

PROGRAMAÇÃO WEB I

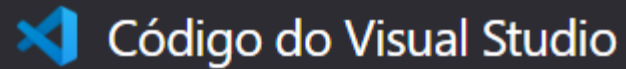
Cascading Style Sheets

Folhas de Estilo em Cascata



```
body {  
  font: x-small;  
  background: #  
  color: black;  
  margin: 0;  
  padding: 0;
```

Sugestões de Ferramentas de Desenvolvimento



<https://code.visualstudio.com/>



<https://www.sublimetext.com/>

Sugestões de Tutoriais

<https://www.w3schools.com/css/default.asp>

<https://alunos.b7web.com.br/curso/html5-e-css3/>

CSS

CSS, sigla em inglês para Cascading Style Sheet, que em português quer dizer Folha de Estilo em Cascata, é uma linguagem que cuida da apresentação visual de páginas web através de regras de estilos. Podemos resumir que é uma linguagem de estilização ou apresentação.

Esta linguagem consiste em dar forma aos elementos HTML presentes na página web. O CSS só existe em função do HTML.



HTML



HTML+CSS







HTML



HTML + CSS



HTML + CSS + JavaScript

CSS	HTML	Javascript
Estiliza o site	Cria a estrutura do site	Aumenta a interatividade
Adiciona estilo para os elementos da página	Controla o layout do conteúdo	Adiciona interatividade no site
Segmenta vários tipos de tela para que a página fique responsiva	Fornece uma estrutura para os padrões da web	Lida com complexas funções lógicas
Lida primeiramente com a visualização da página	É o fundamento básico para criar qualquer tipo de site	Linguagem de programação que melhora a funcionalidade da aplicação

Regra CSS e sua sintaxe

Uma regra CSS é uma declaração que segue uma sintaxe própria e que define como será aplicado estilo a um ou mais elementos HTML.

Um conjunto de regras CSS formam uma ***Folha de Estilos***.

Uma regra CSS, na sua forma mais elementar, compõe-se de três partes:

seletor { **propriedade:** **valor;** }

Seletor

p {
 color: **red**;
}

Valor

Propriedade

SELETOR: é o elemento HTML (ou uma classe a aplicar a um elemento HTML) para o qual a regra será válida. (por exemplo: `<p>`, `<h1>`, `<form>`, `minhaclasse`, etc...);

PROPRIEDADE: é o atributo do elemento HTML ao qual será aplicada a regra (por exemplo: `font`, `color`, `background`, etc...);

● Seletor

● Propriedade ● Valor

Regra

h1 {
 color: **#333**;
 font-size: **12px**;
 text-align: **center**;
}

VALOR: é a característica específica a ser assumida pela propriedade (por exemplo: letra tipo arial, cor azul, fundo verde, etc...).

Na sintaxe de uma regra CSS, escreve-se o seletor e a seguir a propriedade e valor separados por dois pontos e entre chaves { }.



ESTILOS DO CSS

CSS INLINE

O CSS inline é usado para dar estilo a um elemento HTML específico. Para este estilo de CSS você somente vai precisar adicionar o atributo **style** para cada tag HTML. Este tipo de CSS não é realmente recomendado, já que cada tag HTML precisa ser estilizada de maneira individual. Gerenciar o seu site pode se tornar uma tarefa bem difícil de você só usa o CSS inline.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Estilo de inline</title>
5
6   <body style="background-color:black;">
7
8   <h1 style="color:deeppink; font-family:Algerian"> Tipos de CSS</h1>
9   <p style="color:white; font-family:Brush Script MT;
10  font-size:70px"> Exemplo de CSS inline </p>
11
12 </ul>|
13 </body>
14 </html>
```

```
<!DOCTYPE html>
<html>
<head>
  <title>Estilo de inline</title>

  <body style="background-color:black;">

  <h1 style="color:deeppink; font-family:Algerian"> Tipos de CSS</h1>
  <p style="color:white; font-family:Brush Script MT;
font-size:70px"> Exemplo de CSS inline </p>

</ul>
</body>
</html>
```

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Estilo de inline</title>
5
6     <body style="background-color:black;">
7
8     <h1 style="color:deeppink; font-family:Algerian"> Tipos de CSS</h1>
9     <p style="color:white; font-family:Brush Script MT;
10 font-size:70px"> Exemplo de CSS inline </p>
11
12 </ul>|
13 </body>
14 </html>
```

CSS INTERNO

O CSS interno ou incorporado requer que você adicione a tags <style> na seção <head> do seu documento HTML.

Este estilo de CSS é um método efetivo de estilizar uma única página. Contudo, usar esse estilo em múltiplas páginas pode consumir muito tempo, já que você precisa definir as regras CSS para cada página do seu site.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        background-color: blue;
      }
      h1 {
        color: red;
        font-family: Algerian;
        font-size: 40px;
      }
    </style>
  </head>
  <body>
    <h1>Tipos de CSS</h1>
    <p>Exemplo de CSS interno</p>
  </body>
</html>
```

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <style>
5          body {
6              background-color: blue;
7          }
8          h1 {
9              color: red;
10             font-family: Algerian;
11             font-size: 40px;
12         }
13     </style>
14 </head>
15 <body>
16
17     <h1>Tipos de CSS</h1>
18     <p>Exemplo de CSS interno</p>
19
20 </body>
21 </html>
```

```
teste_Sublo.html  teste_interno.html
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <style>
5          body {
6              background-color: blue;
7          }
8          h1 {
9              color:red;
10             font-family:Algerian;
11             font-size:40px;
12         }
13     </style>
14 </head>
15 <body>
16
17     <h1>Tipos de CSS</h1>
18     <p> Exemplo de CSS interno</p>
19
20 </body>
21 </html>
```


CSS EXTERNO

Com o CSS externo, você vai linkar as páginas da internet com um arquivo .css externo. Este tipo de CSS é um método mais eficiente, especialmente se você está estilizando um site grande. Ao editar um arquivo .css, você pode modificar um site inteiro de uma só vez.

```
<!DOCTYPE html>
<html>
<head>
  <title> Estilo Externo</title>
  <link rel="stylesheet" type="text/css"
href="estilo.css">
</head>
<body>

  <h1>Tipos de CSS</h1>
  <p> Exemplo de CSS Externo</p>

</body>
</html>
```

```
body{
  background-color: green;
}

h1 {
  color:white;
  font-family:Comic Sans MS;
  font-size:60px;
}
```

Css_Externo.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> Estilo Externo</title>
5   <link rel="stylesheet" type="text/css" href="estilo.css">
6 </head>
7 <body>
8
9   <h1>Tipos de CSS</h1>
10  <p> Exemplo de CSS Externo</p>
11
12 </body>
13 </html>
```

Estilo.css

```
1 body{
2   background-color: green;
3 }
4
5 h1 {
6   color:white;
7   font-family:Comic Sans MS;
8   font-size:60px;
9 }
```

Estrutura de Pasta e CSS

Antes de continuar e começar a criar nossos exemplos , temos que lembrar da estrutura de pastas. Isso irá facilitar e organizar o desenvolvimento.



COMENTÁRIOS

HTML

<!-- Isto é um comentário de uma única linha -->

CSS

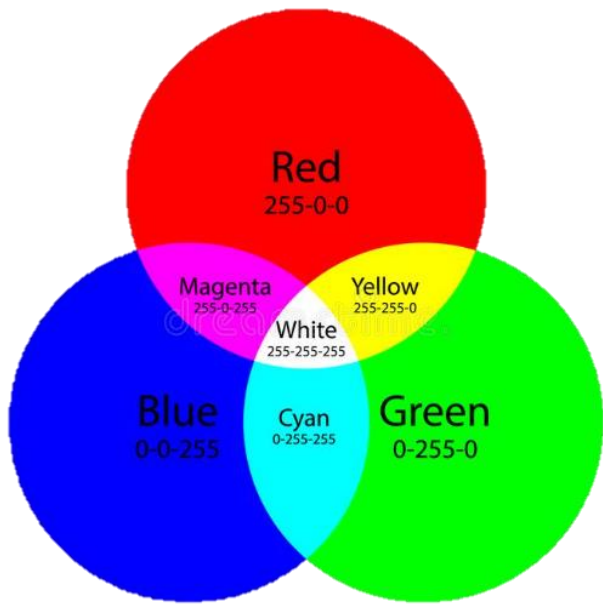
/* Este é um comentário em bloco em CSS */

```
/* h1{  
    color: red;  
} */
```

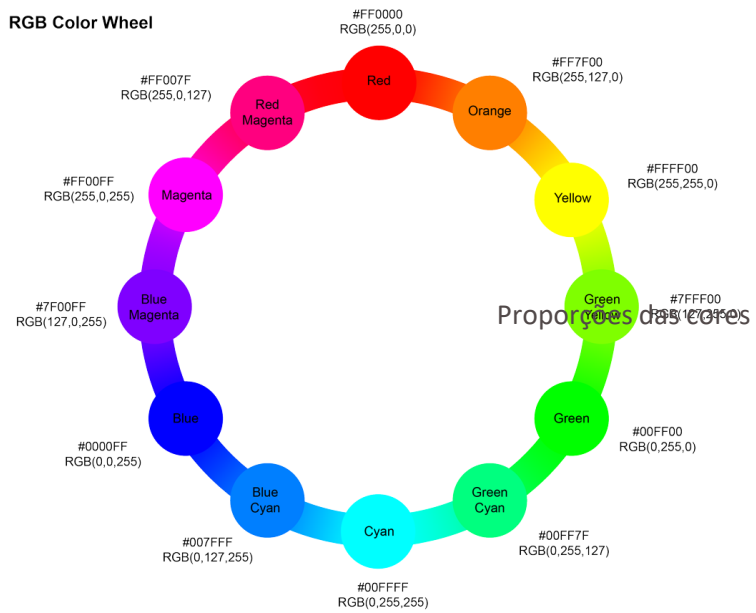
```
/*  
h1{  
    color: red;  
}  
*/
```



uma pausa para falar de cores...

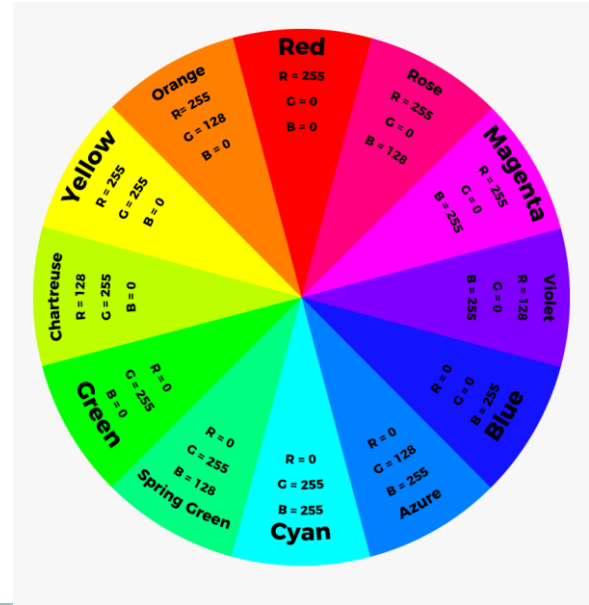


RGB Color Wheel



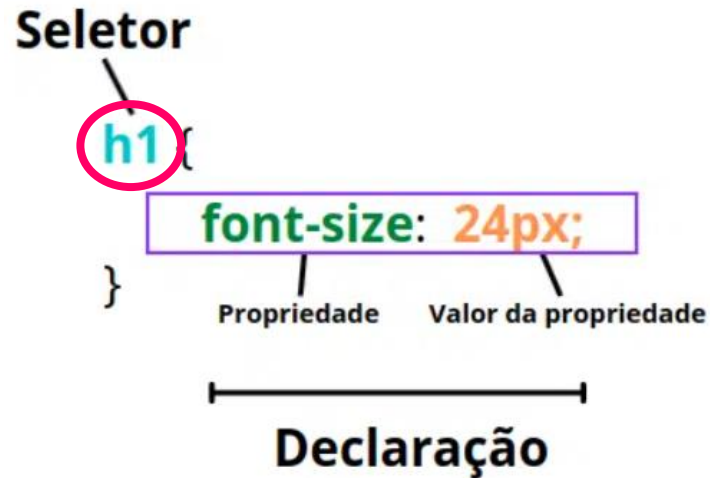
For more creative resources, please visit www.timvandevall.com | All content created by Tim van de Vall | © 2013 Dutch Renaissance Press.

A screenshot of a color selection dialog box. It features a 'Padrão' (Default) tab and a 'Personalizar' (Customize) tab. The 'Personalizar' tab is active, showing a color spectrum and a grayscale bar. Below the spectrum, there are input fields for 'Modelo de Cores' (Color Model) set to 'RGB', and individual channels for 'Vermelho' (Red), 'Verde' (Green), and 'Azul' (Blue), each with a numeric input field and a slider. A 'Hex' field is also present, showing '#000000'. Buttons for 'OK', 'Cancelar' (Cancel), and 'Nova' (New) are visible.



<https://color.adobe.com/>

SELETORES

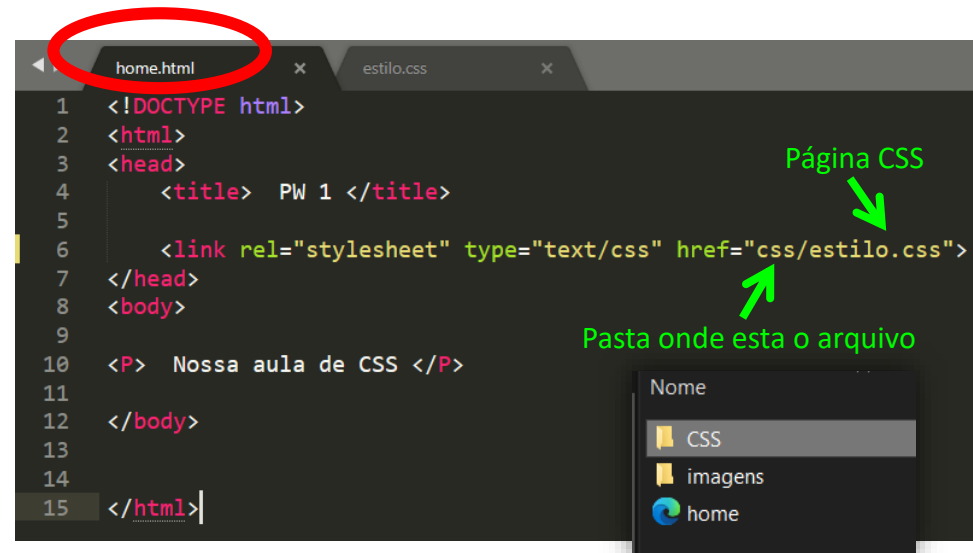


Os seletores são a declaração que o CSS utiliza para identificar os elementos do HTML. Por meio dos seletores, podemos aplicar a formatação em quase qualquer parte de uma página *web*. Existem inúmeros seletores, no entanto, listaremos apenas os mais importantes.

SELETOR POR ELEMENTO : Este seletor corresponde aos nomes das tags dos elementos html, por exemplo **p**, **a**, **h1** ou **div**. Isso significa que a regra será aplicada a **todos** os elementos desse tipo que aparecem no documento.

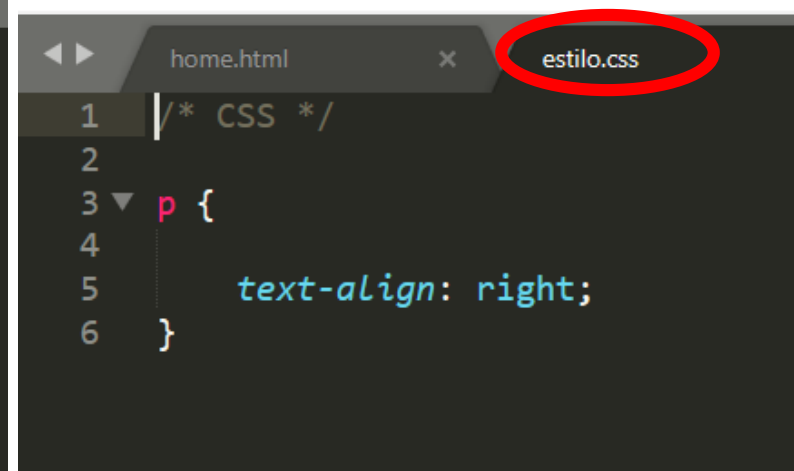
Por exemplo, se queremos que todos os parágrafos tenham um determinado alinhamento ou que todos os links tenham uma certa cor.

HTML



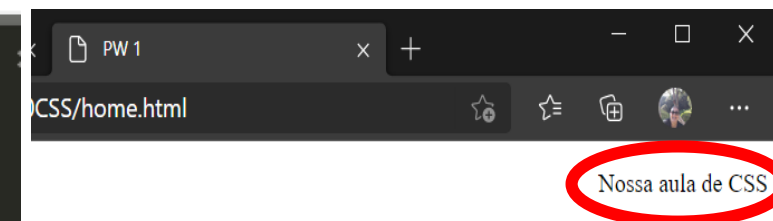
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> PW 1 </title>
5
6   <link rel="stylesheet" type="text/css" href="css/estilo.css">
7 </head>
8 <body>
9
10 <p> Nossa aula de CSS </p>
11
12 </body>
13
14
15 </html>
```

CSS



```
1 /* CSS */
2
3 p {
4
5   text-align: right;
6 }
```

Visualização



SELETOR POR ID

Os elementos html podem ter um atributo chamado *id* que deve ser **único** e serve para identificá-lo dentro do documento. O seletor deve começar com o símbolo **#** seguido pelo **id** do elemento e a regra será aplicada somente a esse elemento **NÃO** recomendado sua repetição

HTML

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> PW 1 </title>
5   <link rel="stylesheet" type="text/css" href="css/estilo.css">
6 </head>
7 <body>
8   <p id="importante"> Nossa aula de CSS </p>
9 </body>
10 </html>
```

CSS

```
1 /* CSS */
2
3
4 #importante {
5
6   background-color: coral; color: white;
7 }
8
```

Visualização



SELETOR POR CLASSE

Os elementos html podem ter um atributo chamado **class** que serve para agrupar diferentes elementos de um mesmo identificador. O seletor deve começar com o símbolo (**.**) seguido pelo nome da classe e a regra será aplicada a todos os elementos que a possuem.

HTML



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title> Nossa aula de CSS</title>
5   <link rel="stylesheet" type="text/css" href="estilo.css">
6 </head>
7 <body>
8
9   <h1 class="texto"> Boa noite </h1>
10
11   <p> Nossa aula de CSS </p>
12
13   <div class="texto"> Seletores </div>
14
15 </body>
16 </html>
```

CSS



```
1
2 .texto {
3   background-color:#F4A460;
4   color:#4B0082;
5   font-family:Comic Sans MS;
6   font-size:60px;
7 }
```

Visualização



O ELEMENTO DIV

O elemento <div> vem da palavra *divisão* a função da *div* é dividir trechos de códigos HTML. Ele pode ser utilizado para agrupar elementos para fins de estilos (usando class ou id).

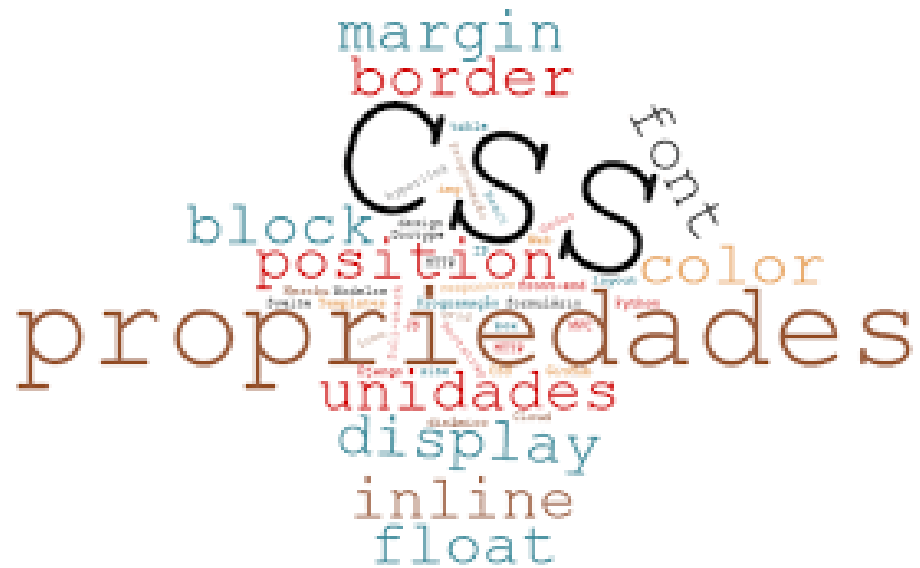
```
<div id="img"> </div>
```



É muito importante não utilizarmos o mesmo nome para os **IDs** e as **classes**. Desse modo, evitamos conflitos. Também não é aconselhável colocar **ID** e **class** no mesmo elemento. No entanto, isso é possível. Vejamos:

```
<p id='amigo1' class='amigo2'> Seu texto </p>
```

```
<p id='amigo3' class='amigo2'> Seu texto </p>
```



Algumas propriedades mais comuns

Backgrounds

Planos de fundo

Background-color



Define a cor do fundo de um elemento.

Background-image



Insere uma imagem como fundo do elemento.

Background-size



Configura o tamanho da imagem no elemento.

Background-position



Define a posição da imagem em relação à caixa.

Background-attachment



Define se a imagem de fundo rolará com a tela.

Background-repeat



Define se a imagem de fundo pode se repetir ou não.

Existem pelo menos oito propriedades CSS destinadas a estilizar o fundo de uma página HTML.

BACKGROUND-COLOR ➡ define a cor do fundo

BACKGROUND-IMAGE ➡ define uma imagem de fundo

BACKGROUND-REPEAT ➡ define a maneira como a imagem de fundo é posicionada

BACKGROUND-ATTACHMENT ➡ define se a imagem de fundo "rola" ou não com a tela

BACKGROUND-POSITION ➡ define como e onde a imagem de fundo é posicionada

BACKGROUND-CLIP ➡ define a área do box onde a imagem de fundo é aplicada

BACKGROUND-ORIGIN ➡ define a posição de origem da imagem no box

BACKGROUND-SIZE ➡ define as dimensões da imagem no box

BACKGROUND-COLOR

define a cor do fundo

Valores em hexadecimal: #ffc6d9

Valores em rgb: rgb(255,235,0)

Valores em hsl: hsl(210,100%,40%)

Valores em nome: red, blue, green...etc

Transparente: transparent

```
body {  
    background-color: #92a8d1;  
}
```

BACKGROUND-IMAGE

define uma imagem de fundo

URL: url(caminho/nomeimagem.jpg)

URL: url('caminho/nomeimagem.jpg')

URL: url("caminho/nomeimagem.gif")

Gradiente linear: linear-gradient(45deg, blue, red)

Gradiente radial: radial-gradient(20px 50px, green, blue, red)

```
body {  
    background-image: url("paper.gif");  
    background-color: #cccccc;  
}
```

*****O uso de aspas simples ou duplas no URL da imagem é facultativo*****

BACKGROUND-REPEAT

define a maneira como a imagem de fundo é posicionada

imagem não repete: **no-repeat**

imagem repete na vertical e horizontal: **repeat**

imagem repete na vertical: **repeat-y**

imagem repete na horizontal: **repeat-x**

imagem toca as quatro bordas internas espaçadas por igual: **space**

imagem toca as quatro bordas preenchendo o fundo : **round**

BACKGROUND-ATTACHMENT

define se a imagem de fundo "rola" ou não com a tela

imagem permanece fixa em relação à viewport (janela do navegador) e não rola juntamente com o conteúdo: **fixed**

imagem não permanece fixa em relação à viewport (janela do navegador) e rola juntamente com o conteúdo - este é o valor padrão: **scroll**

imagem permanece fixa em relação à viewport (janela do navegador) mas quando aplicada ao box de um elemento rola juntamente com o conteúdo se o elemento tiver um mecanismo de rolagem definido por overflow: scroll: **local**

BACKGROUND-POSITION

define como e onde a imagem de fundo é posicionada

canto superior esquerdo: **left top**

centro esquerdo: **left center**

inferior esquerdo: **left bottom**

bem no topo: **right top**

centro direito: **right center**

Inferior direito: **right bottom**

topo centr: **center top**

centro centro: **center center**

parte inferior central: **center bottom**

O primeiro valor é a posição horizontal e o segundo valor é a vertical. O canto superior esquerdo é 0% 0%. O canto inferior direito é 100% 100%. Se você especificar apenas um valor, o outro valor será de 50%. O valor padrão é: 0% 0% : **x% y%**

BACKGROUND-CLIP

define a área do box onde a imagem de fundo é aplicada

a imagem ocupa até a área das bordas do box: **border-box**

a imagem ocupa até a área de padding : **padding-box**

a imagem ocupa até a área de conteúdo do boxe não ocupa as áreas de padding e bordas se houver: **content-box**

BACKGROUND-ORIGIN

define a posição de origem da imagem no box

a imagem tem por origem o canto superior : **border-box**

a imagem tem por origem o canto superior esquerdo : **padding-box**

a imagem tem por origem o canto superior esquerdo do box sem considerar o padding e as bordas : **content-box**

BACKGROUND-SIZE

define as dimensões da imagem no box

a imagem terá as dimensões definidas por unidade de medida CSS: 150px 70px

a imagem terá suas dimensões originais - este é o valor padrão: auto

a imagem terá as dimensões de modo a que sua maior dimensão ocupe toda a extensão do box e a menor dimensão seja relação entre largura e altura seja preservado: contain

a imagem terá as dimensões de modo a que sua menor dimensão ocupe toda a extensão do box e a maior dimensão seja relação entre largura e altura seja preservado: cover

BORDAS

As propriedades para as bordas, definem as características das quatro bordas de um elemento HTML

border-width : Especifica a largura da borda

border-style : Especifica o estilo da borda

border-color: Especifica a cor da borda

initial : Define esta propriedade com seu valor padrão

Inherit: Herda esta propriedade de seu elemento pai

Propriedade de estilo de borda

none: nenhuma borda

hidden: equivalente a none

dotted: borda pontilhada

dashed: borda tracejada

solid: borda contínua

double: borda dupla

groove: borda entalhada

ridge: borda em relevo

inset: borda em baixo relevo

outset: borda em alto relevo

Propriedade de largura de borda

thin: borda fina

medium: borda média

thick: borda grossa

length: unidade de medida CSS (px, pt,cm)

MARGIN

margem

A propriedade **margin** define área de margem, ou seja, a quantidade de espaço ao redor do elemento

margin-top: define a margem superior;

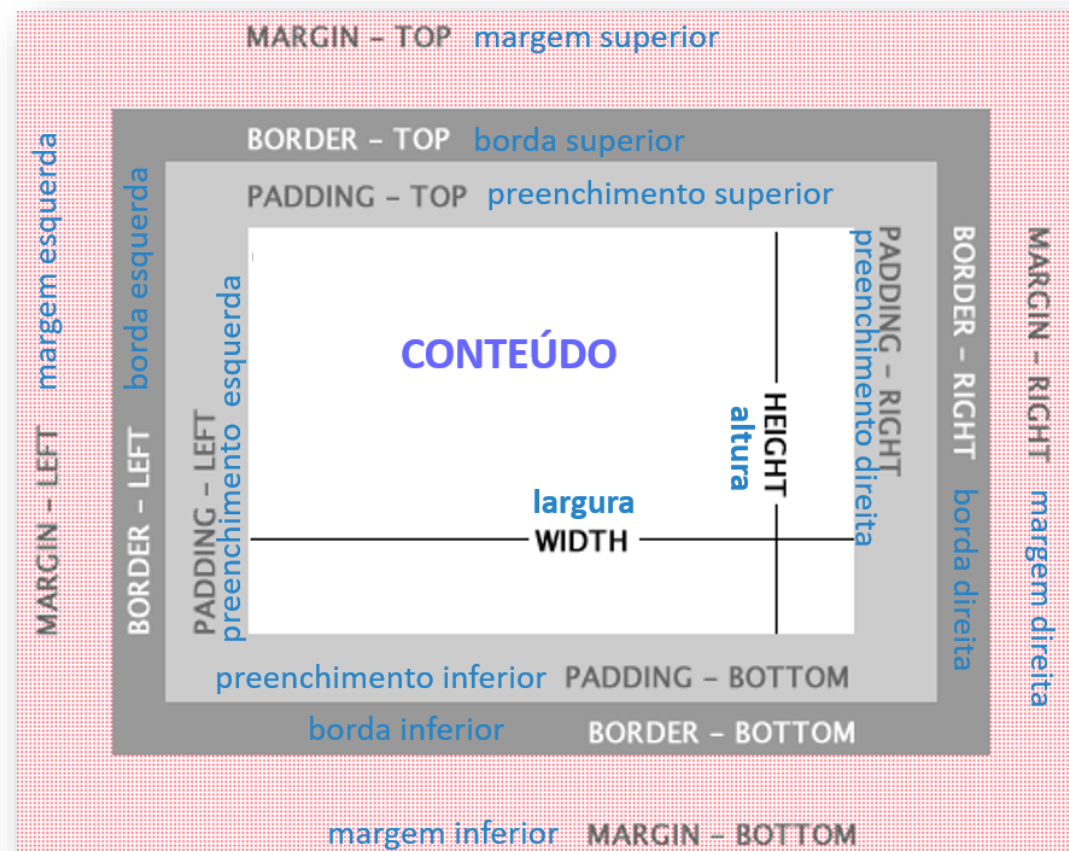
margin-right: define a margem direita;

margin-bottom: define a margem inferior;

margin-left: define a margem esquerda;

Margin: defini todas as quatro margens.

```
p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}
```



PADDING

preenchimento

O **padding** tem um funcionamento muito similar ao do **margin**, porém ao invés de dar uma espaçamento externo, ele dá um interno

padding-top: define o espaçamento superior;

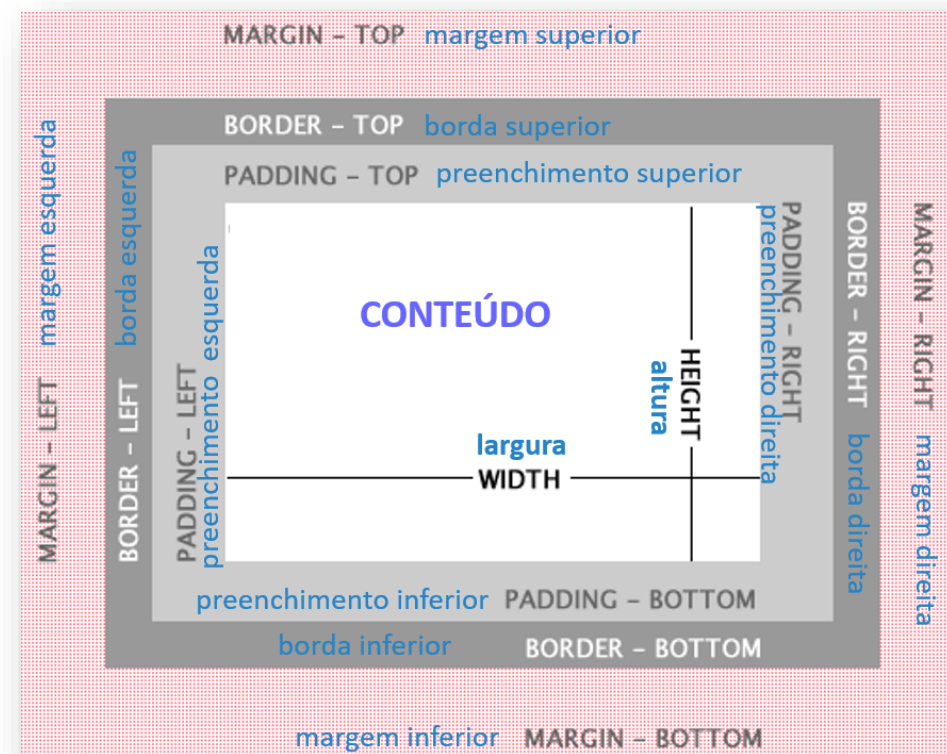
padding-right: define o espaçamento à direita;

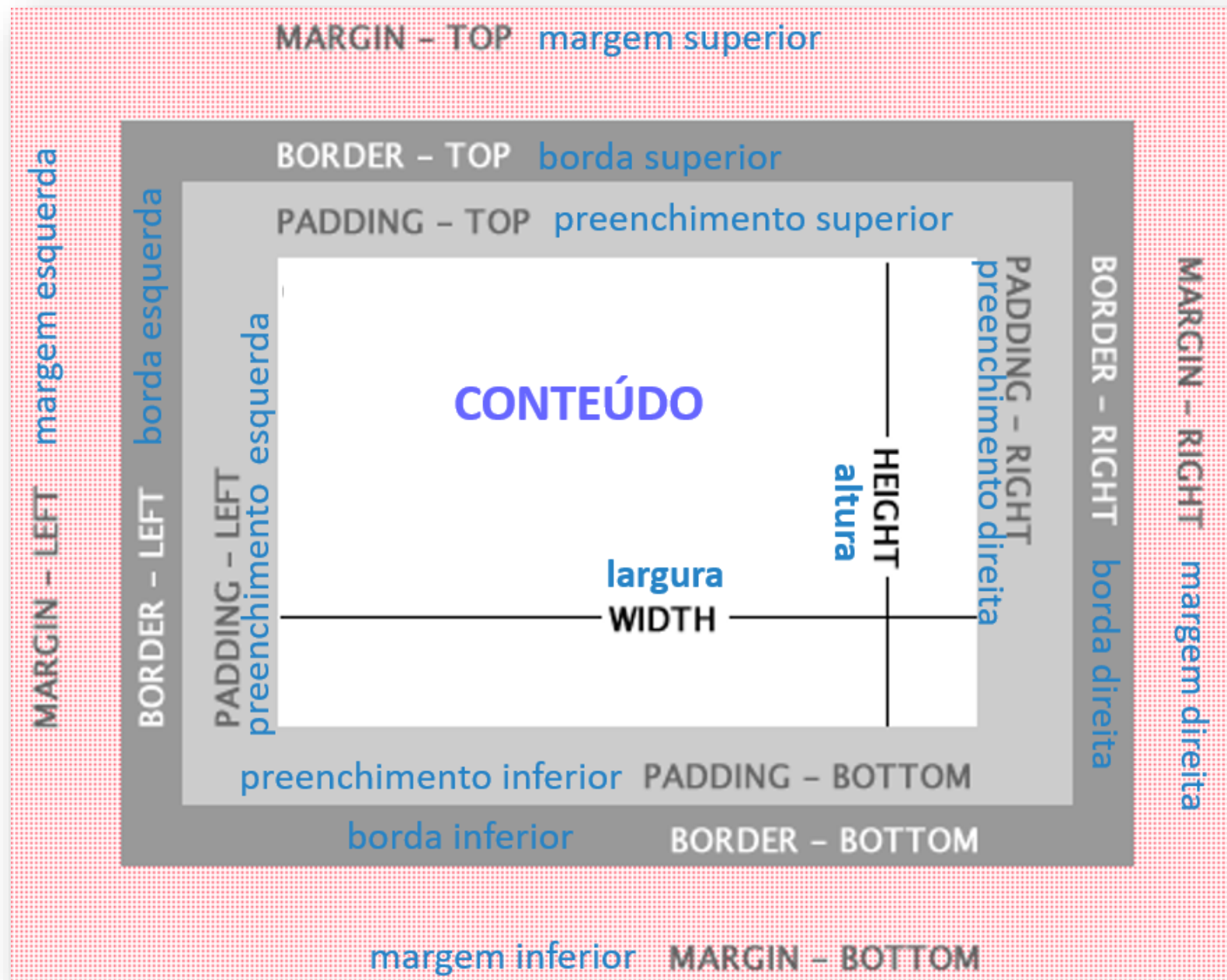
padding-bottom: define o espaçamento inferior;

padding-left: define o espaçamento à esquerda;

Padding: defini o espaçamentos de todos os lados.

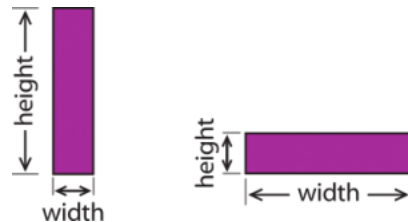
```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}
```





Height, Width and Max-width

Altura, largura e largura máxima



height e **width** são propriedades usadas para definir a altura e a largura de um elemento.
max-width é usada para definir a largura máxima de um elemento.

Valores das propriedades Height e Width

Auto: Este é o padrão o navegador calcula a altura e a largura

Length: Define a altura/largura em px, cm, etc.

Initial : Define a altura/largura para seu valor padrão

Inherit: A altura/largura será herdada de seu valor pai

```
div {  
    max-width: 500px;  
    height: 100px;  
    background-  
color: powderblue;  
}
```


Text

texto

Propriedades para estilizar textos

color: muda a cor do texto

font-family: define uma lista de fontes para ser utilizada no elemento

font-size: muda o tamanho do texto

font-style: muda o estilo da fonte

```
h1 {  
    color:rgb(38, 20, 82);  
    font-family: "Trebuchet MS", Arial, Tahoma, sans-serif;  
    font-size:40px;  
    font-style: italic;  
}
```

line-height: muda a altura de cada linha, ou entrelinha
word-spacing: determina o espaçamento entre palavras
letter-spacing: determina o espaçamento entre letras

```
h1 {  
  color:rgb(38, 20, 82);  
  font-family: "Trebuchet MS", Arial, Tahoma, sans-serif;  
  font-size:40px;  
  line-height: 1.5px;  
  word-spacing: 5px;  
  letter-spacing: 2px;  
}
```


Text Alignment

Alinhamento de Texto

text-align: defini o alinhamento : center, left, right ou justify

text-align-last: alinha a última linha de um texto: right ou justify e center

Links

Os links podem ser estilizados com qualquer propriedade CSS (por exemplo color, font-family, background

a:link: define o estilo do link no estado inicial;

a:visited: define o estilo do link visitado;

a:hover: define o estilo do link quando passa-se o mouse sobre ele;

a:active: define o estilo do link ativo (o que foi "clicado").

Exemplos:

```
a:link { /* link não visitado */  
  color: red;  
}
```

```
a:hover { /* passe o mouse sobre o link */  
  color: hotpink;  
}
```

```
a:visited { /* link visitado */  
  color: green;  
}
```

```
a:active { /* link selecionado */  
  color: blue;  
}
```

Formatação de Listas com Propriedades CSS

List

Já aprendemos a criar listas no HTML, certo? Agora, iremos aprender a modificar a aparência das listas. Com isso, poderemos tornar as listas visualmente mais atraentes.

Para modificar a aparência das listas, o CSS oferece uma propriedade exclusiva de formatação a ***list-style***. Tal propriedade é dividida em três características.

list-style-type: define o tipo do marcador da lista;

list-style-image: define uma imagem como marcador da lista;

list-style-position: define a posição do marcador de lista;

list-style: maneira abreviada para definir DUAS propriedades

List-style-type

O *list-style-type* é responsável por configurar o tipo de marcador que a lista irá utilizar.

```
ul.a {  
  list-style-type: circle;  
}  
  
ul.b {  
  list-style-type: square;  
}  
  
ol.c {  
  list-style-type: upper-roman;  
}  
  
ol.d {  
  list-style-type: lower-alpha;  
}
```

list-style-image

list-style-image especifica uma imagem como o marcador de item da lista

```
ul {  
  list-style-image: url('imagem.jpg');  
}
```

list-style-position

list-style-position especifica a posição dos marcadores de item de lista. Para essa propriedade existem dois valores :

outside: marcador fora do alinhamento do texto

inside: marcador alinhado com texto

```
ul.a {  
    list-style-position: outside;  
}
```

```
ul.b {  
    list-style-position: inside;  
}
```

TABELAS

Para tabelas existem algumas propriedades.

Bordas: especificar as bordas de uma tabela

border-width: largura

border-style: estilo da linha

border-color: cor

border: tudo junto – largura estilo cor

Alinhamento: propriedade define o alinhamento de uma tabela

text-align: alinhamento horizontal

vertical-align: alinhamento vertical

```
td {  
    text-align:right;  
    vertical-align:bottom;  
}
```