

# Módulo II - CSS

**Aqui é um ambiente seguro;**

**Não existe pergunta boba;**

**Pausas são importante;**

**Somos um coletivo;**

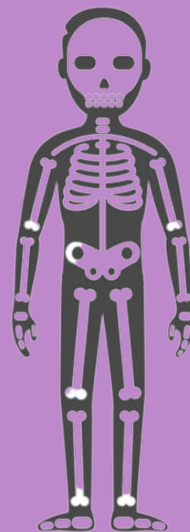
**Bebam água;**

**Feedbacks são tudo de bom!**

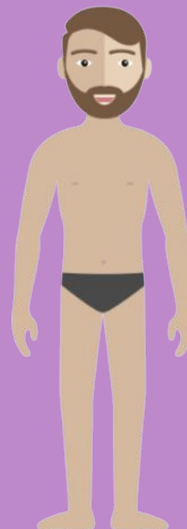


# O que é CSS?

**CSS** (**C**ascading **S**tyle **S**heets) é uma linguagem de folha de estilo. Como o **HTML** é usado para estruturar um documento da web(esqueleto), o **CSS** estiliza o seu documento(corpo) - layouts de página, cores e fontes são todos personalizados com **CSS**. Pense no **HTML** como a base (cada casa tem um) e no **CSS** como as escolhas estéticas.



HTML



CSS

# Conexão com o HTML

Podemos conectar o CSS com o HTML de três formas:

- Inline
- Interno

```
15      <body>
16      |   <h1>Começando com CSS</h1>
17      |   <h1 style="color: darkblue;"> Estamos implementando o CSS inline</h1>
18      | </body>
```

Inline

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <style>
5        body {
6          background-color: rgba(162, 85, 213, 0.305);
7        }
8        h1 {
9          color: rgb(8, 47, 62);
10         padding: 60px;
11        }
12      </style>
13    </head>
14
15    <body>
16      <h1>Começando com CSS</h1>
17    </body>
18  </html>
```

Interno

# Conexão com o HTML

Externo:

# style.css U X

pretalab > # style.css > ...

```
1  body {  
2    background-color: rgba(162, 85, 213, 0.305);  
3  }  
4  h1 {  
5    color: rgb(8, 47, 62);  
6    padding: 60px;  
7  }
```

```
<head>  
  <link rel="stylesheet" type="text/css" href="style.css"/>  
</head>
```

## Sintaxe:

```
elemento{  
    propriedade: valor;  
}
```

O elemento pode ser representado por um seletor(tag), classe ou id.

# Seletores:

p

{color: blue}

↓

↓

**seletor**{

**propriedade: valor;**

}

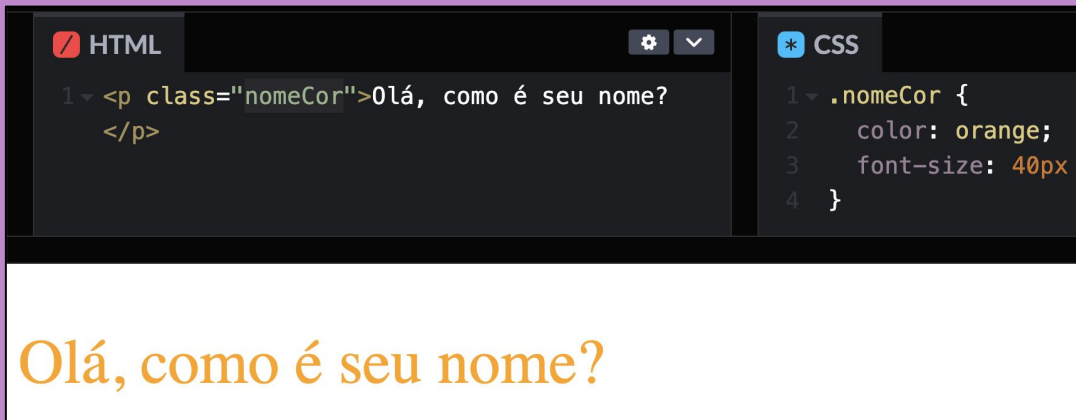
O seletor seleciona o elemento do HTML que será estilizado.

# classes:

```
.classe{  
  propriedade: valor;  
}
```

É possível selecionar elementos por meio de uma classe definida como atributo em uma tag.

No CSS, a sintaxe é o caractere . (ponto) seguido do nome da classe.





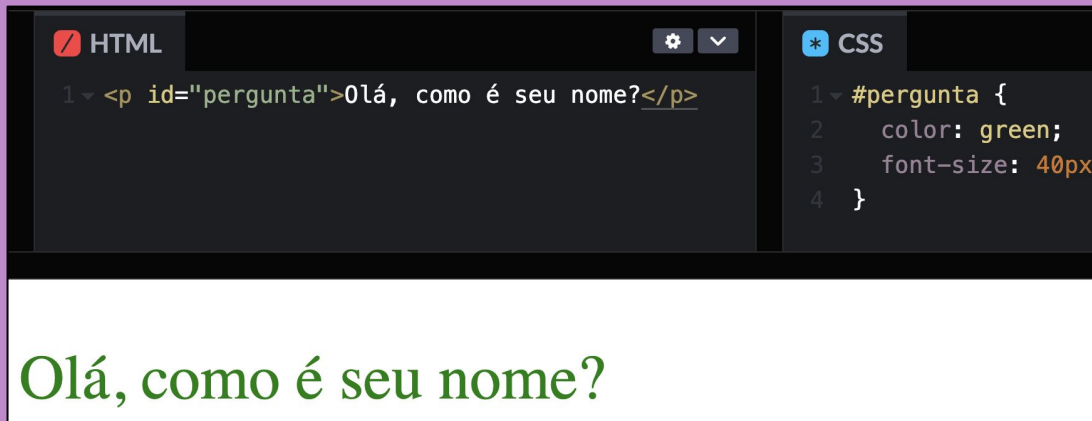
# ids:

#id{

**propriedade:** valor;

}

O ID, como o nome já diz, é uma identificação única: podendo ser utilizada apenas uma vez no documento inteiro. Normalmente é utilizada para identificar elementos estruturais da página.



# Esquema de Cores

## Nominal

`blue`

## Hexadecimal

`#FFFFFF`

## RGB

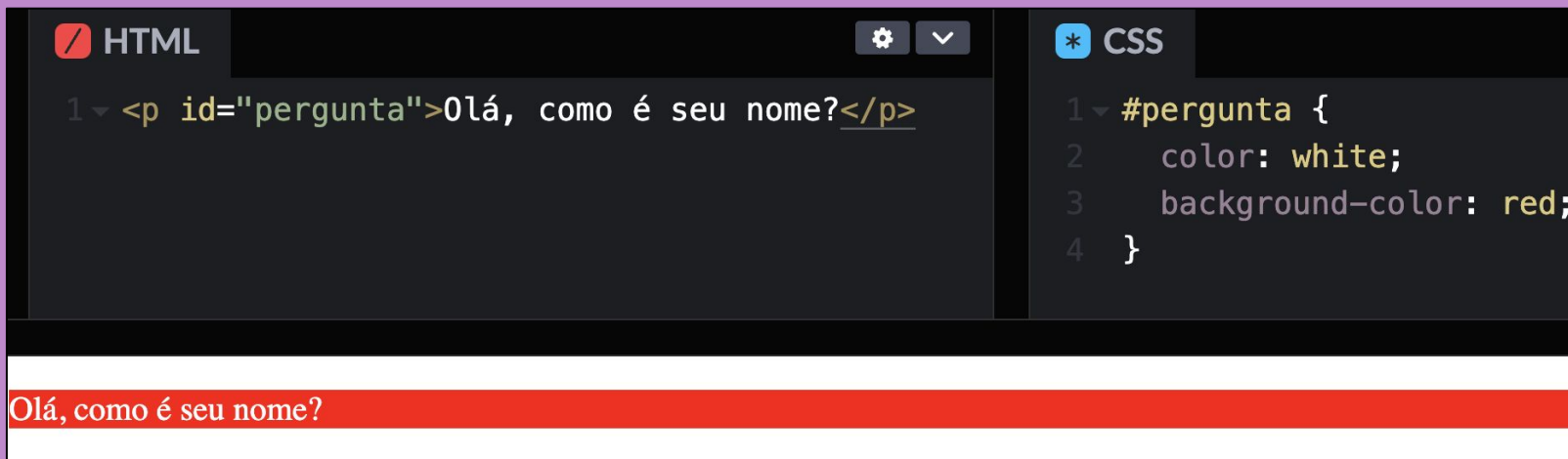
`rgb(255,255,255)`

O CSS trabalha com um esquema de cores possibilitando usar cores nominais, hexadecimais ou em rgb.

# Cores

A propriedade **background-color** define a cor de fundo de um elemento.

A propriedade **color** define a cor da fonte

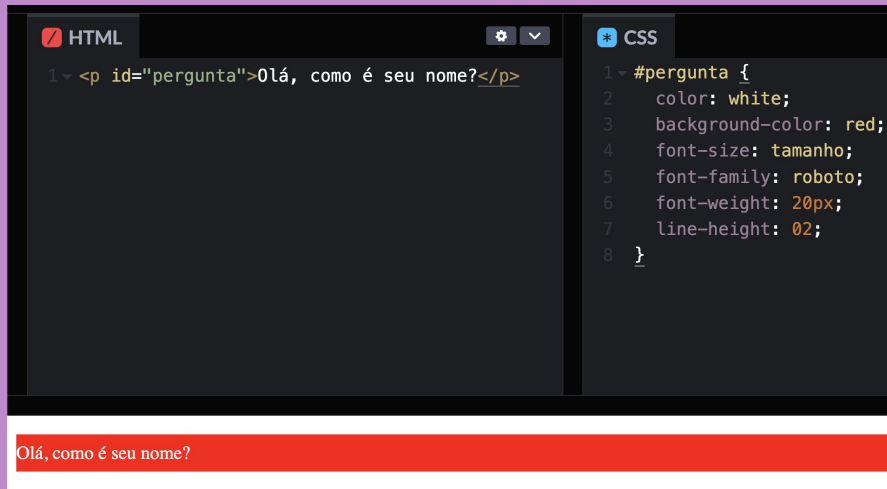


```
HTML
1 <p id="pergunta">Olá, como é seu nome?</p>

CSS
1 #pergunta {
2   color: white;
3   background-color: red;
4 }
```

Olá, como é seu nome?

# Fontes



The image shows a code editor with two panels. The left panel, labeled 'HTML', contains the following code:

```
1 <p id="pergunta">Olá, como é seu nome?</p>
```

The right panel, labeled 'CSS', contains the following code:

```
1 #pergunta {  
2   color: white;  
3   background-color: red;  
4   font-size: tamanho;  
5   font-family: roboto;  
6   font-weight: 20px;  
7   line-height: 02;  
8 }
```

Below the code editor, a preview of the rendered HTML is shown. It consists of a red rectangular box containing the text 'Olá, como é seu nome?' in white font.

Com o CSS é possível categorizar as fontes de diversas formas:

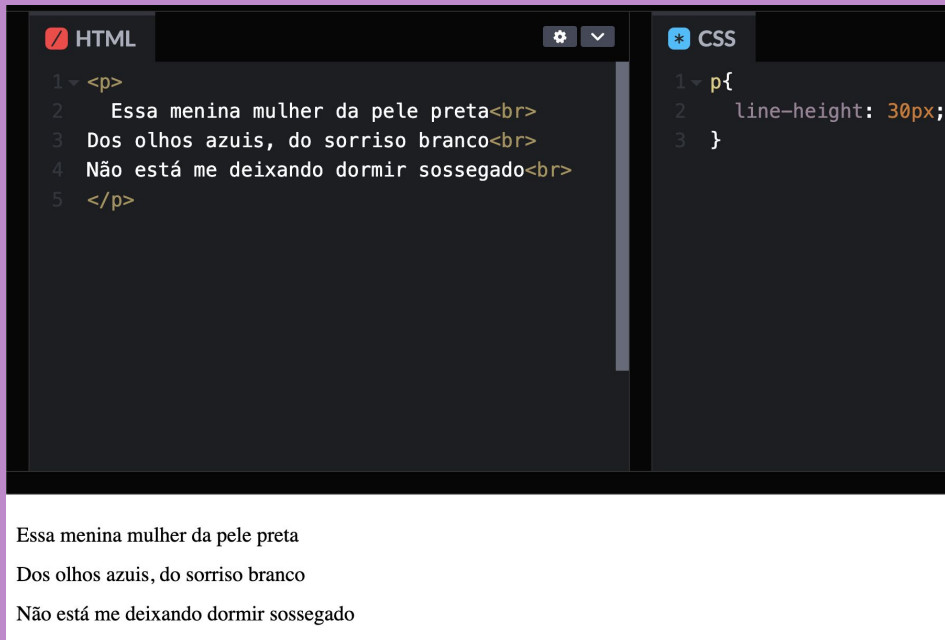
**font-size:** tamanho

**font-family:** tipo de fonte

**font-weight:** peso ou a intensidade da fonte

**line-height:** define a distância entre as linhas de texto

# Espaçamento



The image shows a code editor with two panels. The left panel is labeled 'HTML' and contains the following code:

```
1 <p>
2   Essa menina mulher da pele preta<br>
3   Dos olhos azuis, do sorriso branco<br>
4   Não está me deixando dormir sossegado<br>
5 </p>
```

The right panel is labeled 'CSS' and contains the following code:

```
1 p{
2   line-height: 30px;
3 }
```

Below the code editor, the rendered output is shown on a white background. It displays three lines of text, each on a new line, with a significant vertical gap between them, demonstrating the effect of the `line-height` property.

Essa menina mulher da pele preta  
Dos olhos azuis, do sorriso branco  
Não está me deixando dormir sossegado

A propriedade **line-height** é utilizada para definir a distância entre as linhas de texto.

# Bordas

A propriedade **border** é utilizada para definir a distância entre as linhas de texto.

Podemos definir as bordas de acordo com as direções e necessidades

**border**: abreviação referente as quatro bordas;

**border-top**: propriedades da borda superior;

**border-bottom**: define propriedades da borda inferior.

**border-right**: propriedades da borda direita;

**border-left**: propriedades da borda esquerda;

**border-style**: define o estilo de borda dos elementos

**border-width**: define a largura do elemento

**border-color**: define cores visíveis.

[Saiba mais aqui!](#)

# Bordas

```
HTML
1 <p>
2   Essa menina mulher da pele preta<br>
3   Dos olhos azuis, do sorriso branco<br>
4   Não está me deixando dormir sossegado<br>
5 </p>

CSS
1 p{
2   line-height: 30px;
3   border-width: 5px;
4   border-style: solid;
5   border-color: red;
6 }
```

Essa menina mulher da pele preta

Dos olhos azuis, do sorriso branco

Não está me deixando dormir sossegado

[Saiba mais aqui!](#)

# Precedência CSS

A ordem de precedência dos cascading style sheets é esta:  
atributos de estilos inline têm precedência sobre tags de style inseridas  
tags de style inseridas por sua vez têm precedência sobre style sheets por link

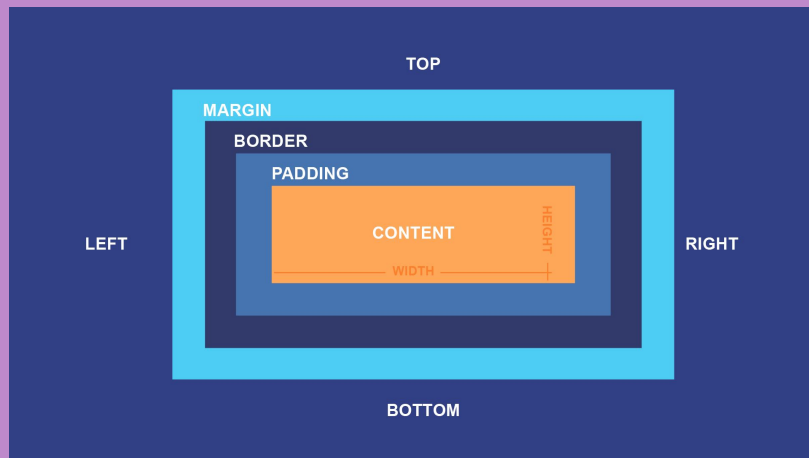
- Seletores Universais
- Tipo de Seletores
- Classes seletoras
- Seletores ID
- Estilo Inline

[Saiba mais!](#)

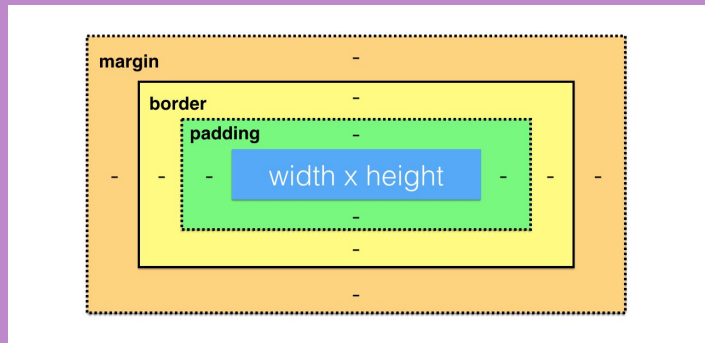


# Box model

É o modelo de estilização em caixa com diversas propriedades que facilitam a vida do desenvolvedor.

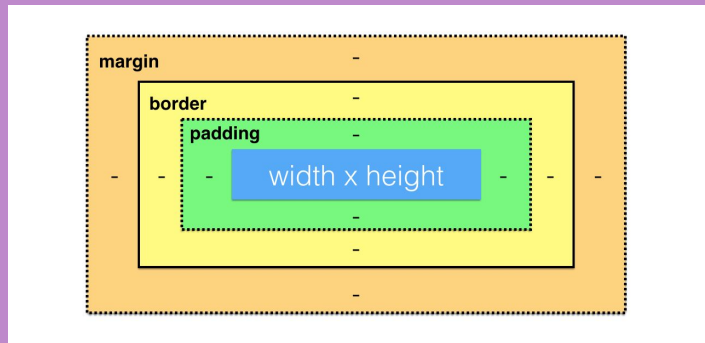


# Largura e altura



Usamos as propriedades **width** e **height** para definir as dimensões. Elas podem assumir valores em comprimento (como **px**, **mm**, etc.), **porcentagem** (como 40%) ou nenhum (o valor padrão). A propriedade **min-width** e **min-height** é usada para definir a largura e altura mínimas do elemento. A propriedade **max-width** e **max-height** é usada para definir a largura e altura máximas do elemento. Quando o valor é definido como nenhum, significa que não há largura máxima e altura máxima definidas para o elemento.

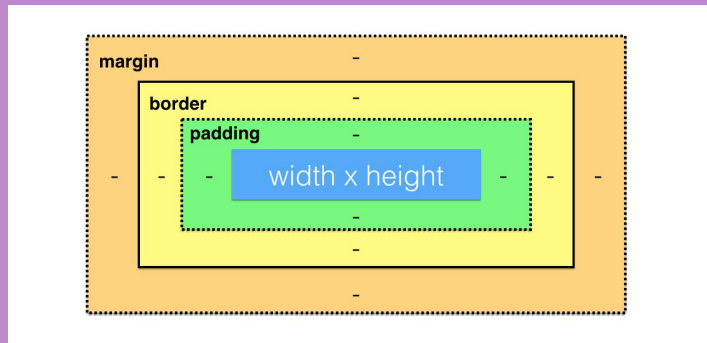
# Preenchimento - padding



A propriedade padding define uma a distância entre o conteúdo de um elemento e suas bordas:

**padding:** define o espaçamento nos 4 lados;  
**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;

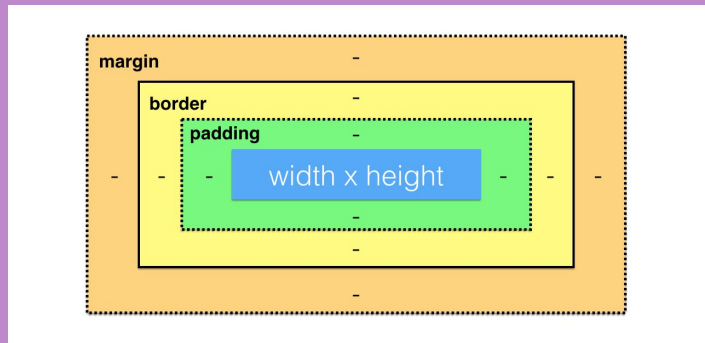
# Preenchimento - margin



Na imagem temos a parte laranja que é a margem da parte azul.

Podemos definir o valor da margem com a propriedade `margin` em comprimento, porcentagem, automático e herança.

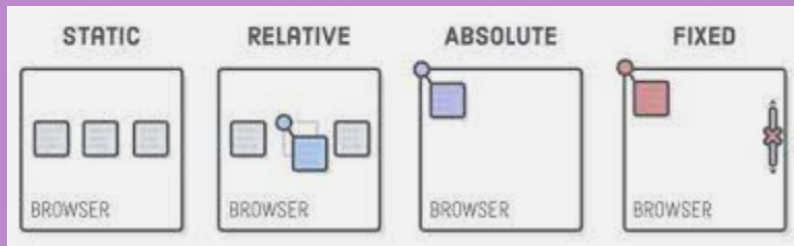
# Tamanho do box



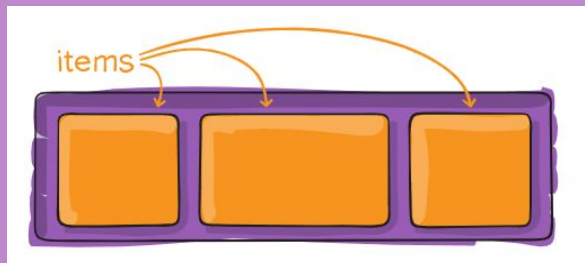
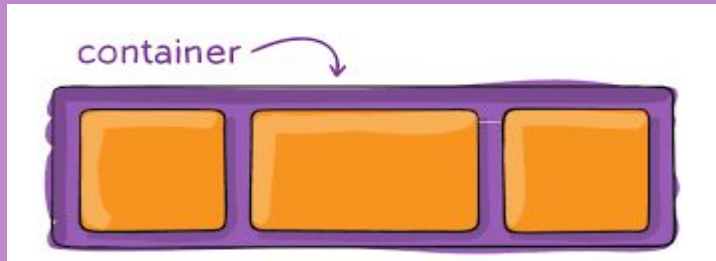
A propriedade `box-sizing` é utilizada para alterar a propriedade padrão da box model, usada para calcular larguras (widths) e alturas (heights) dos elementos.

# Posicionamento

A propriedade **position** recebe os valores **static**, **relative**, **absolute** e **fixed** para posicionar elementos.



# display



Define a organização e o tipo de caixa que vai ser renderizada em uma página web de forma organizada. inline ou bloco dependendo do valor fornecido. Ele permite um contexto flexível para todos os seus filhos diretos. O display trabalha como uma tabela, onde o seu conteúdo pode ser colocado em linhas e colunas

# Float

determina que um item seja colocado ao longo do lado direito ou esquerdo do seu container, onde textos e elementos em linha irão se posicionar ao seu redor.



# Clear

Em conjunto com o float, a propriedade clear é usada quando queremos que o próximo elemento fique abaixo e não ao lado como ficaria se fosse só com o float

# Clear-fix

Caso algum dos elementos dentro do container seja mais alto que o outro, com a propriedade clearfix conseguimos igualar o espaçamento entre elas:

## Sem Clearfix

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...



## Com Clearfix

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum...



# Clear-fix

Caso algum dos elementos dentro do container seja mais alto que o outro, com a propriedade clearfix conseguimos igualar o espaçamento entre elas:

**E isso é tudo...**

