

# CSS - background

background, градиент,  
работа с изображением и мелкие стили

# background

- background-color
- background-image
- background-repeat
- background-attachment
- background-position
- background-clip
- background-origin
- background-size
- background

# background-color

Свойство `background-color` устанавливает цвет фона элемента. Цвет рисуется за фоновыми изображениями. Для блочных элементов цвет фона распространяется на всю ширину и высоту блока элемента, для строчных — только на область их содержимого.

## Значения:

**цвет**                      Значение принимает все форматы цвета свойства `color`. Значение по умолчанию `transparent`.

`initial`                      Устанавливает значение свойства в значение по умолчанию.

`inherit`                      Наследует значение свойства от родительского элемента.

```
background-color: salmon;  
background-color: #00ff00;  
background-color: rgba(255, 128, 128, 0.5);  
background-color: currentColor;  
background-color: transparent;  
background-color: inherit;  
background-color: initial;
```

# background-image

Свойство `background-image` устанавливает фоновое изображение (одно или несколько) элемента. Значение `none` считается слоем изображения, но ничего не рисует. Изображение, которое является пустым (нулевой ширины или нулевой высоты), которое не загружается или не может быть отображено (например, потому что оно не в поддерживаемом формате изображения) также считается слоем, но ничего не рисует.

Семантически важные изображения должны предоставляться в разметке документа, например, с тегом `<img>`.

## Значения:

изображение	Обозначает 2D-изображение. Это может быть ссылка на URL, нотация <code>image()</code> или запись градиента. Значение по умолчанию <code>none</code> .
<code>inherit</code>	Наследует значение свойства от родительского элемента.

## Синтаксис

```
background-image: none;  
background-image: url(http://site.ru/rose.png);  
background-image: url(tl.png), url(tr.png);  
background-image: linear-gradient(white, gray);  
background-image: repeating-radial-gradient(circle closest-side at 20px 30px, red 1px, transparent 0);  
background-image: image("sprites.svg#xywh=40,0,20,20");  
background-image: inherit;
```

CSS

# background-repeat

Свойство `background-repeat` определяет, как фоновые изображения укладываются в области фона после того, как для них установлены размеры и позиционирование. Если значение свойства имеет два ключевых слова, первое используется для горизонтального направления, второе — для вертикального.



Значения:

`repeat-x`

Изображение повторяется в горизонтальном направлении. Вычисляется в `repeat no-repeat`.

`repeat-y`

Изображение повторяется в вертикальном направлении. Вычисляется в `no-repeat repeat`.

`repeat`

Изображение повторяется в обоих направлениях так часто, чтобы покрыть область отрисовки фона. Если изображение не помещается, оно обрезается. Вычисляется в `repeat repeat`. Значение по умолчанию.

`space`

Изображение повторяется столько раз, сколько оно помещается в области фона, не обрезаясь, изображения расположены на равном расстоянии друг от друга. Первое и последнее изображения касаются краев области. Если область рисования фона больше, чем область позиционирования фона, шаблон повторяется, чтобы заполнить область рисования фона. Если недостаточно места для двух копий изображения, то размещается только одно изображение, а свойство `background-position` определяет его положение. Вычисляется в `space space`.

`round`

Изображение повторяется так часто, чтобы заполнить область фона, масштабируясь и не обрезаясь. Вычисляется в `round round`.

`no-repeat`

Изображение размещается один раз и не повторяется. Вычисляется в `no-repeat no-repeat`.

`initial`

Устанавливает значение свойства в значение по умолчанию.

`inherit`

Наследует значение свойства от родительского элемента.

# background-attachment

Свойство background-attachment указывает, является ли фоновое изображение фиксированными относительно области просмотра или прокручивается вместе с элементом или его содержимым.

Fixed - создает эффект параллакса (!важно)

Пример - [ТЫК](#)

## Значения:

scroll

Фоновое изображение прокручивается вместе с текстом и другим содержимым. Значение по умолчанию.

fixed

Предотвращает перемещение, фиксирует фоновое изображение на заднем плане.

local

Фоновое изображение прокручивается вместе с содержимым элемента.

initial

Устанавливает значение свойства в значение по умолчанию.

inherit

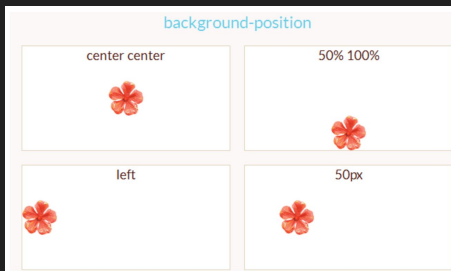
Наследует значение свойства от родительского элемента.

```
background-attachment: scroll;  
background-attachment: fixed;  
background-attachment: local;  
background-attachment: inherit;  
background-attachment: initial;
```

# background-position

Если для элемента заданы фоновые изображения, свойство `background-position` указывает их начальное положение (после любого изменения размера) в соответствующей области расположения фона.

```
background-position: top;  
background-position: bottom;  
background-position: left;  
background-position: right;  
background-position: center;  
background-position: 25% 75%;  
background-position: 0 0;  
background-position: 1cm 2cm;  
background-position: 10ch 8em;  
background-position: 0 0, center;  
background-position: bottom 10px right 20px;  
background-position: right 3em bottom 10px;  
background-position: bottom 10px right;  
background-position: top right 10px;  
background-position: inherit;  
background-position: initial;
```



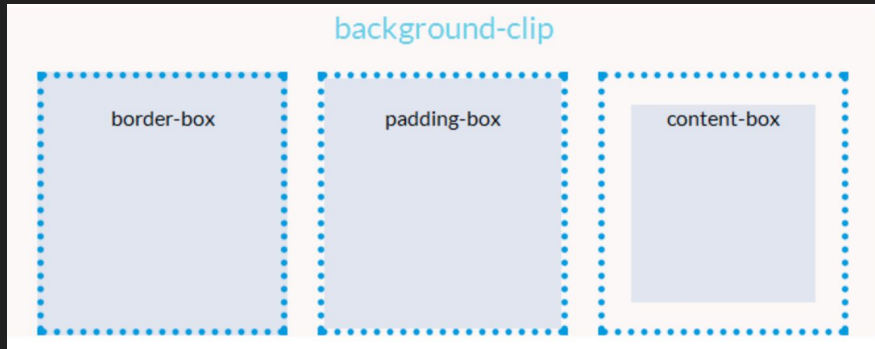
Значения:

<code>%</code>	Горизонтальное смещение вычисляется по формуле <code>ширина области расположения фона — ширина фонового изображения</code> . Вертикальное смещение по формуле <code>высота области расположения фона — высота фонового изображения</code> , где размер изображения — это размер, заданный свойством <code>background-size</code> . Значение по умолчанию <code>0% 0%</code> .
<code>длина</code>	Значение длины дает фиксированную длину в качестве смещения.
<code>left</code>	Вычисляет до <code>0%</code> для горизонтальной позиции, если задано одно или два значения, в противном случае смещение происходит относительно левого края.
<code>center</code>	Вычисляет в <code>left 50%</code> для горизонтального положения, если не указано иное горизонтальное положение, или как <code>top 50%</code> для вертикального положения, если оно задано.
<code>right</code>	Вычисляет в <code>100%</code> для горизонтального положения, если задано одно или два значения, в противном случае смещение происходит относительно правого края.
<code>top</code>	Вычисляет в <code>0%</code> для вертикальной позиции, если задано одно или два значения, в противном случае смещение происходит относительно верхнего края.
<code>bottom</code>	Вычисляет в <code>100%</code> для вертикальной позиции, если задано одно или два значения, в противном случае смещение происходит относительно нижнего края.
<code>initial</code>	Устанавливает значение свойства в значение по умолчанию.
<code>inherit</code>	Наследует значение свойства от родительского элемента.

# background-clip

Свойство `background-clip` определяет область рисования фона. Фон всегда рисуется под рамкой элемента, если таковая имеется.

Корневой элемент имеет другую область рисования фона, поэтому свойство `background-clip` на него не влияет.



## Значения:

<code>border-box</code>	Фон закрашивает область в пределах рамки элемента. Значение по умолчанию.
<code>padding-box</code>	Фон закрашивает область в пределах внутренних полей элемента.
<code>content-box</code>	Фон закрашивает только область содержимого.
<code>initial</code>	Устанавливает значение свойства в значение по умолчанию.
<code>inherit</code>	Наследует значение свойства от родительского элемента.

```
background-clip: border-box;  
background-clip: padding-box;  
background-clip: content-box;  
background-clip: text;  
background-clip: inherit;  
background-clip: initial;
```



# background-origin

Свойство `background-origin` указывает область расположения фона для элементов, которые выводятся на экране как единый блок (например, не абзацы текста).

Если для элемента установлено `background-attachment: fixed`, свойство не будет иметь эффекта.

Если для элемента заданы `background-clip: padding-box`, `background-origin: border-box`, `background-position: top left`, и элемент имеет ненулевую рамку, тогда верхняя и левая части фонового изображения будут обрезаны.

## Значения:

<code>padding-box</code>	Фон позиционируется относительно верхних границ области внутренних полей элемента. Значение по умолчанию.
<code>border-box</code>	Фон позиционируется относительно верхних границ рамки элемента.
<code>content-box</code>	Фон позиционируется относительно верхних границ области содержимого элемента.
<code>initial</code>	Устанавливает значение свойства в значение по умолчанию.
<code>inherit</code>	Наследует значение свойства от родительского элемента.

```
background-origin: border-box;  
background-origin: padding-box;  
background-origin: content-box;  
background-origin: inherit;  
background-origin: initial;
```

# background-size

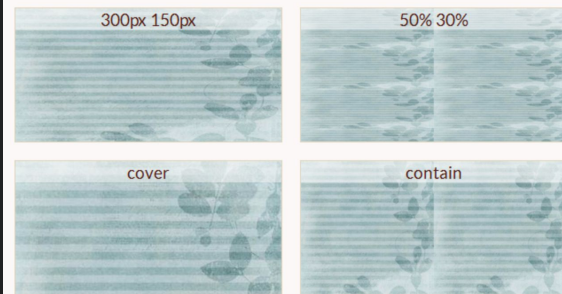
Свойство `background-size` устанавливает размер фоновых изображений.

## Значения:

<code>auto</code>	Значение по умолчанию. Высота и ширина изображения равны его оригинальным размерам.
<code>длина</code>	Размер задается парой значений, первое значение устанавливает ширину изображения, второе — высоту. Для того, чтобы фон масштабировался вместе с текстом, размеры изображения нужно задавать в <code>em</code> .
<code>%</code>	Задаёт размер фонового изображения в процентах от ширины или высоты элемента, которое заполняется фоном.
<code>cover</code>	Масштабирует изображение с сохранением пропорций так, чтобы его ширина или высота равнялась ширине или высоте блока.
<code>contain</code>	Масштабирует изображение с сохранением пропорций таким образом, чтобы оно целиком поместилось внутри блока.
<code>initial</code>	Устанавливает значение свойства в значение по умолчанию.
<code>inherit</code>	Наследует значение свойства от родительского элемента.

```
background-size: cover;  
background-size: contain;  
background-size: 50%;  
background-size: 3.2em;  
background-size: 12px;  
background-size: auto;  
background-size: 50% auto;  
background-size: 3em 25%;  
background-size: auto 6px;  
background-size: auto auto;  
background-size: auto, auto;  
background-size: 50%, 25%, 25%;  
background-size: 6px, auto, contain;  
background-size: inherit;  
background-size: initial;
```

## background-size



# background

Свойство `background` позволяет описать в одном объявлении следующие свойства фона: `background-color`, `background-image`, `background-position`, `background-size`, `background-repeat`, `background-origin`, `background-clip` и `background-attachment`. Необязательно указывать все перечисленные свойства, если какое-либо свойство будет пропущено, оно примет значение по-умолчанию.

# Множественные фоны

Фон блока элемента может иметь несколько слоев в CSS3. Количество слоев определяется количеством значений, разделенных запятыми, указанных в свойстве `background-image`. Значение `none` по-прежнему создает слой.

Первое изображение в списке — это слой, отображаемый ближайший к пользователю, следующий отрисовывается за первым, и так далее. Цвет фона, если он есть, закрашивается под всеми остальными слоями.

```
div {  
  width: 680px;  
  height: 630px;  
  background-image: url(https://html5book.ru/wp-content/uploads/2015/02/flower\_rose.png),  
    url(https://html5book.ru/wp-content/uploads/2015/02/love.png),  
    url(https://html5book.ru/wp-content/uploads/2015/02/border\_white.png);  
  background-repeat: no-repeat;  
  background-position: bottom right, center center, top left;  
}
```



# Градиент

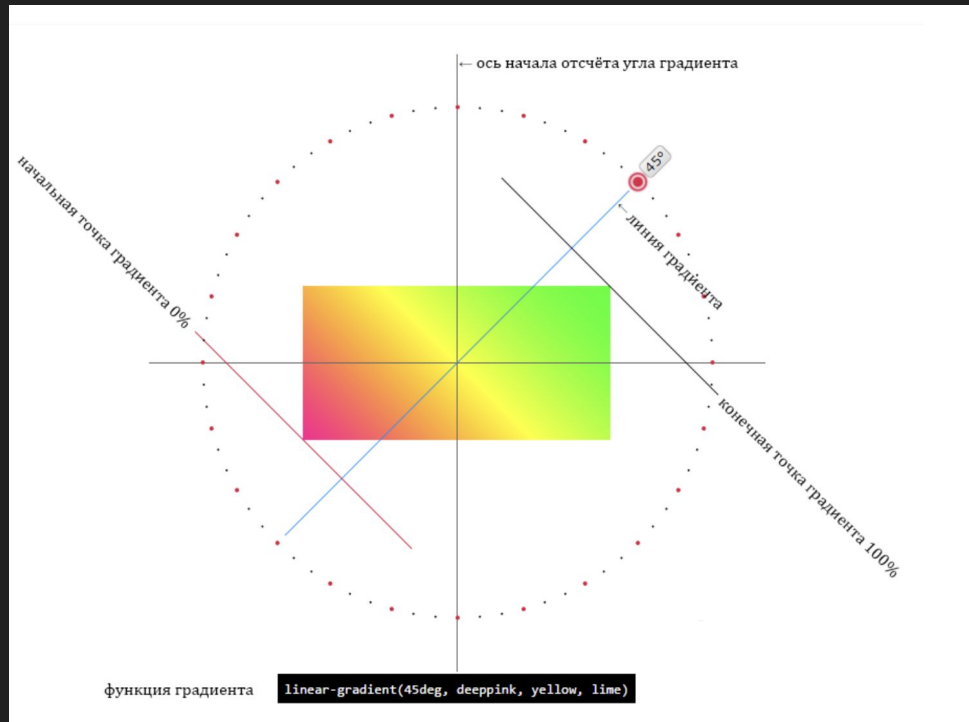
CSS-градиент представляет собой переходы от одного цвета к другому.

Градиенты создаются с помощью функций `linear-gradient()` и `radial-gradient()`.

Градиентный фон можно устанавливать в свойствах `background`, `background-image`, `border-image` и `list-style-image`.

- Линейный градиент `linear-gradient()`
- Радиальный градиент `radial-gradient()`
- Повтор градиента
- Кроссбраузерный градиент
- Комбинация градиента и фонового изображения

# Линейный градиент linear-gradient()



Линейный градиент создается с помощью двух и более цветов, для которых задано направление, или линия градиента.

`background: linear-gradient(угол / сторона или угол наклона с помощью ключевого слова (пары ключевых слов), первый цвет, второй цвет и т.д.);`

# Линейный градиент linear-gradient()

Направление градиента может быть задано двумя способами:

- с помощью угла наклона в градусах deg, значение которого определяет угол наклона линии внутри элемента.
- с помощью ключевых слов to top, to right, to bottom, to left, которые соответствуют углу градиента, равному 0deg, 90deg, 180deg и 270deg соответственно.
- Если направление задано парой ключевых слов, например, to top left, то начальная точка градиента будет расположена в противоположном направлении, в данном случае справа внизу.

```
div {  
  height: 200px;  
  background: linear-gradient(to top left, powderblue, pink);  
}
```

```
div {  
  height: 200px;  
  background: linear-gradient(to right, #F6EFD2, #CEAD78);  
}
```

```
div {  
  height: 200px;  
  background: linear-gradient(45deg, #EECFBA, #C5DDE8);  
}
```

Для неравномерного распределения цветов указывается начальная позиция каждого цвета через точки остановки градиента, так называемые color stops. Точки остановки задаются в %, где 0% — начальная точка, 100% — конечная точка, например:

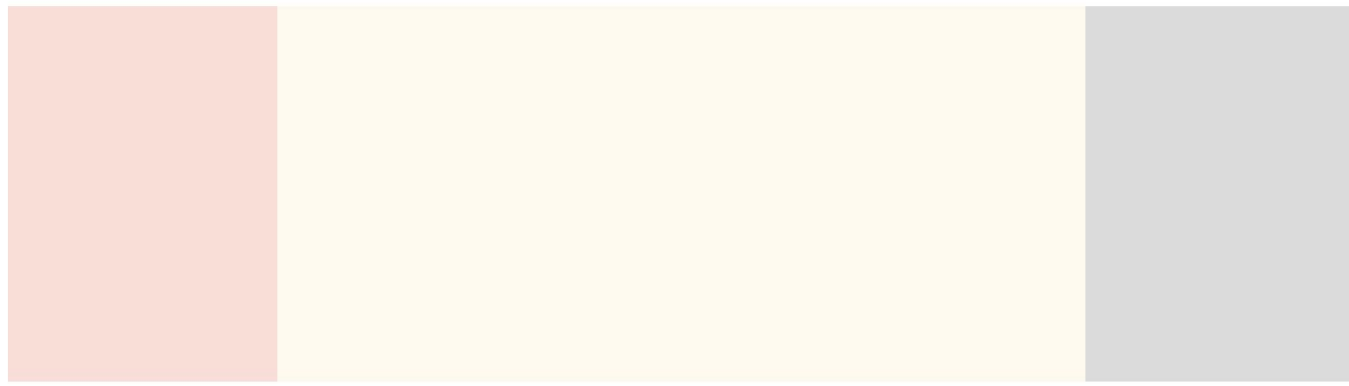
```
div {  
  height: 200px;  
  background: linear-gradient(to top, #E4AF9D 20%, #E4E4D8 50%, #A19887 80%);  
}
```





Для чёткого распределения цветных полос каждый последующий цвет нужно начинать с точки остановки предыдущего цвета:

```
div {  
  height: 200px;  
  background: linear-gradient(to right, #FFDDD6 20%, #FFF9ED 20%, #FFF9ED 80%, #DBDBDB 80%);  
}
```



# Радиальный градиент radial-gradient()

Радиальный градиент отличается от линейного тем, что цвета выходят из одной точки (центра градиента) и равномерно распределяются наружу, рисуя форму круга или эллипса.

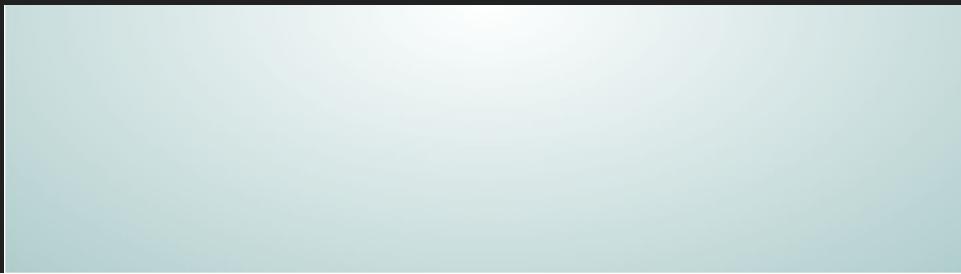
background: radial-gradient(форма градиента / размер / позиция центра, первый цвет, второй цвет и т.д.);

Форма градиента определяется ключевыми словами `circle` или `ellipse`. Если форма не задана, по умолчанию радиальный градиент принимает форму эллипса.

```
div {  
  height: 200px;  
  background: radial-gradient(white, #FFA9A1);  
}
```



Позиция центра задаётся с помощью ключевых слов, используемых в свойстве `background-position`, с добавлением приставки `at`. Если позиция центра не задана, используется значение по умолчанию `at center`.

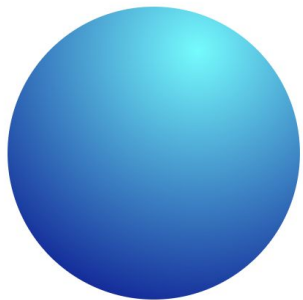


```
div {  
  height: 200px;  
  background: radial-gradient(at top, #FFFFFF, #A7CECC);  
}
```

Размер градиента задаётся с помощью ключевых слов. Значение по умолчанию `farthest-corner` (к дальнему углу).

Значение	Описание
<code>closest-side</code>	Размер градиента рассчитывается из расстояния до любой ближней стороны блока для <code>circle</code> или до ближних сторон по <code>X</code> и по <code>Y</code> для <code>ellipse</code> .
<code>farthest-side</code>	Размер рассчитывается из расстояния до дальних сторон.
<code>closest-corner</code>	Размер рассчитывается из расстояния до ближних углов.
<code>farthest-corner</code>	Размер рассчитывается из расстояния до дальних углов.

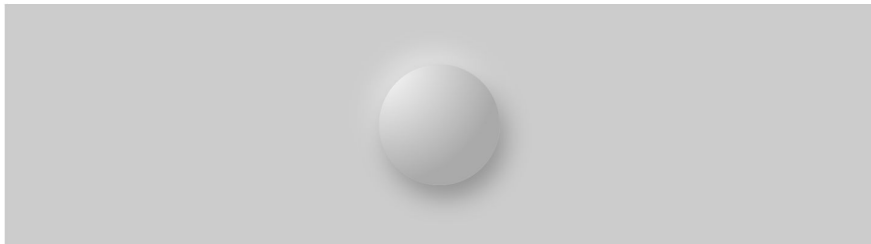
Мяч



```
div {  
  width: 200px;  
  height: 200px;  
  border-radius: 50%;  
  margin: 0 auto;  
  background: radial-gradient(circle at 65% 15%, aqua, darkblue);  
}
```

CSS

## Кнопка



HTML

```
<div class="wrap"><div class="button"></div></div>
```

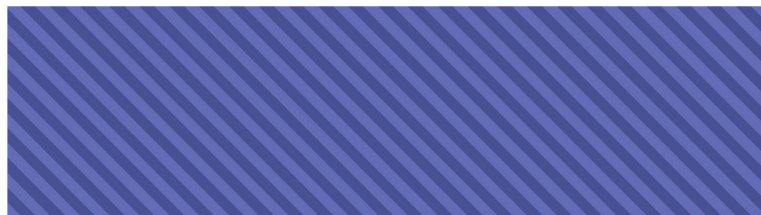
CSS

```
.wrap {  
  height: 200px;  
  padding: 50px 0;  
  background: #cccccc;  
}  
.button {  
  width: 100px;  
  height: 100px;  
  border-radius: 50%;  
  margin: 0 auto;  
  background: radial-gradient(farthest-side ellipse at top left, white, #aaaaaa);  
  box-shadow: 5px 10px 20px rgba(0,0,0,0.3), -5px -10px 20px rgba(255,255,255,0.5);  
}
```

# Повтор градиента

В дополнение к линейному и радиальному градиентам существует повтор градиента, который задается с помощью функций `repeating-linear-gradient()` и `repeating-radial-gradient()` соответственно. Фон из повторяющихся градиентов выглядит неаккуратно, поэтому рекомендуется начинать следующий цвет с точки остановки предыдущего, создавая таким образом полосатые узоры.

```
div {  
  height: 200px;  
  background: repeating-linear-gradient(45deg, #606dbc, #606dbc 10px, #465298 10px,  
}
```



```
div {  
  height: 200px;  
  background: repeating-radial-gradient(circle, #B9ECFE, #B9ECFE 10px, #82DDFF 10px,  
}
```





# Кроссбраузерный градиент

Для корректного отображения градиентов во всех браузерах необходимо добавить кроссбраузерную запись.

Вендорные префиксы

## Линейный градиент

```
CSS
-ms-filter: "progid:DXImageTransform.Microsoft.gradient (GradientType=0, startColor=red, endColor=red)";
background: -webkit-linear-gradient(left, red, #f06d06); /* Safari 5.1, iOS 5.0-6.0 */
background: -moz-linear-gradient(left, red, #f06d06); /* Firefox 3.6-15 */
background: -o-linear-gradient(left, red, #f06d06); /* Opera 11.1-12 */
background: linear-gradient(to right, red, #f06d06); /* Opera 15+, Chrome 25+, IE 9+
```

## Повтор линейного градиента

```
CSS
background: -webkit-repeating-linear-gradient(red, yellow 10%, green 20%); /* Safari 5.1-6.0 */
background: -o-repeating-linear-gradient(red, yellow 10%, green 20%); /* Opera 11.6-12.0 */
background: -moz-repeating-linear-gradient(red, yellow 10%, green 20%); /* Firefox 3.6-15 */
background: repeating-linear-gradient(red, yellow 10%, green 20%); /* Стандартный синтаксис
```

## Радиальный градиент

```
CSS
background: -webkit-radial-gradient(red, yellow, green); /* Safari 5.1-6.0 */
background: -o-radial-gradient(red, yellow, green); /* Opera 11.6-12.0 */
background: -moz-radial-gradient(red, yellow, green); /* Firefox 3.6-15 */
background: radial-gradient(red, yellow, green); /* Стандартный синтаксис */
background: -webkit-radial-gradient(60% 55%, farthest-side, red, yellow, black); /* Safari 5.1-6.0 */
background: -o-radial-gradient(60% 55%, farthest-side, red, yellow, black); /* Opera 11.6-12.0 */
background: -moz-radial-gradient(60% 55%, farthest-side, red, yellow, black); /* Firefox 3.6-15 */
background: radial-gradient(60% 55%, farthest-side, red, yellow, black); /* Стандартный синтаксис
```

## Повтор радиального градиента

```
CSS
background: -webkit-repeating-radial-gradient(red, yellow 10%, green 15%); /* Safari 5.1-6.0 */
background: -o-repeating-radial-gradient(red, yellow 10%, green 15%); /* Opera 11.6-12.0 */
background: -moz-repeating-radial-gradient(red, yellow 10%, green 15%); /* Firefox 3.6-15 */
background: repeating-radial-gradient(red, yellow 10%, green 15%); /* Стандартный синтаксис
```

# Комбинация градиента и фонового изображения

За счет комбинации градиента и изображения можно создавать интересные эффекты. Для градиента нужно использовать полупрозрачные цвета, чтобы картинка оставалась видимой.



```
div {  
  height: 453px;  
  background: linear-gradient(45deg, rgba(103, 0, 31, .8), rgba(34, 101, 163, .5)), url(https://html5book.ru/wp-content/uploads/2016/12/photo-8.jpg);  
  background-size: cover;  
}
```

# Изображение

## Свойство object-fit



Оригинал изображения – 300\*200px



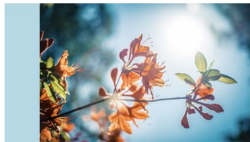
width: 300px;  
height: 150px;  
object-fit: fill;



width: 300px;  
height: 150px;  
object-fit: contain;



width: 300px;  
height: 150px;  
object-fit: cover;



width: 300px;  
height: 150px;  
object-fit: scale-down;

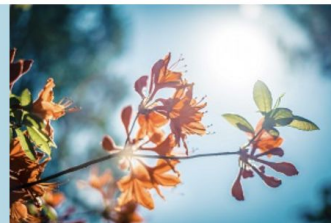


width: 300px;  
height: 150px;  
object-fit: none;

## Свойство object-position



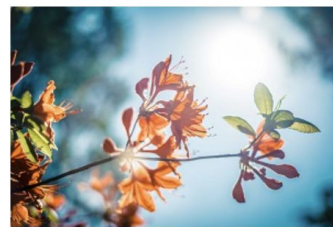
object-fit: fill;  
object-position: 20px 20px;



object-fit: contain;  
object-position: right bottom;



object-fit: cover;  
object-position: 20% 50px;



object-fit: scale-down;  
object-position: 0 0;

# object-fit

Свойство object-fit определяет, каким образом содержимое замещаемых элементов будет подогнано к краям контейнера тега в случае, когда для элемента заданы ширина и высота, отличные от его собственных размеров.

Заменяемыми элементами являются элементы, чьи размеры и внешний вид определяются внешним ресурсом. К ним относятся элементы `<img>` `<input type="image">`, `<embed>`, `<iframe>`, `<video>`, `<canvas>` и `<object>` в случае когда они содержат встраиваемое содержимое, а также элемент `<audio>`, использующий пользовательские элементы управления.

# object-fit

## object-fit

<code>fill</code>	Значение по умолчанию. Содержимое замещаемого элемента полностью заполняет область контейнера тега, используя его высоту и ширину.
<code>contain</code>	Содержимое элемента масштабируется, выравниваясь по центру с сохранением пропорций таким образом, чтобы полностью поместиться внутри контейнера.
<code>cover</code>	Содержимое элемента обрезается, выравниваясь по центру с сохранением пропорций таким образом, чтобы полностью заполнить область контейнера.
<code>scale-down</code>	Содержимое элемента выбирает из двух значений <code>none</code> и <code>contain</code> то значение, которое даёт меньшие размеры.
<code>none</code>	Замещаемое содержимое не изменяет свои собственные размеры, чтобы поместиться и заполнить область контейнера.
<code>initial</code>	Устанавливает это свойство в значение по умолчанию.
<code>inherit</code>	Наследует значение свойства от родительского элемента.

Свойство `object-fit` масштабирует, обрезает или растягивает содержимое замещаемых элементов. Свойство хорошо поддерживается в последних версиях Chrome, Firefox и Safari.

# object-position

## object-position

### Значения:

left top ,  
left center ,  
left bottom ,  
right top ,  
right center ,  
right bottom ,  
center top ,  
center  
center ,  
center  
bottom

Позиционирование по горизонтали и вертикали задаётся с помощью пары ключевых слов. Если задано одно ключевое слово, второе примет значение center .

px / em / %

Указывается два значения, первое определяет расстояние между левой стороной изображения и левым краем элемента-контейнера (по горизонтали), второе значение указывает расстояние между верхней стороной изображения и верхним краем элемента-контейнера (по вертикали). Также можно использовать отрицательные значения для обрезки части контента.  
Процентное значение рассчитывается относительно содержимого и контейнера, в котором оно расположено. Одновременно можно комбинировать значения в px, em и % .

initial

Устанавливает значение свойства в значение по умолчанию.

inherit

Наследует значение свойства от родительского элемента.

Свойство object-position используется в сочетании с object-fit и задаёт положение содержимого замещаемого элемента внутри контейнера относительно координатных осей X и Y. Значение по умолчанию 50% 50%. Свойство анимируется. Не наследуется.

# Стилизация курсора: свойство cursor

Свойство `cursor` указывает тип курсора, который будет отображаться для устройства, когда точка доступа курсора находится в пределах границ элемента. Браузеры могут игнорировать свойство над собственными элементами управления, например, полосами прокрутки. Браузеры также могут игнорировать свойство `cursor` и отображать его по своему выбору, чтобы указать различные состояния пользовательского интерфейса, например, когда страница не отвечает или когда пользователь выделяет текст.

Свойство наследуется.

```
cursor: pointer;  
cursor: auto;  
cursor: url(hand.png), url(cursor.png), pointer;  
cursor: url(cursor2.png) 2 2, pointer;  
cursor: inherit;  
cursor: initial;
```

# Цвет каретки вставки: свойство `caret-color`

Символ каретки является видимым индикатором точки вставки в элементе, в который пользователь вставляет текст (и, возможно, другой контент).

Свойство `caret-color` контролирует цвет этого видимого индикатора.

Свойство наследуется.

## Значения:

<code>auto</code>	Браузеры используют <code>currentColor</code> . Браузеры также могут автоматически корректировать цвет каретки, чтобы обеспечить хорошую видимость и контрастность с окружающим контентом, на основе <code>currentColor</code> , фона, теней и т.д. Значение по умолчанию.
цвет	Каретка вставки окрашивается указанным цветом.
<code>inherit</code>	Наследует свойство от родительского элемента.
<code>initial</code>	Устанавливает это свойство в значение по умолчанию.

```
caret-color: auto;  
caret-color: transparent;  
caret-color: currentColor;  
caret-color: red;  
caret-color: #5729e9;  
caret-color: rgb(0, 200, 0);  
caret-color: hsla(228, 4%, 24%, 0.8);
```



# Ресурсы

Background подробно - [ТЫК](#)

Работа с градиентом - [ТЫК](#)

Object-fit & Object-position, работа с изображением - [ТЫК](#)

Курсор и каретка - [ТЫК](#)