



# Анимации CSS

---



# Простые анимации

```
@keyframes rocket
{
    0% { opacity: 0; }
    100% { opacity: 1; }
}
```

```
.box {
    animation: rocket 5s infinite; /* IE 10+ */
    width: 100px;
    height: 100px;
    background: #dadada;
}
```



# Свойства анимации



```
animation-name: myAnamtion;  
animation-duration: 5s;  
animation-timing-function: easy-in;  
animation-delay: 2s;  
animation-iteration-count: 2;  
animation-direction: alternate;
```

=

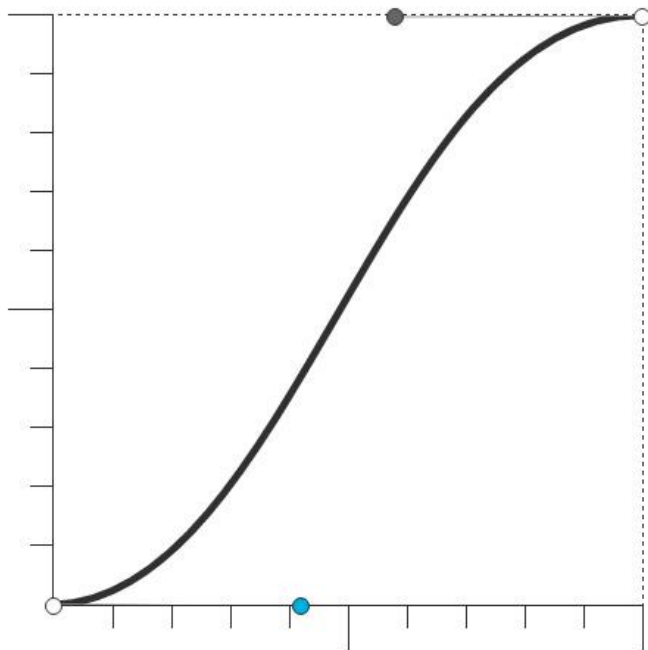
```
animation: myAnimation 5s easy-in 2s 2 alternate;
```



# animation-timing-function

## CSS cubic-bezier Builder

Drag the colored knobs to adjust the curve.



[Click here](#) to compare transition timing functions:

cubic-bezier(.42, 0, .58, 1)



linear:



ease-out:



ease-in-out:



ease-in:



```
-webkit-transition: all 1s cubic-bezier(.42, 0, .58, 1);  
-moz-transition: all 1s cubic-bezier(.42, 0, .58, 1);
```





# animation-timing-function

Значения:	
ease	Функция по умолчанию, анимация начинается медленно, разгоняется быстро и замедляется в конце. Соответствует cubic-bezier(0.25,0.1,0.25,1).
linear	Анимация происходит равномерно на протяжении всего времени, без колебаний в скорости. Соответствует cubic-bezier(0,0,1,1).
ease-in	Анимация начинается медленно, а затем плавно ускоряется в конце. Соответствует cubic-bezier(0.42,0,1,1).
ease-out	Анимация начинается быстро и плавно замедляется в конце. Соответствует cubic-bezier(0,0,0.58,1).
ease-in-out	Анимация медленно начинается и медленно заканчивается. Соответствует cubic-bezier(0.42,0,0.58,1).
cubic-bezier(x1, y1, x2, y2)	Позволяет вручную установить значения от 0 до 1.





# animation-delay

Значения:	
время	Задержка анимации задается в секундах s или миллисекундах ms. Значение по умолчанию 0.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.





## animation-iteration-count

Значения:	
число	С помощью целого числа задается количество повторов анимации. Значение по умолчанию 1.
infinite	Анимация проигрывается бесконечно.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.





# animation-direction

Значения:	
alternate	Анимация проигрывается с начала до конца, затем в обратном направлении.
alternate-reverse	Анимация проигрывается с конца до начала, затем в обратном направлении.
normal	Значение по умолчанию, анимация проигрывается в обычном направлении, с начала и до конца.
reverse	Анимация проигрывается с конца.
initial	Устанавливает значение свойства в значение по умолчанию.
inherit	Наследует значение свойства от родительского элемента.







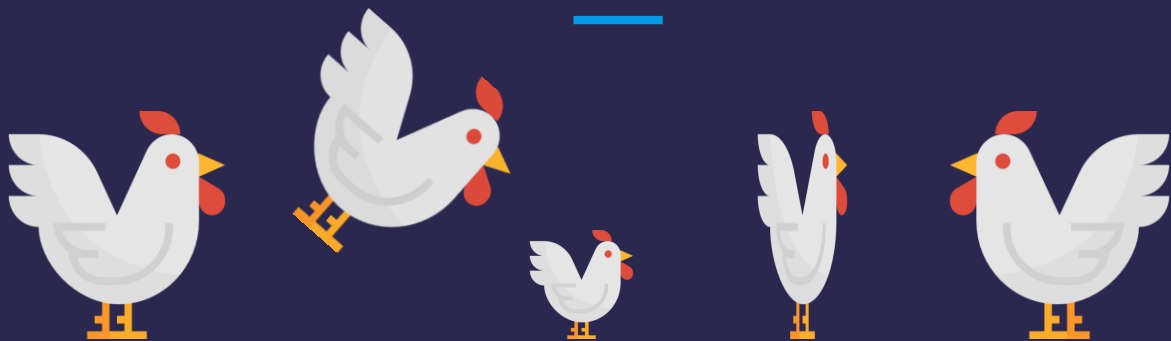
## animation-play-state

paused	Останавливает анимацию.
running	Значение по умолчанию, означает проигрывание анимации.



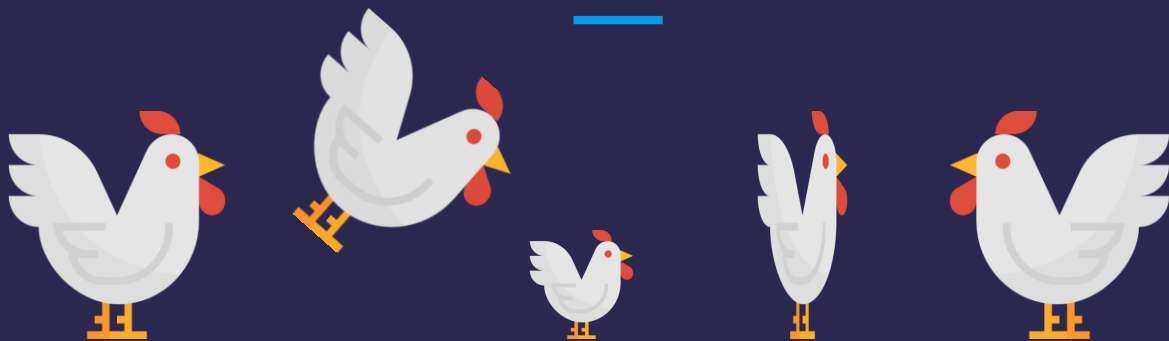


# Трансформации





# Свойства и методы



# Свойства

- transform
- transform-origin

transform-origin -  
устанавливает координаты  
точки, относительно которой  
будет происходить  
трансформация элемента.



# Методы

Функция	Описание
<code>none</code>	Значение по умолчанию, означает отсутствие трансформации. Также отменяет трансформацию для элемента из группы трансформируемых элементов.
<code>matrix(scaleX, skewX, skewY, scaleY, translateX, translateY)</code>	<p>Смещает элементы и задает способ их трансформации, позволяя объединить несколько функций 2D-трансформаций в одной. В качестве трансформации допустимы поворот, масштабирование, наклон и изменение положения.</p> <p>Значение <code>a</code> изменяет масштаб по горизонтали. Значение от 0 до 1 уменьшает элемент, больше 1 — увеличивает.</p> <p>Значение <code>c</code> деформирует (сдвигает) стороны элемента по оси <code>Y</code>, положительное значение — вверх, отрицательное — вниз.</p> <p>Значение <code>b</code> деформирует (сдвигает) стороны элемента по оси <code>X</code>, положительное значение — влево, отрицательное — вправо.</p> <p>Значение <code>d</code> изменяет масштаб по вертикали. Значение меньше 1 уменьшает элемент, больше 1 — увеличивает.</p> <p>Значение <code>x</code> смещает элемент по оси <code>X</code>, положительное — вправо, отрицательное — влево.</p> <p>Значение <code>y</code> смещает элемент по оси <code>Y</code>, положительное значение — вниз, отрицательное — вверх.</p>
<code>translate(x,y)</code>	Сдвигает элемент на новое место, перемещая относительно обычного положения вправо и вниз, используя координаты <code>x</code> и <code>y</code> , не затрагивая при этом соседние элементы. Если нужно сдвинуть элемент влево или вверх, то нужно использовать отрицательные значения.
<code>translateX(n)</code>	Сдвигает элемент относительно его обычного положения по оси <code>X</code> .
<code>translateY(n)</code>	Сдвигает элемент относительно его обычного положения по оси <code>Y</code> .



# Методы

<code>scale(x,y)</code>	Масштабирует элементы, делая их больше или меньше. Значения от 0 до 1 уменьшают элемент. Первое значение масштабирует элемент по ширине, второе — по высоте. Отрицательные значения отображают элемент зеркально.
<code>scaleX(n)</code>	Функция масштабирует элемент по ширине, делая его шире или уже. Если значение больше единицы, элемент становится шире, если значение находится между единицей и нулем, элемент становится уже. Отрицательные значения отображают элемент зеркально по горизонтали.
<code>scaleY(n)</code>	Функция масштабирует элемент по высоте, делая его выше или ниже. Если значение больше единицы, элемент становится ниже, если значение находится между единицей и нулем — ниже. Отрицательные значения отображают элемент зеркально по вертикали.
<code>rotate(угол)</code>	Поворачивает элементы на заданное количество градусов, отрицательные значения от -1deg до -360deg поворачивают элемент против часовой стрелки, положительные — по часовой стрелке. Значение <code>rotate(720deg)</code> поворачивает элемент на два полных оборота.
<code>skew(x-угол,y-угол)</code>	Используется для деформирования (искажения) сторон элемента относительно координатных осей. Если указано одно значение, второе будет определено браузером автоматически.
<code>skewX(угол)</code>	Деформирует стороны элемента относительно оси X.
<code>skewY(угол)</code>	Деформирует стороны элемента относительно оси Y.





# СВОЙСТВО transition



# Переходы: transition

Свойство transition применяется для добавления плавности перехода свойств элемента из одного состояния в другое.

**transition:** width 2s;



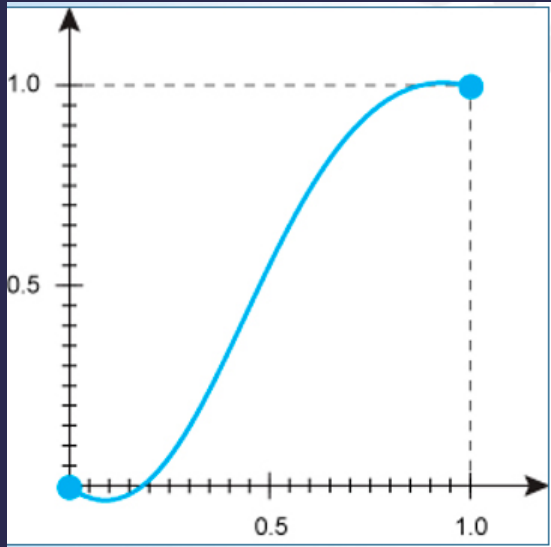


# Переходы: transition

transition-property: top;

transition-duration: 2s;

transition-timing-function: ease-in-out;





# Градиенты



# Градиенты

## Ultimate CSS Gradient Generator

A powerful Photoshop-like CSS gradient editor from [ColorZilla](#).

For Firefox

For Chrome

Gradient Generator

### Presets



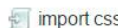
Name:



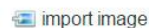
hue/saturation



reverse



import css



import image

### Stops

Opacity:

Location:

Color:

Location:

### Preview



Orientation:

Size:  x

☐ IE

### CSS

[switch to scss](#)

```
/* Permalink - use to edit and share this gradient:
http://colorzilla.com/gradient-
editor/#1e5799+0,2989d8+50,207cca+51,7db9e8+100;Blue+Gloss
*/
background: #1e5799; /* Old browsers */
background: -moz-linear-gradient(-45deg,
  hsla(212,67%,36%,1) 0%, hsla(207,69%,51%,1) 50%,
  hsla(208,73%,46%,1) 51%, hsla(206,70%,70%,1) 100%); /*
FF3.6-15 */
background: -webkit-linear-gradient(-45deg,
  hsla(212,67%,36%,1) 0%,hsla(207,69%,51%,1)
  50%,hsla(208,73%,46%,1) 51%,hsla(206,70%,70%,1) 100%);
/* Chrome10-25,Safari5.1-6 */
background: linear-gradient(135deg, hsla(212,67%,36%,1)
  0%,hsla(207,69%,51%,1) 50%,hsla(208,73%,46%,1)
  51%,hsla(206,70%,70%,1) 100%); /* W3C, IE10+, FF16+,
Chrome26+, Opera12+, Safari7+ */
filter: progid:DXImageTransform.Microsoft.gradient(
  startColorstr='#1e5799',
  endColorstr='#7db9e8',GradientType=1 ); /* IE6-9
fallback on horizontal gradient */
```

Color format:

☒ Comments



# Градиенты

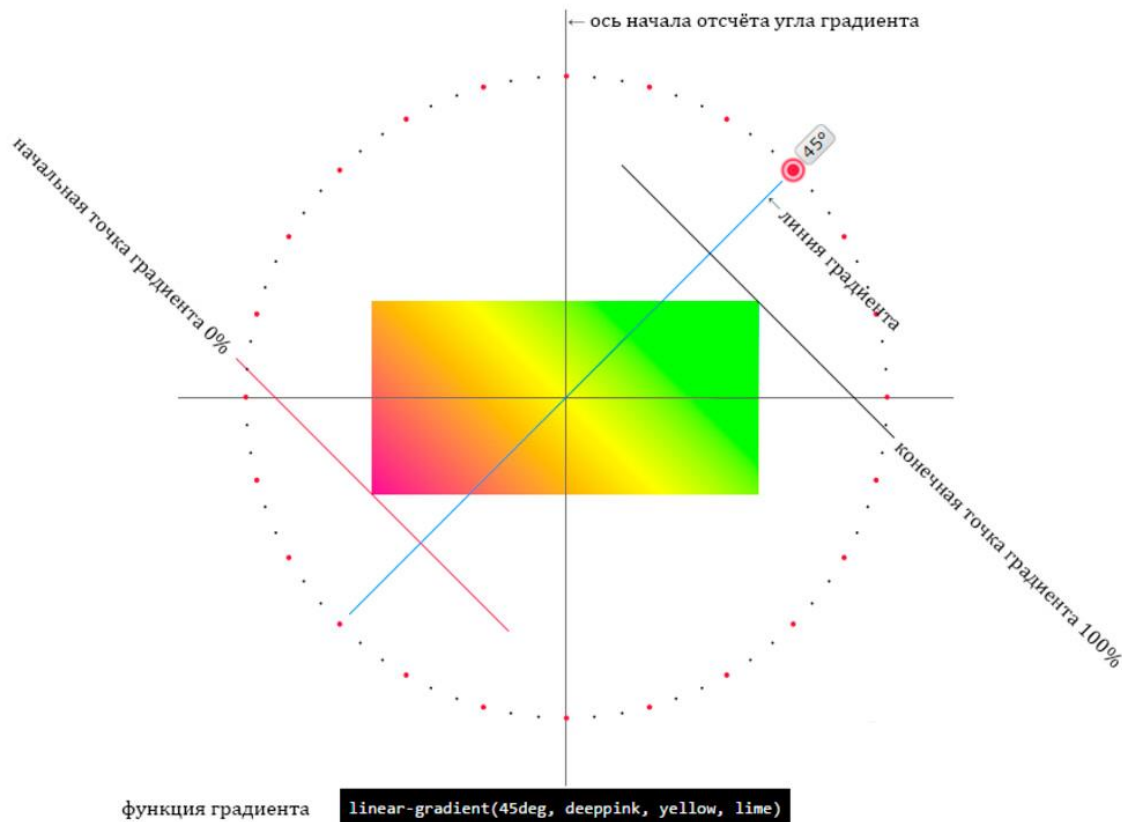
```
.gradient {  
    height: 100px;  
    background: red;  
    background: linear-gradient(red, #f06d06, blue, white);  
}
```

```
<div class="gradient"></div>
```

Result



# Линейный градиент linear-gradient()



# linear-gradient()

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



# Неравномерные градиенты

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:

**background:** linear-gradient(to left bottom, red 5%, blue 10%, yellow 15%, green);



# Радиальный градиент radial-gradient()



**background:** radial-gradient(white, blue);

**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.



# Радиальный градиент radial-gradient()

**background:** radial-gradient(circle, white, blue);



**background:** radial-gradient(ellipse, white, blue);

