

INSTITUTO FEDERAL

Mato Grosso do Sul

DESENVOLVIMENTO WEB 1:
CSS Transformação e Transição

TRANSFORMAÇÕES

- As transformações são efeitos que nos permitem mudar a forma, tamanho e posição de um elemento.
- Há dois tipos: 2D e 3D.
- A propriedade usada é a "transform", e passamos uma função relacionada à propriedade que queremos alterar.

Veja mais em [MDN webdocs](https://developer.mozilla.org/pt-BR/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/css3_transforms.asp), [cssreference.io](https://cssreference.io/transform/)

TRANSFORMAÇÕES

transform: **translate**(qtde_horizontal, qtde_vertical);
move um elemento em relação à sua posição original.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: translate(50px, 100px);  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [css-tricks](#) e [translate.io](#)

- Os parâmetros passados para a função são a **quantidade** que o elemento deve se **mover** na **horizontal** (eixo X) e na **vertical** (eixo Y).
- Caso queira alterar apenas o X ou apenas o Y, existem as funções

TRANSFORMAÇÕES

`transform: translate(50px, 100px);`

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: translate(50px, 100px);  
}
```



Veja mais em [MDN webdocs](https://developer.mozilla.org/en-US/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/default.asp), cssreference.io

TRANSFORMAÇÕES

`transform: rotate(qtde_rotacao);`
rotaciona um elemento de acordo com o ângulo passado.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: rotate(20deg);  
}
```

- Por padrão, o centro do elemento é utilizado como ponto para as transformações.
- Você pode alterar este ponto utilizando a propriedade "transform-origin", passando como valor "top", "right", "bottom" ou "left".

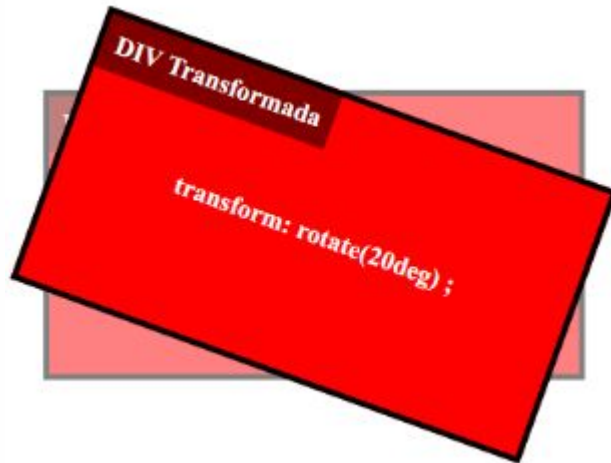
Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

TRANSFORMAÇÕES

`transform: rotate(20deg);`
rotaciona 20 graus a partir do centro do elemento.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: rotate(20deg);  
}
```



Veja mais em [MDN webdocs](https://developer.mozilla.org/en-US/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/default.asp), cssreference.io

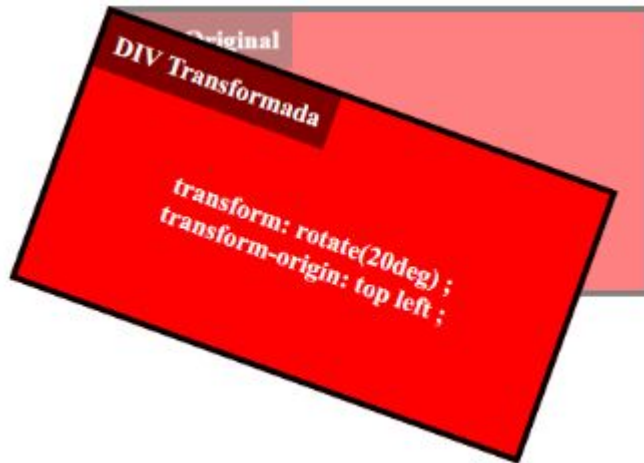
TRANSFORMAÇÕES

Caso queira que uma das pontas seja utilizada, basta juntar dois valores, como por exemplo, "top left".

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: rotate(20deg);  
  transform-origin: top left;  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference](#)



TRANSFORMAÇÕES

transform: **scale**(qtde_largura, qtde_altura);

permite redimensionar o elemento de acordo com um valor relativo

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform:scale(1.5, 0.8);  
}
```

- A função espera receber o valor da largura e depois da altura. O valor 1 seria igual a 100%, 0.5 igual a 50%.
- Caso queira mudar apenas uma das propriedades, utilize a função `scaleX()` ou `scaleY()`.

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

TRANSFORMAÇÕES

`transform: scale(1.5, 0.8);`

a largura para ser 50% maior do que o original (150%) e a altura apenas 80%

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: scale(1.5, 0.8);  
}
```

DIV Transformada

`transform: scale(1.5, 0.8);`

Veja mais em [MDN webdocs](https://developer.mozilla.org/en-US/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/css3_transforms.asp), cssreference.io

TRANSFORMAÇÕES

transform: **skew**(horizontal, vertical);
inclina o elemento de acordo com os eixos X e Y.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform:skew(20deg, 10deg);  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

- Os parâmetros passados para a função são o angulo que o elemento deve inclinar na horizontal (eixo X) e na vertical (eixo Y).
- Caso queira mudar apenas uma das propriedades, utilize a função skewX() ou skewY()

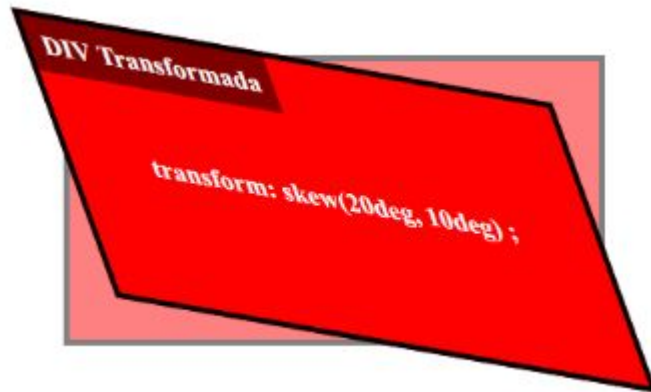
TRANSFORMAÇÕES

`transform: skew(20deg, 10deg);`

a largura para ser 50% maior do que o original (150%) e a altura apenas 80%

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform:skew(20deg, 10deg);  
}
```



Veja mais em [MDN webdocs](https://developer.mozilla.org/en-US/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/default.asp), cssreference.io

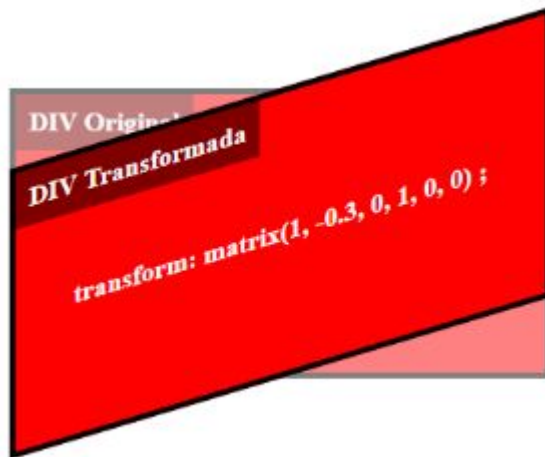
TRANSFORMAÇÕES

transform: **matrix**(scaleX, skewY, skewX, scaleY, translateX e translateY);
Esta função combina todas as funções de transformação 2D em uma só.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: matrix(1, -0.3, 0, 1, 0, 0);  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#)



TRANSFORMAÇÕES

- Além das transformações 2D, temos também as 3D. A diferença é o acréscimo do eixo Z nas funções.
- Primeiro temos que indicar um valor de perspectiva para ativar os efeitos 3D. Esse valor vai indicar a intensidade do efeito, como se fosse o tamanho do local onde o objeto 3D irá existir.
- Quanto maior o valor mais perto o elemento estará e, portanto, um efeito menor será aplicado.

- Podemos indicar a perspectiva de duas maneiras:

1 - Propriedade "perspective"

2 - Função "perspective()" na nossa já conhecida propriedade transform

Quanto maior o valor mais perto o elemento estará.

EXEMPLO DE UTILIZAÇÃO

```
/* como propriedade */  
div{  
    perspective: 500;  
}  
/* como função na propriedade  
transforme*/  
div{  
    transform: perspective(500);  
}
```

TRANSFORMAÇÕES

transform: **perspective**(proximidade) **rotate3d**(X,Y,Z);
inclina o elemento de acordo com os eixos X e Y.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform:skew(20deg, 10deg);  
}
```

- Passamos os ângulos relativos aos eixos X, Y e Z.
- Caso queira mudar apenas um dos eixos, utilize as funções rotateX, rotateY ou rotateZ.

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

TRANSFORMAÇÕES

- Nas duas figuras utilizamos `rotateX(40deg)` para fazer o elemento rotacionar no eixo X.
- Note a diferença de efeito de profundidade obtido.



TRANSFORMAÇÕES

transform: **perspective**(proximidade) **translate3d**(X,Y,Z);
inclina o elemento de acordo com os eixos X e Y.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform:skew(20deg, 10deg);  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

- Passamos os ângulos relativos aos eixos X, Y e Z.
- Caso queira mudar apenas um dos eixos, utilize as funções `translateX`, `translateY` ou `translateZ`.

TRANSFORMAÇÕES

- Aqui utilizamos a função `translateZ` com o valor `-100px`.
- Parece que nós redimensionamos o elemento, mas na verdade nós fizemos ele se afastar da tela.



TRANSFORMAÇÕES

transform: **perspective**(proximidade) **scale3d**(X,Y,Z);
inclina o elemento de acordo com os eixos X e Y.

EXEMPLO DE UTILIZAÇÃO

```
div{  
  height: 200px;  
  width: 400px;  
  background-color: red;  
  border: 5px solid black;  
}  
div:hover{  
  transform: skew(20deg, 10deg);  
}
```

Veja mais em [MDN webdocs](https://developer.mozilla.org/en-US/docs/Web/CSS/transform), [w3schools](https://www.w3schools.com/css/css3_transforms.asp), cssreference.io

- Passamos os valores relativos aos eixos X, Y e Z.
- Caso queira mudar apenas um dos eixos, utilize as funções `scaleX`, `scaleY` ou `scaleZ`.

- As transições fazem parte do CSS3 e nos permitem fazer com que os valores das propriedades sejam alterados suavemente.
- Imagine a largura de uma <div>. Em um momento queremos que ela tenha 200px de largura, e num segundo momento queremos que ela tenha 300px. Ao invés de ir de um valor para o outro de uma vez, podemos fazer com que haja algo mais suave, fazendo a largura ir crescendo até chegar aos 300px, ou seja, havendo uma suave transição.

TRANSIÇÕES

- As transições são a base para vermos como criar animações.
- Criar transições nos ajudam a fazer o usuário focar nos elementos importantes da nossa página e entender melhor como alguma funcionalidade funciona, dando mais realidade aos elementos.
- Mas não devemos exagerar. Criar várias animações desnecessárias deixará seu layout muito carregado e a página ficará lenta, principalmente em dispositivos móveis.

PROPRIEDADE TRANSITION

transition

permite definir a transição entre dois estados de um elemento

A propriedade "transition" é uma simplificação de quatro propriedades que podemos declarar separadamente:

- **transition-property:** indica a propriedade a ser animável;
- **transition-duration:** tempo de duração da transição;
- **transition-timing-function:** função da curva de velocidade;
- **transition-delay:** atraso antes da transição iniciar.

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

TRANSIÇÕES

- O que a propriedade transition faz é comparar os valores das propriedades em comum entre os dois estados do link ou de qualquer outro elemento, assim ela modifica suavemente os valores quando há a ativação da função.
- Esta é uma técnica simples e que serve para manipularmos transições básicas como cor, tamanho, posição etc.

PROPRIEDADE TRANSITION

transition: [propriedade] [duração] [aceleração] [atraso];

EXEMPLO DE UTILIZAÇÃO

```
div{
  background-color: red;
  border: 5px solid black;
  transition: all 1s ease 0.5s;
}
div:hover{
  transform:skew(20deg, 10deg);
}
```

EXEMPLO DE UTILIZAÇÃO

```
div{
  background-color: red;
  border: 5px solid black;
  transition-property: all;
  transition-duration: 1s;
  transition-timing-function: ease;
  transition-delay: 0.5s;
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

Curva da Velocidade da Transição

transition-timing-function

indica a curva de aceleração do elemento

EXEMPLO DE UTILIZAÇÃO

```
div{  
  background-color: red;  
  border: 5px solid black;  
  transition: all 1s;  
  transition-timing-function: ease;  
}  
div:hover{  
  transform: skew(20deg, 10deg);  
}
```

Veja mais em [MDN webdocs](#), [w3schools](#), [cssreference.io](#)

Curva da Velocidade da Transição

- Quando criamos uma transição, temos algo chamado de "Curva da Velocidade de Transição".
- Essa propriedade também é muito presente em softwares de animação, também conhecida como "Curva de Animação". Ela indica a velocidade em que a animação será realizada.

Curva da Velocidade da Transição

valores pré-definidos:

- **ease** a transição começa acelerando e termina desacelerando
- **linear** a transição mantém uma velocidade constante
- **ease-in** a transição começa acelerando e mantém uma velocidade constante até o final
- **ease-out** a transição começa com uma velocidade constante e vai desacelerando no final
- **ease-in-out** a transição é lenta no começo e no final, mantendo uma velocidade constante no meio
- **cubic-bezier(n,n,n,n)** permite que a gente defina nossos próprios valores para a curva da velocidade

MATERIAIS COMPLEMENTARES

- [MDN - Usando CSS transforms](#)
- [medium - css3 transform e transition](#)
- [devmedia - transformacoes-3d-em-css3-na-pratica](#)
- [caelum-html-css-javascript](#)
- [tableless - exemplos css transforms](#)
- [exemplo css transformação 3d](#)

REFERÊNCIAS

- <https://htmlreference.io/>
- <https://cssreference.io>
- <https://css-tricks.com/snippets/css/>
- <https://tympanus.net/codrops/category/playground/>
- <https://www.awwwards.com/>
- <https://dribbble.com/>
- <https://www.typewolf.com/>