



# Трансформация

CSS

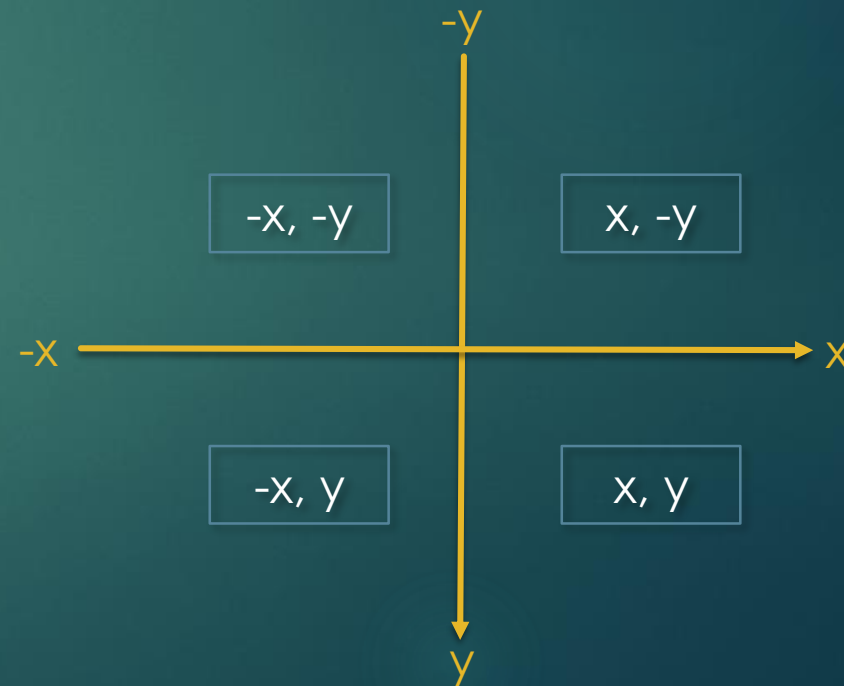
# CSS

## Трансформации

★ CSS3 трансформации определяют, каким станет тот или иной **блочный** элемент.

Трансформации преобразовывают элемент, не влияя на остальные части веб-документа, т.е. другие объекты не сдвигаются относительно него

Все трансформации CSS происходят относительно “системы координат”



Трансформации  
не оказывают влияния  
на соседние элементы

Свойство `transform` -- задает вид преобразования элемента. Свойство описывается с помощью значений функций трансформации

### Свойство `transform`

Значения:

`function()`

Пример:

```
div {  
    transform: rotate(45deg);  
}
```

#### Transform functions

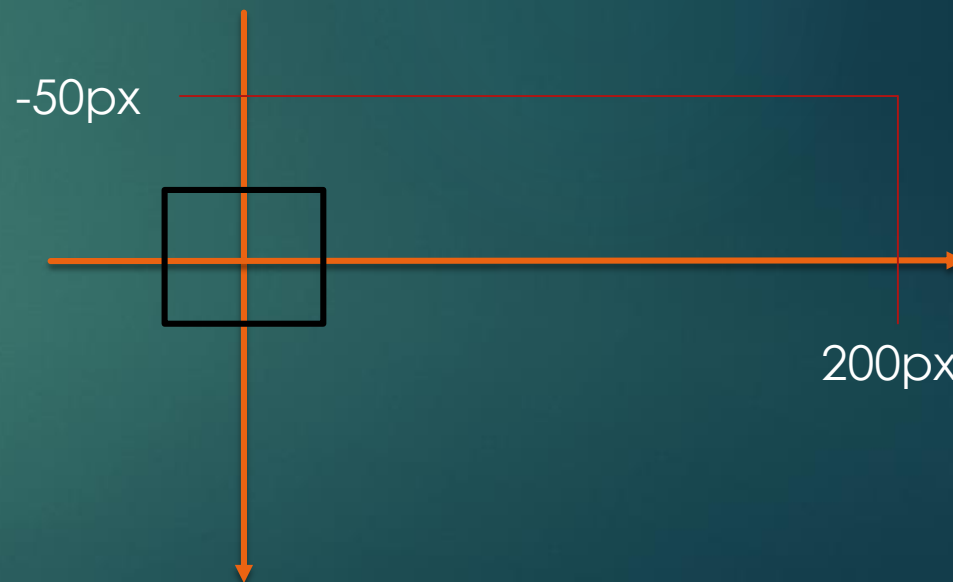
|                             |                       |
|-----------------------------|-----------------------|
| <code>translate(x,y)</code> | Сдвигает элемент      |
| <code>scale(x, y)</code>    | Масштабирует элементы |
| <code>rotate(deg)</code>    | Поворачивает элементы |
| <code>skew(deg)</code>      | Деформирует элемент   |

## Трансформации

Функция `translate(x, y)` позволяет переместить элемент на новое место, перемещая его относительно обычного положения, используя координаты `x` и `y`.  
Значением выступают единицы длины, а также проценты, могут принимать отрицательные значения

```
.one:hover{  
  transform: translate(200px, -50px);  
  transition: all 1s linear;  
}
```

`translateX(n)` – перемещает элемент по горизонтали  
`translateY(n)` – перемещает элемент по вертикали

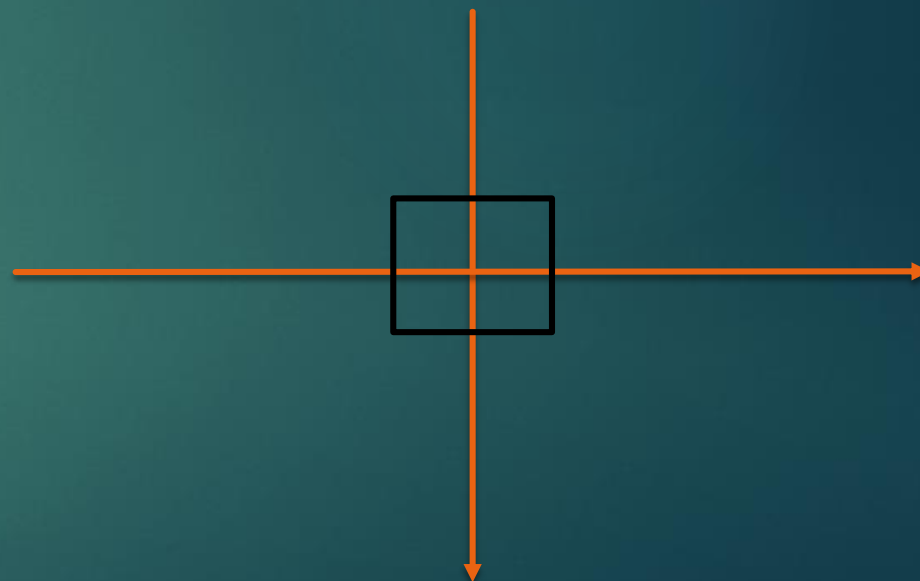


## Трансформации

Функция `scale(x, y)` масштабирует элемент относительно его текущего размера  
Значением выступают единицы измерения, проценты  
Значением может выступать множитель, тогда используем число без единиц измерения.  
Значения от 0 до 1 позволяют уменьшить элемент  
Отрицательные значения отображают элемент зеркально

```
.one:hover{  
  transform: scale(2);  
  transition: all 1s linear;  
}
```

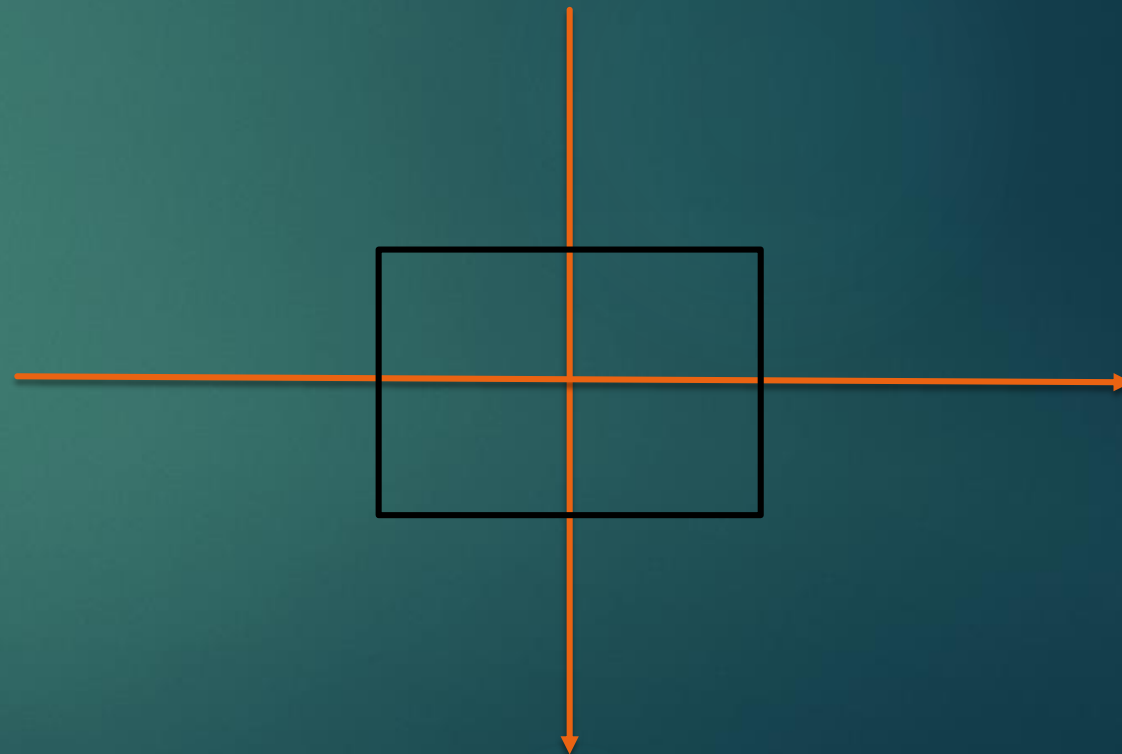
`scaleX(n)` – перемещает элемент по горизонтали  
`scaleY(n)` – перемещает элемент по вертикали



## Трансформации

Функция `rotate(n deg)` поворачиваю элемент вокруг его центра на указанный угол  
Положительные значения поворачивают элемент по часовой стрелке  
Отрицательные против

```
.one:hover{  
  transform: rotate(180deg);  
  transition: all 1s linear;  
}
```



# CSS

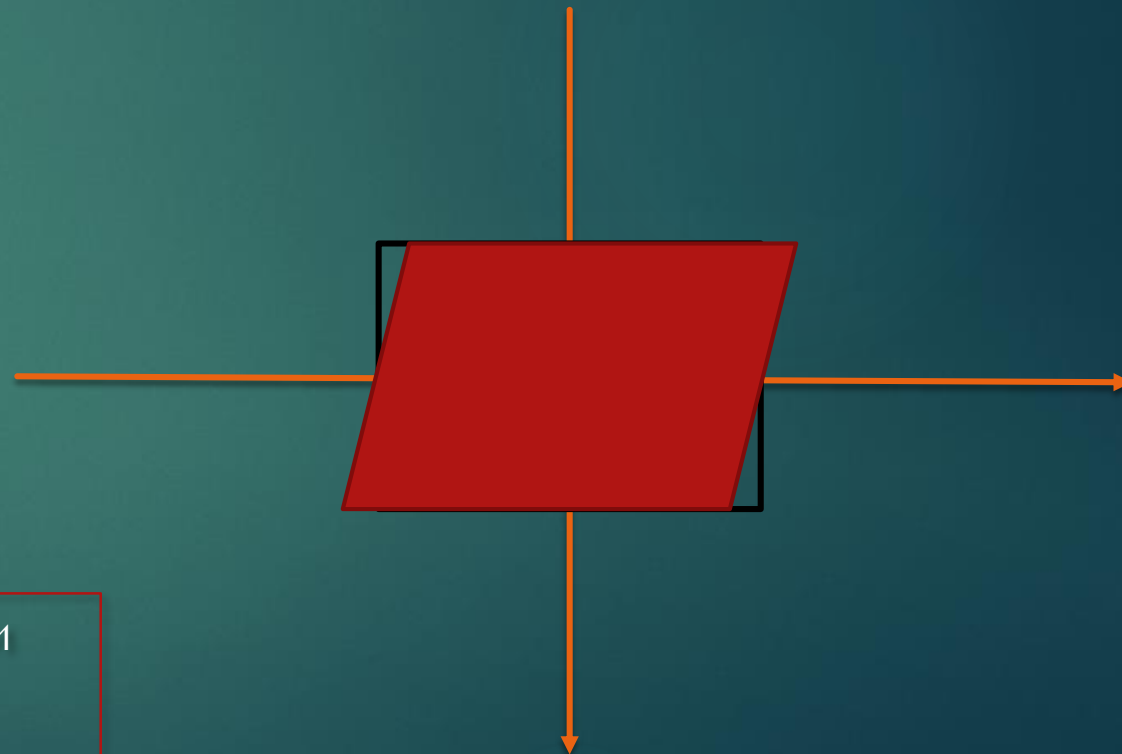
## Трансформации

transform

Функция `skew(x, y)` Используется для деформирования (искажения) сторон элемента относительно осей координат  
Значением выступает значение угла

```
.one:hover{  
  transform: skew(20deg, 0deg);  
  transition: all 1s linear;  
}
```

`skewX(n)` – деформирует элемент по горизонтали  
`skewY(n)` – деформирует элемент по вертикали





## Трансформации

Функция `matrix(scaleX, skewY, skewX, scaleY, translateX, translateY)` позволяет объединить несколько трансформаций в одной.

```
.one:hover{  
  transform: matrix(2, 0deg, 20deg, 0.5, 20px, 40px);  
  transition: all 1s linear;  
}
```

Также можно задавать цепочку трансформаций  
Для этого записываем функции трансформации через пробел в порядке проявления

```
.one:hover{  
  transform: translate(20%, 30%) rotate(90deg);  
  transition: all 1s linear;  
}
```



Свойство `transform-origin` позволяет установить точку, относительно которой происходит трансформация.

Значение по умолчанию -- `center`

### Свойство `transform-origin`

---

Значения:

`<keyword> | <длина> | %`

Пример:

```
div {  
    transform-origin: 50px 100px;  
}
```

Первое значение задает смещение по оси X, а второе по оси Y

Также допускается использование ключевых слов

# CSS

## Трансформации

transform

```
/* FireFox 3.6 - 15 */
```

```
-moz-transform: rotate(180deg);
```

```
/* Opera 11 – 12.10 */
```

```
-o-transform: skale(2);
```

```
/* Chrome10 - 25 , Safari 5.1 – 6, IE 9 - 10 */
```

```
-webkit-transform: skew(20deg, 0deg);
```

```
/* W3C, IE10+, FF16+, Chrome26+, Opera12+, Safari7+ */
```

```
transform: translate(20px 50px);
```

```
/* IE9 */
```

```
-ms-transform: rotate(45deg);
```