



## margin, padding e border

**Margin, padding e border** são três regras importantes do CSS que nos permitem definir espaçamentos no entorno e dentro do elemento.

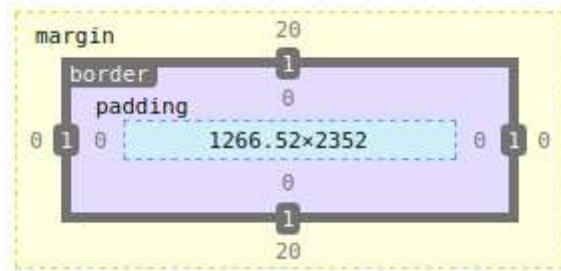


Imagem 1 - Margin, padding e border (Fonte da imagem: do autor)

## Margin

É uma margem em volta do elemento, usamos na maioria das vezes para "desgrudar" o elemento de outros elementos.

Margens podem ser definidas por um valor (todos os lados com mesma margem), com dois valores (cima e baixo com um valor e direita e esquerda com outro) ou com 4 valores (um para cada lado em sentido horário começando do topo).

*Exemplos de **Margin**:*

```
div {  
  margin: 10px;  
} /* Todos os lados com margem 10px */  
  
div {  
  margin: 10px 5px;  
} /* margem de 10px no topo e embaixo e de 5px dos > lados*/  
  
div {  
  margin: 1px 2px 3px 4px;  
} /* Topo 1px, direita 2px, baixo 3px e esquerda > 4px*/  
  
/* Lados determinados em sentido horário (top, right, bottom, left) */
```

Ainda podemos determinar uma margem apenas de um dos lados:

*Exemplos de **margin** por lado:*

```
div {  
  margin-top: 10px;  
}  
  
div {  
  margin-right: 10px;
```

```
}
```

```
div {  
  margin-bottom: 10px;  
}
```

```
div {  
  margin-left: 10px;  
}
```

Da mesma forma que dissemos que os arquivos de CSS se sobreescrevem, as regras também o fazem. Isso quer dizer que você pode usar uma regra mais genérica e depois sobreescrevê-la com uma mais particular.

Por exemplo:

*Sobrescrevendo uma das margens*

```
div {  
  margin: 10px;  
}
```

```
div {  
  margin-left: 20px;  
}
```

O resultado será que todas as margens são setadas para 10px e depois a margem esquerda é sobrescrita para o valor 20px.

# Padding

---

Pense no `padding` como uma margem interna, todos os elementos ou textos serão afastados das bordas do objeto pai de acordo com os tamanhos definidos.

Usar `padding` faz o tamanho do elemento aumentar. Para alterar a forma como é feito esse cálculo de larguras e alturas dos elementos, podemos utilizar a propriedade `box-sizing`.

As propriedades de altura (`height`) e largura (`width`) de um elemento são calculadas levando em conta apenas o seu conteúdo. Com isso, propriedades como `padding`, `margin` ou `border` não são levadas em conta.

A `box-sizing` pode ser definida como `content-box` ou `border-box`:

- `content-box`: esse é o comportamento padrão, onde a altura e largura do elemento levam em conta apenas o conteúdo;
- `border-box`: ao utilizar o `box-sizing: border-box`, definimos que a largura e a altura do elemento agora incluirão, também, as propriedades `padding`, `margin` e `border`.

Para definir o `padding`, faça do mesmo jeito do `margin` usando 4, 2 ou 1 valor.

*Exemplos de `Padding`:*

```
div {  
  padding: 10px;  
} /* Todos os lados com padding 10px */
```

```
div {  
  padding: 10px 5px;
```

```
} /* padding de 10px no topo e embaixo e de 5px dos > lados*/

div {
  padding: 1px 2px 3px 4px;
} /* Topo 1px, direita 2px, baixo 3px e esquerda > 4px*/

/* Lados determinados em sentido horário (top, right, bottom, left) */
```

*Exemplos de **padding** por lado:*

```
div {
  padding-top: 10px;
}

div {
  padding-right: 10px;
}

div {
  padding-bottom: 10px;
}

div {
  padding-left: 10px;
}
```

*Sobrescrevendo o **padding** de um dos lados:*

```
div {  
  padding: 10px;  
}
```

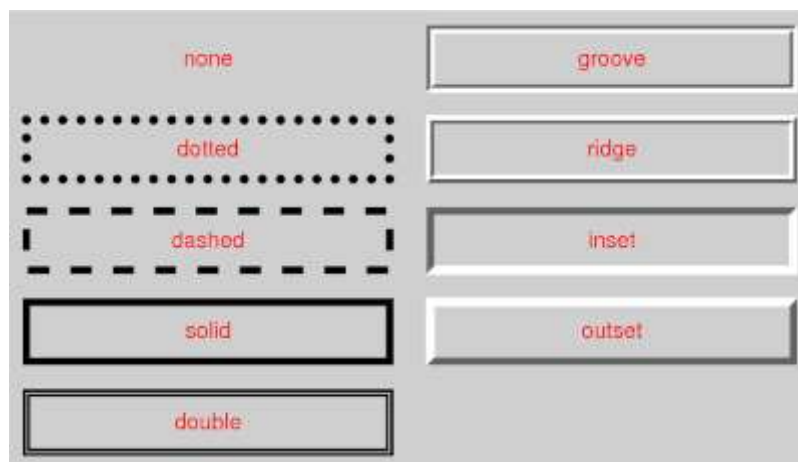
```
div {  
  padding-left: 20px;  
}
```

## Border

---

Bordas tem 3 valores, o estilo, o tamanho e a cor. Desses, apenas o estilo é obrigatório. Esses 3 valores são chamados de **line-width**, **line-style** e **color** nas documentações. Os valores possíveis de **line-style** são predefinidos, escolha um entre:

**none** | **hidden** | **dotted** | **dashed** | **solid** | **double** | **groove** | **ridge** | **inset** | **outset**



## Imagem 2 - Tipos de borda (Fonte da imagem: do autor)

### Remove as bordas

```
div {  
  border: none;  
}
```

### Borda solida com cor vermelha

```
div {  
  border: solid red;  
}
```

### Resultado:



### Borda solida com cor vermelha e 5px de espessura

```
div {  
  border: 5px solid red;  
}
```

*Resultado:*



A ordem dos valores não é relevante, no geral usamos tamanho, estilo, cor.

## Alinhamento de elementos

---

Quando estivermos trabalhando com elementos aninhados em outros elementos, precisamos lembrar de descontar o **padding** do elemento pai, pois ele mudará o posicionamento dos elementos filhos, por exemplo:

```
<div style="padding:10px;">
  <div style="height:100px; background-color:#c1c1c1"></div>
</div>
<br />
<div style="">
  <div style="height:100px; background-color:#c1c1c1"></div>
</div>
```



Os dois retângulos acima são `div` dentro de outra `div`. Uma das `div` pai tem `padding` a outra não.

Isso desalinha as duas `div` internas, apesar delas serem iguais.

Se não tivermos controle do pai, se foi criado por uma biblioteca ou usa o `padding` para posicionar outro elemento, por exemplo, podemos remediar a situação com um valor negativo de `margin` igual ao `padding` do pai, o que o anula.

```
<div style="padding:10px;">
  <div style="height:100px; margin: -10px; background-color:#c1c1c1"></div>
</div>
<br />
<div style="">
  <div style="height:100px; background-color:#c1c1c1"></div>
</div>
```

Sendo assim, leve sempre em conta que mudanças de `padding` e `margin` afetam o posicionamento dos elementos e o alinhamento com demais elementos da página e seu uso deve ser feito com cuidado para não gerar problemas, especialmente em layouts responsivos, onde os elementos mudam de lugar para se adequar às telas menores.

## Bônus

---

Bordas podem ser bastante grosseiras mesmo usando apenas 1px de espessura, podemos dar uma amenizada nisso usando cores RGBA, que permitem transparência...

```
<div style="width:200px;height:50px;border:1px solid black"></div>  
<br />  
<div style="width:200px;height:50px;border:1px solid rgba(0,0,0,0.3)"></div>
```





Essa técnica é interessante porque a borda não está apenas mais clara, ela está deixando passar a cor de fundo, no caso branca. Isso significa em outras cores de fundo essa borda se transforma em uma versão pouco mais escura da cor de fundo.

Outra técnica muito usada para fazer cards é de usar `box-shadow` no lugar da borda, isso está fora do escopo dessa lição, mas vamos deixar um exemplo:

```
<div style="width:200px;height:50px;border:1px solid black"></div>  
<br />  
<div style="width:200px;height:50px;box-shadow:0 0 5px rgba(0,0,0,0.3)"></div>
```



Você pode conhecer melhor esse conceito usando [esse gerador](#).

## Referências e Links úteis

---

- [CSS Margin](#). MDN Web Docs, 04/06/2021. Acesso em: 27/06/2022.
- [CSS Padding](#). MDN Web Docs, 27/11/2021. Acesso em: 27/06/2022.
- [Border](#). MDN Web Docs, 05/04/2022. Acesso em: 27/06/2022.
- [CSS Box Model](#). W3 Schools. Acesso em: 27/06/2022.
- [Box-sizing](#). MDN Web Docs, 17/07/2021. Acesso em: 30/06/2022.
- [CSS box-sizing Property](#). W3 Schools. Acesso em: 30/06/2022.

Próximo Tópico