



# Enxurrada de Bits

CSS

# O que é CSS?

- É a sigla para **Cascading Style Sheets** (Folhas de estilo em cascatas).
- É uma linguagem para especificar como os elementos de um site serão apresentados para o usuário, através de regras.
- Tem a responsabilidade de alterar a aparência de páginas web.

# Definindo o estilo

- Adicionamos um novo elemento: `<style>...</style>`
- Podemos colocá-lo no `<head>` ou no `<body>`, mas a **melhor prática é no cabeçalho**

```
<!DOCTYPE html>
<html>
  <head>
    <title>Este é o título que aparece na aba</title>
    <style>
      /* Escrever o código CSS aqui dentro */
    </style>
  </head>
</html>
```

# Utilizando CSS em arquivos externos .css

- A prática mais correta
- Permite uma melhor divisão do código
- Velocidade de carregamento mais rápida
- O mesmo arquivo .css pode ser usado para várias páginas

Considerando que o arquivo seja style.css:

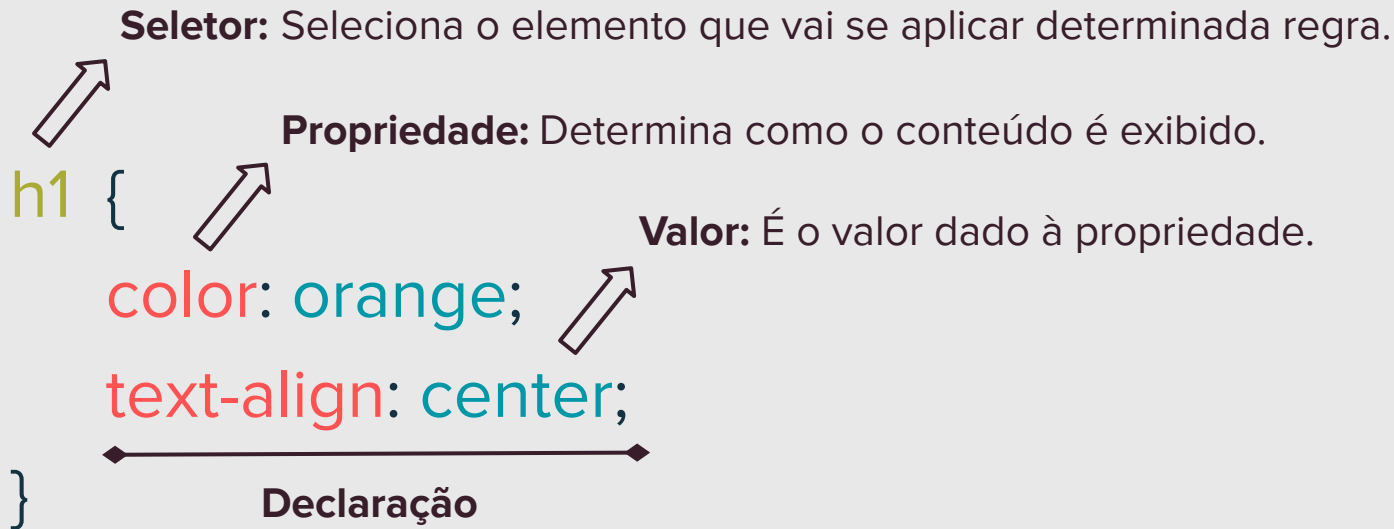
```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Este é o título que aparece na aba</title>
  <link rel="stylesheet" href="style.css">
</head>
</html>
```

# Utilizando CSS em arquivos externos .css

E, no arquivo .css, escrevemos o conteúdo que antes era colocado na tag <style>:

```
h1 {  
    font-family: Norican,cursive;  
    background-color: □black;  
    color: ■white;  
    text-align: center;  
    font-size: 400%;  
    border-style: solid;  
    border-color: ■red;  
}  
  
p {  
    font-family: Pacifico,cursive;  
    font-size: 150%;  
    color: ■rgb(0,255,0)  
}
```


# Anatomia de uma regra CSS



# Propriedades

**background-color** (cor de fundo)

Define a cor do fundo

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

**color** (cor)

Define a cor do texto

```
body {  
  color: white;  
}
```

# Propriedades

## **font-family** (fonte)

Define a fonte a ser usada no texto

```
h1 {  
  font-family: Arial;  
}
```

## **font-size** (tamanho)

Define o tamanho das letras

```
p {  
  font-size: 32px;  
}
```



# Propriedades

## **text-align** (alinhamento)

```
p {  
  text-align: center;  
}
```

A propriedade CSS `text-align` descreve como conteúdo inline, como texto, é alinhado no elemento pai em bloco. `text-align` não controla o alinhamento de elementos em bloco, apenas o seu conteúdo inline.

```
p {  
  text-align: justify;  
}
```

A propriedade CSS `text-align` descreve como conteúdo inline, como texto, é alinhado no elemento pai em bloco. `text-align` não controla o alinhamento de elementos em bloco, apenas o seu conteúdo inline.

```
p {  
  text-align: right;  
}
```

A propriedade CSS `text-align` descreve como conteúdo inline, como texto, é alinhado no elemento pai em bloco. `text-align` não controla o alinhamento de elementos em bloco, apenas o seu conteúdo inline.

```
p {  
  text-align: left;  
}
```

A propriedade CSS `text-align` descreve como conteúdo inline, como texto, é alinhado no elemento pai em bloco. `text-align` não controla o alinhamento de elementos em bloco, apenas o seu conteúdo inline.

# Propriedades

**letter-spacing** (espaçamento entre letras)

Define o espaçamento entre os caracteres de um bloco de texto.

```
body {  
  letter-spacing: 10px;  
}
```

**word-spacing** (espaçamento entre palavras)

Define o espaçamento entre as palavras de um bloco de texto.

```
body {  
  word-spacing: 10px;  
}
```

# Propriedades

**background-image** (imagem de fundo)

Define uma imagem de plano de fundo.

```
body {  
  background-image: url(http://www.example.com/imagem.png);  
}
```

**float** (ponto flutuante)

Determina que um elemento deve ser retirado do seu fluxo normal e colocado ao lado direito (right) ou esquerdo (left) do seu local, onde outros elementos em linha irão se posicionar ao seu redor.

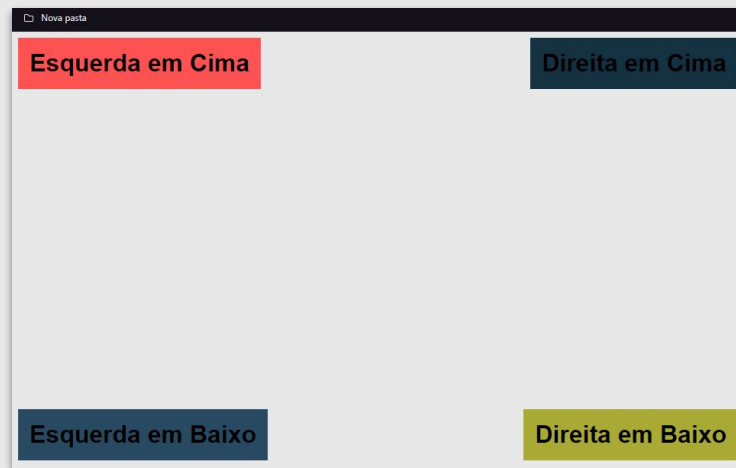
```
body {  
  float: right;  
}
```

# Como funciona: float

## Sem o float



## Com o float



Exemplo prático: <https://fegemo.github.io/cefet-front-end/classes/html4/#26>

# Propriedades

## **width** (largura)

Define a largura do elemento na página

```
h1 {  
  width: 100px; /* tamanho fixo, absoluto da tela */  
  width: 2em; /* ajustado de acordo com o tamanho da fonte */  
  width: 120%; /* ajustado em relação à largura do elemento pai */  
}
```

## **height** (altura)

Define a altura do elemento na página

```
p {  
  height: 100px; /* tamanho fixo, absoluto da tela */  
  height: 2em; /* ajustado de acordo com o tamanho da fonte */  
  height: 120%; /* ajustado em relação à largura do elemento pai */  
}
```

# Propriedades

## **padding** (preenchimento)

Define o espaçamento entre o conteúdo e a borda do elemento

```
p {  
  padding: 30px;  
}
```

## **margin** (margem)

Define as margens entre o elemento e as coisas fora dele

```
body {  
  margin-left: 20%;  
  margin-right: 20%;  
  margin-top: 20%;  
  margin-bottom: 20%;  
}
```

# Propriedades

## **border** (borda)

Define a borda do elemento

```
p {  
  border: 3px solid pink;  
}
```

1º argumento: espessura da borda

2º argumento: tipo de linha da borda

3º argumento: cor da borda

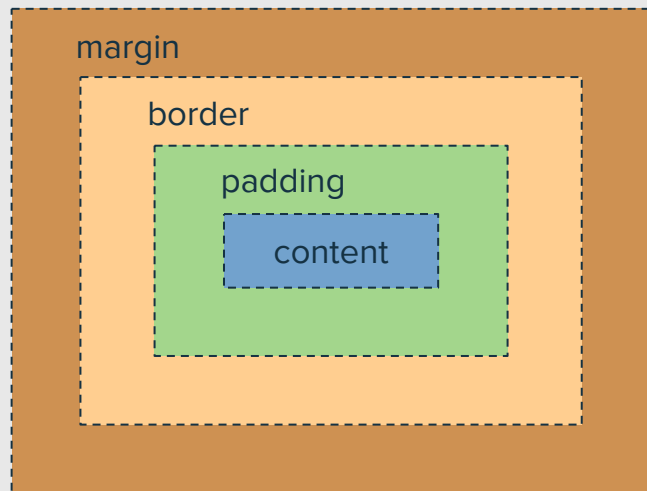
Exemplos de estilos de borda (2º argumento):

**solid**

**dotted**

**dashed**

# Box Model



espaço do conteúdo



espaço de preenchimento



bordas



espaço externo



# Propriedades: box-sizing (altera o significado da width e height de um elemento)

- box-sizing: **content-box** (valor padrão)

width = largura do conteúdo

```
<div class="produto">  
    
  <span class="nome">caixinha</span>  
</div>
```

```
.produto {  
  width: 50px;  
  padding: 10px;  
  border: 2px dashed #66aa55  
}
```



74px de largura = 50px do elemento + 20px padding + 4px border

# Propriedades: box-sizing (altera o significado da width e height de um elemento)

- box-sizing: **border-box**

width = largura do [conteúdo + padding + border]

Mais intuitivo 👍

```
<div class="produto">
  
  <span class="nome">caixinha</span>
</div>
```

```
.produto {
  width: 50px;
  padding: 10px;
  border: 2px dashed #66aa55;
  box-sizing: border-box;
}
```



50px de largura = 20px padding + 4px border + 26px que sobra pro elemento

Dúvidas?

# Referências

Material da disciplina de Programação Web do prof. Flávio Coutinho:

<https://fegemo.github.io/cefet-web/classes/css1/>

<https://fegemo.github.io/cefet-web/classes/css2/>

Documentação da Mozilla Developer Network:

<https://developer.mozilla.org/pt-BR/docs/Glossario/CSS>

[https://developer.mozilla.org/pt-BR/docs/Web/CSS/Shorthand\\_properties](https://developer.mozilla.org/pt-BR/docs/Web/CSS/Shorthand_properties)

# Leiam

Propriedade background:

<https://fegemo.github.io/cefet-web/classes/css2/#propriedades-background>

<https://developer.mozilla.org/pt-BR/docs/Web/CSS/background>

overflow:

<https://developer.mozilla.org/en-US/docs/Web/CSS/overflow>

# Colocando em prática

Crie um arquivo html (.html) como o que vimos anteriormente, e faça uma página web como esta abaixo utilizando um arquivo CSS externo (.css):

## Dicas:

- Utilize diferentes tags de título para cada canto
- Use o *font-size* para adequar o tamanho da letra
- A fonte é Arial
- Coloque as cores, de acordo com a figura
- Utilize a tag `<br>` para separar os de cima dos de baixo
- Lembre-se da propriedade *float* para colocar lado a lado.

