

Путь к карьере Frontend Fullstack разработчика

Модуль 1. WEB CORE

Уровень 10. Фон и декоративные элементы
Часть 1



Свойство `background-color`

Свойство `background-color` позволяет задать цвет фона любого элемента на странице, будь то блок, строка или строчно-блочный элемент.

- Применяется к любым типам элементов.
- Фон заливается под весь текст, даже если он занимает несколько строк.
- Можно использовать полупрозрачные цвета или ключевое слово `transparent`.
- Значение по умолчанию: `transparent` (прозрачный).
- Свойство фона не наследуется.
- Изменение цвета фона можно анимировать с помощью свойства `transition`.

По умолчанию кнопки в браузерах имеют не самый привлекательный серый фон.

Чтобы создать красивую кнопку, нужно "обнулить" этот фон, установив `background-color: transparent`

Затем вы можете задать любой другой цвет фона, который соответствует вашему дизайну.

```
.element {
  background-color: black; /* Чёрный фон */
}
.element {
  background-color: #ffffff; /* Белый фон */
}
.element {
  background-color: transparent; /* Прозрачный фон */
}
```

Свойство `background-image`

Свойство `background-image` позволяет вам задать фоновое изображение для любого элемента на странице.

```
div {  
  background-image: url("background.png");  
}
```

Элементу желательно задать одновременно и цвет фона `background-color`, и фоновую картинку. Если картинка не загрузится, то вместо неё пользователь увидит фоновый цвет.

```
.background-url {  
  body {  
    background-image: url("background.png");  
    background-color: #09ff00;  
  }  
}
```

Стоит понимать разницу между `<div>` с фоновой картинкой и, собственно, картинкой ``. Картинке можно задать только один параметр размера: высоту или ширину, и браузер сам рассчитает второй. У блока с фоном, внутри которого нет контента, надо указывать оба параметра

Свойство `background-position`

Свойство `background-position` позволяет вам точно контролировать, где именно будет располагаться фоновое изображение внутри элемента. По умолчанию картинка "прилипает" к левому верхнему углу, но вы можете легко изменить это поведение.

```
background-position: горизонтальная-позиция вертикальная-позиция;
```

Как это работает:

- Задаётся двумя значениями: первое - горизонтальное положение, второе - вертикальное.
- Можно использовать ключевые слова (`top`, `bottom`, `left`, `right`, `center`) или проценты/единицы измерения.

```
background-image: url("cat.png");  
background-position: bottom right; /* Картинка в правом нижнем углу */  
}
```

Значения в процентах указывают на позицию изображения относительно размеров элемента.

Значения в пикселях или других абсолютных единицах позволяют точно позиционировать изображение.

Свойство `background-repeat`

Свойство `background-repeat` определяет, как фоновое изображение будет повторяться внутри элемента. Это позволяет создавать различные эффекты, от заполнения всего элемента повторяющимися плитками до отображения изображения только один раз.

- `repeat`: Изображение повторяется и по горизонтали, и по вертикали (значение по умолчанию).
- `repeat-x`: Изображение повторяется только по горизонтали.
- `repeat-y`: Изображение повторяется только по вертикали.
- `no-repeat`: Изображение не повторяется и отображается один раз.

```
.repeat {  
  height: 100vh;  
  background-image: linear-gradient(  
    #49a16c 50px,  
    #064236 0  
  );  
  background-size: 100% 100px;  
  background-repeat: repeat-y;  
}  
<div class="repeat"></div>
```

С помощью повторения фона и линейного градиента (`linear-gradient`) можно создавать полосатые фоны.

Свойство `background-size`

Свойство `background-size` позволяет вам контролировать размер фонового изображения внутри элемента.

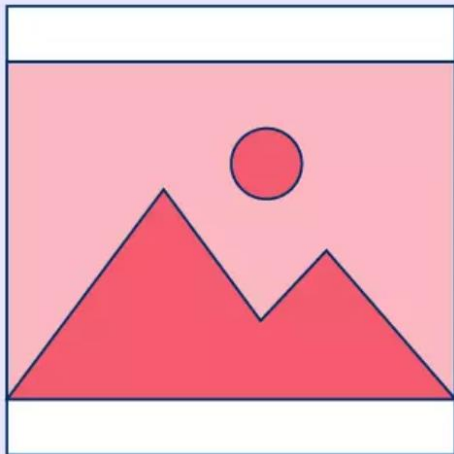
```
.size-contain {  
  background-image: url('https://via.placeholder.com/150');  
  background-size: contain; /* Масштабирование до полного вписывания */  
}  
.size-pixels {  
  background-image: url('https://via.placeholder.com/150');  
  background-size: 100px 50px; /* Размер в пикселях */  
}
```

- **auto**: Размер изображения определяется автоматически (по умолчанию).
- **cover**: Изображение масштабируется, чтобы полностью покрыть элемент, сохраняя пропорции. Части изображения могут быть обрезаны.
- **contain**: Изображение масштабируется, чтобы полностью вписаться в элемент, сохраняя пропорции. Могут остаться пустые области.
- **Абсолютные значения**: Задают ширину и высоту изображения в пикселях или других единицах измерения (например, 100px 50px).
- **Процентные значения**: Задают ширину и высоту изображения в процентах от размера элемента (например, 50% 50%).

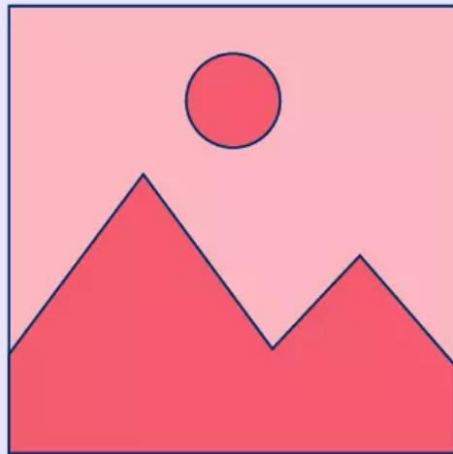
Contain or cover

Значение **cover** используется чаще всего для создания красивых фонов.

background-size: contain



background-size: cover



Многоуровневый фон в CSS

CSS позволяет использовать несколько фоновых изображений в одном элементе, что открывает широкий спектр возможностей для создания сложных и интересных визуальных эффектов

Для задания нескольких фоновых изображений используется синтаксис, в котором различные изображения перечисляются через запятую:

```
element {  
    background-image: imageUrl1, imageUrl2, imageUrl3;  
}
```

Для каждого слоя фоновой изображения можно задать свои настройки, используя свойства с множественными значениями (через запятую):

- **background-position**: Положение каждого изображения.
- **background-size**: Размер каждого изображения.
- **background-repeat**: Повторение каждого изображения.
- **background-origin**: Откуда начинать отсчёт позиции фона (от рамки, от внутреннего отступа или от содержимого).
- **background-clip**: Область, в которой будет виден фон (до рамки, до внутреннего отступа или только в области содержимого).
- **background-attachment**: Фиксированный или прокручиваемый фон.

Наложение изображений

В этом примере мы создали фон с повторяющимся паттерном и градиентом поверх него, растянутым на всю область элемента.

```
.element {  
    /* Два фоновых изображения */  
    background-image: url('pattern.png'), url('gradient.jpg');  
    background-repeat: repeat, no-repeat; /* Первое повторяется, второе - нет */  
    background-size: 50px, cover; /* Первое изображение 50px, второе растягивается  
на весь элемент */  
}
```

linear-gradient()

Линейный градиент описывает изменение цвета вдоль прямой линии. В отличие от фоновых изображений градиент не может иметь конкретных размеров и его фактический размер совпадает с размером элемента.

```
.element {  
  background-image: linear-gradient(#6e4aff, #49A16C);  
}
```

Мы можем задавать направление градиента, используя ключевые слова с приставкой to: to left, to top, to right, to bottom (по умолчанию).

Значения имеют следующие эквиваленты в углах:

- to top — 0deg;
- to bottom — 180deg;
- to right — 90deg;
- to left — 270deg;

radial-gradient()

Функция **radial-gradient()** используется для задания фона в виде радиального (кругового, эллиптического) градиента.

```
.radial-gradient-at-corner {  
  width: 200px;  
  height: 200px;  
  background: radial-gradient(circle at top left, red, blue, green);  
}  
  
<body>  
<div class="radial-gradient-at-corner">Градиент из угла</div>  
</body>
```

Мы можем задавать положение центральной точки, используя те же ключевые слова, что и для background-position, но с приставкой **at**: **at left**, **at top**, **at right**, **at bottom**, **at center** (по умолчанию). Значения можно сочетать, чтобы поместить центр градиента в нужный угол элемента: **at top left** — центр градиента в верхнем левом углу.

Размер: можно указать **closest-side**, **closest-corner**, **farthest-side** или **farthest-corner**.

box-shadow

Свойство **box-shadow** позволяет создавать реалистичные тени для элементов на странице, добавляя им глубину и визуальную привлекательность. С помощью тени можно создать блок с несколькими рамками!

```
element {  
  box-shadow: смещение-по-горизонтали смещение-по-вертикали радиус-размытия радиус-расширения цвет;  
}
```

```
.shadow-basic {  
  box-shadow: 10px 10px 5px rgba(0, 0, 0, 0.5); /* Основная тень */  
}  
  
.shadow-blur {  
  box-shadow: 10px 10px 20px rgba(0, 0, 0, 0.5); /* Тень с размытием */  
}
```

- **смещение-по-горизонтали:** Положительное значение смещает тень вправо, отрицательное - влево.
- **смещение-по-вертикали:** Положительное значение смещает тень вниз, отрицательное - вверх.

Свойство `border`

Рамки (**border**) - это линии, которые окружают элемент, выделяя его на странице и улучшая визуальную структуру. CSS позволяет гибко настраивать рамки, задавая их толщину, стиль и цвет.

border состоит из трёх других свойств (в скобках значения по умолчанию):

- **border-width** (medium);
- **border-style** (none);
- **border-color** (currentcolor).

```
.border-full {  
  border-width: 3px;  
  border-style: solid;  
  border-color: magenta;  
}
```

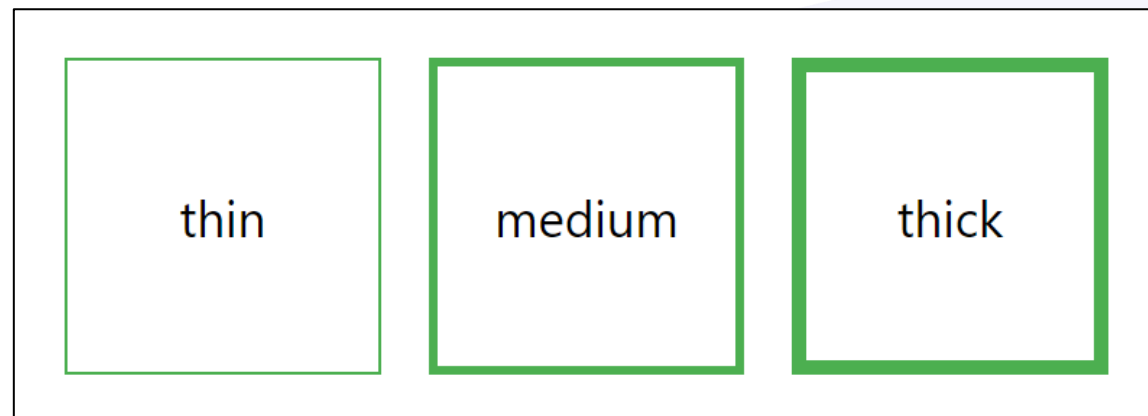
Lorem, ipsum dolor
sit amet consectetur
adipiscing elit.

border-width – толщина границы

Свойство **border-width** определяет толщину границы. По умолчанию данное свойство имеет значение **medium**, что эквивалентно 3px.

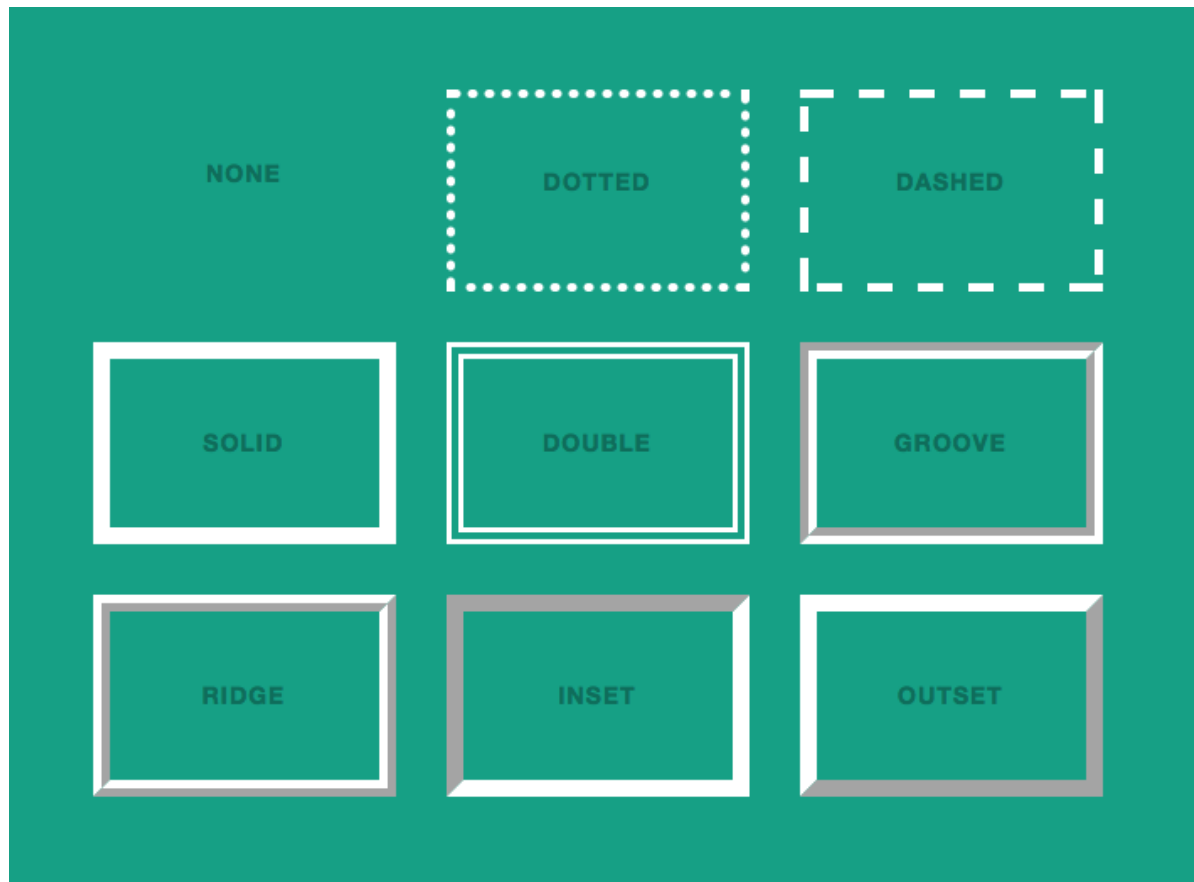
Задавать значение можно с помощью ключевых слов: **thin** (эквивалентно 1px), **medium** (3px) и **thick** (5px), а также используя различные единицы измерения: px, em, rem, vh, vw.

```
.thin {  
  border-width: thin;  
}  
.medium {  
  border-width: medium;  
}  
.thick {  
  border-width: thick;  
}
```



border-style – тип линии

Свойство **border-style** предназначено для установки стилей границам.



border-color: раскрашиваем границы

Свойство border-color позволяет вам задать цвет для границ элемента. По умолчанию границы наследуют цвет текста элемента, но вы можете легко изменить это, используя различные способы задания цвета.

Способы задания цвета:

- Названия цветов: **red**, **green**, **blue** и т.д.
- Шестнадцатеричные коды: **#ff0000**, **#00ff00** и т.д.
- Функции: **rgb()**, **rgba()**, **hsl()**, **hsla()**.
- Ключевое слово: **transparent** (прозрачный).

```
.element {  
    border-color: red; /* Все границы будут красными */  
}  
  
.element {  
    border-color: red green blue yellow; /* Верхняя - красная, правая  
- зелёная, нижняя - синяя, левая - жёлтая */  
}
```


border-radius: сглаживаем углы

Свойство **border-radius** позволяет нам делать углы элементов плавными и округлыми, даже если у них нет видимой рамки. Оно действует и на фон элемента, что открывает возможности для интересных дизайнерских решений.

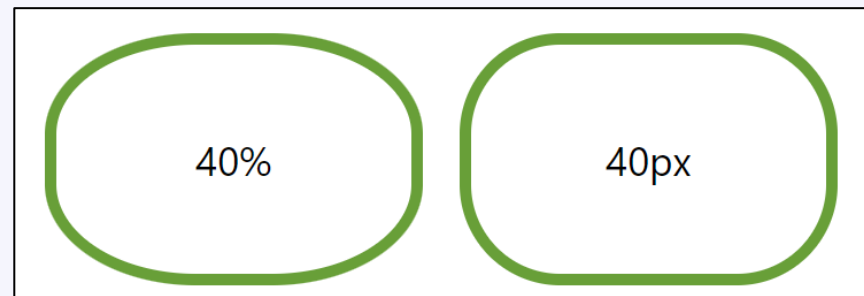
Чаще всего **border-radius** используется с одним значением, например:

```
.radius-all {  
  border-radius: 15px; /* Все углы скруглены одинаково */  
}
```

Вы можете задавать радиус в пикселях (**px**) или процентах (**%**). Проценты рассчитываются относительно ширины и высоты элемента.

Пример, показывающий разницу между **border-radius: 40px** и **border-radius: 40%** применительно к прямоугольнику.

Чтобы сделать элемент круглым, установите **border-radius: 50%**



Some day I will be this good



Домашнее задание

Уровень 10. Фон и декоративные элементы
Лекции 0-9

