кроссбраузерной поддержки web технологий.
4. [MDN](#). Справочник разработчика.

