

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>padding</title>
<style>

    * {
        box-sizing: border-box;
        /* Usamos a propriedade box-sizing com o valor border-box, para desligar o nosso "content-box" padrão. Isso faz com
        que os nossos elementos mantenham o tamanho padrão, independente de nós usarmos padding neles, ou não.*/
    }

    body {
        margin: 0;
        padding: 0;
        /*Aqui nós temos o uso de padding com apenas um valor, trazendo um padding 0 para todo o padding da nossa body*/
    }

    .bloco {
        width: 50vw;
        height: 50vh;
        background-color: blue;
        padding: 50px 15px;
        /* Perceba que graças ao padding, a nossa div "bloco" ganha espaçamentos internos acima e embaixo de 50px e na
        direita e esquerda de 15px. Mas ainda assim, por estarmos usando "box-sizing" com o valor "border-box" ela não
        permite que a div "bloco" ultrapasse o tamanho de 50vw na largura.*/
    }

    .item {
        width: 100%;
        height: 100%;
        background-color: skyblue;
        padding: 50px 20px 5px 35px;
```

```
/*Perceba que dessa vez demos a div item paddings diferenciados para cada posição dentro da div, e temos uma div
interna a ela de cor branca para mostrar o padding da nossa div "item". Por ter 100% de tamanho referente a div pai,
a div "item" não ultrapassará o tamanho de 50% de largura da nossa tela por causa da propriedade "box-sizing" com o
valor "border-box".*/
}

.item2 {
  width: 100%;
  height: 100%;
  background-color: white;
  /* Criamos um segundo item dentro do item para poder ver o padding do item 1. Por ter 100% de tamanho referente a
div pai dela que no caso é "item", a div "item2" não ultrapassará o tamanho de 50% de largura da nossa tela por
causa da propriedade "box-sizing" com o valor "border-box".*/
}

</style>
</head>
<body>

<!--
```

Propriedade: padding (espaçamento interno)

Objetivo: é uma propriedade que define um valor para o espaçamento interno, porém ela é uma forma abreviada de definirmos os espaçamentos dos lados top, right, bottom e left, sem ter que usar as propriedades individuais do padding para isso. Lembrando que, quando falamos de padding estamos nos referindo ao espaçamento que existe a partir de dentro da borda do elemento até o conteúdo do elemento.

Geralmente usamos o padding para facilitar a escrita e definir o valor de espaçamento sobre mais de uma posição de uma única vez.

- Valores Aceitáveis:

- length: Qualquer valor absoluto ou relativo;
- initial: Volta ao valor inicial;

- inherit: Para herdar o valor do elemento pai;

- Valor Padrão: 0;

- Possui valor herdado? - NÃO;

OBS: Cuidado com os "Side Effect" (Efeito Colateral) sobre os padding, os padding por padrão aumentam o tamanho dos nossos elementos. Por isso podemos ter uma quebra de layout indesejada. Para evitar isso temos 2 possibilidades:

- Compensar a quebra de layout por compensar o tamanho de conteúdo dos nossos elementos, essa é de longe o método menos utilizado;
- E temos o método mais utilizado que é usando a propriedade box-sizing, usando o valor "border-box", que vai fazer o nosso navegador forçar o elemento a manter o tamanho independente do seu padding.

Podemos usar a nossa propriedade padding de 3 formas:

- padding: 1únicovalor - quando usamos um único valor ele equivale para as 4 posições;
- padding: 1º 2ºvalores - também conhecido como "padding collapse", é quando usamos 2 valores que equivalem a: 1º valor para altura e baixo, 2º valor para esquerda e direita;
- padding: 1º 2º 3º 4ºvalores - quando especificamos cada valor: cima, direita, baixo e esquerda;

-->

```
<div class="bloco">
```

```
  <div class="item">
```

```
    <div class="item2"></div>
```

```
  </div>
```

```
</div>
```

```
</body>
```

```
</html>
```