

# CURSO TÉCNICO EM INFORMÁTICA



**HTML/CSS**  
**AULA03**



## O QUE É CSS?

**CSS** é a sigla para **Cascading Style Sheets**, que em português significa Folhas de Estilo em Cascata.

O **CSS** é uma linguagem usada para descrever a aparência dos elementos HTML. Ou seja, enquanto o HTML organiza o conteúdo, o CSS define como esse conteúdo será apresentado – como cores, fontes, tamanhos, espaçamentos, e até animações!

### Principais funções do CSS:

- Controlar a aparência dos elementos (como cor, fonte, borda, margem, fundo, etc.).
- Definir o **layout das páginas** (como posicionamento e distribuição dos blocos).
- Adaptar o site para diferentes dispositivos (como telas de celular, tablets e impressoras).



# CURSO TÉCNICO EM INFORMÁTICA

## POR QUE USAR CSS?

- **Economiza trabalho:** Com CSS, você pode alterar o estilo de várias páginas ao mesmo tempo, usando apenas um arquivo.
- **Organiza o design:** Separa o conteúdo (HTML) da apresentação (css).
- **Facilita a manutenção:** Para mudar a aparência do site, você só precisa editar o arquivo CSS, sem tocar no HTML.

Como o CSS pode ser usado?

Existem três maneiras de aplicar CSS:

### 1. CSS Inline (dentro do próprio elemento):

```
<P STYLE = "COLOR:BLUE;">TEXTO AZUL</P>
```

### 2. CSS Interno (dentro da página HTML, usando <style>): →

```
<head>
  <style>
    p {
      color: red;
    }
  </style>
</head>
```



# CURSO TÉCNICO EM INFORMÁTICA

## 3. CSS Externo (em um arquivo separado **.css**):

```
<link rel="stylesheet" href="estilos.css">
```

No arquivo **estilos.css**, você teria algo como:

```
p {  
    color: green;  
}
```



The screenshot shows a code editor interface with two tabs: "index.html" and "estilo.css".

**index.html:**

```
1  <!DOCTYPE html>  
2  <html lang="en">  
3  <head>  
4      <meta charset="UTF-8">  
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">  
6      <title>Document</title>  
7      <link rel="stylesheet" href="estilo.css">  
8  </head>  
9  <body>  
10     <p>Este é o conteúdo principal do artigo.</p>  
11  </body>  
12 </html>
```

**estilo.css:**

```
p {  
    color: green;  
}
```

Este é o conteúdo principal do artigo.



# CURSO TÉCNICO EM INFORMÁTICA

## Vantagem do CSS Externo:

Se você quiser mudar a cor de todos os parágrafos do seu site, basta editar o arquivo CSS – não precisa alterar página por página!

## Exemplo de HTML com CSS Externo:

### Arquivo **index.html**:

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="estilos.css">
</head>
<body>

<h1>Bem-vindo!</h1>
<p>Este é um exemplo de página usando CSS externo.</p>

</body>
</html>
```

### Arquivo **estilo.css**:

```
h1 {
  color: darkblue;
  text-align: center;
}

p {
  color: gray;
  font-size: 18px;
}
```



## Sintaxe do CSS

**CSS Syntax (ou sintaxe do CSS)** é o conjunto de regras que define como escrever os estilos para modificar a aparência dos elementos HTML.

Uma regra básica de CSS é formada por um seletor e um bloco de declarações.

### A estrutura é assim

```
seletor {  
    propriedade: valor;  
}
```

#### Entendendo cada parte:

##### Seletor:

- Indica qual elemento HTML será estilizado (exemplo: p, h1, div).

##### Propriedade:

- Define o que você quer mudar (como cor, tamanho, fonte, espaçamento).

##### Valor:

- Define o novo valor para a propriedade (por exemplo, azul, **20px**, Arial).

##### Bloco de declarações:

- As propriedades e valores ficam entre chaves {}, e cada declaração termina com ponto e vírgula ;.



## Sintaxe do CSS

```
p {  
    color: blue;  
    font-size: 16px;  
}
```

Explicação:

- O seletor p indica que queremos aplicar estilos a todos os parágrafos (**<p>**).
- Dentro do bloco:
  - **color: blue;** muda a cor do texto para azul.
  - **font-size: 16px;** muda o tamanho da fonte para 16 pixels.



## Sintaxe do CSS

```
h1 {  
    text-align: center;  
    color: darkred;  
}
```

Explicação:

- O seletor **h1** aplica as regras a todos os títulos **<h1>**.
- **text-align: center;** centraliza o texto.
- **color: darkred;** muda a cor do título para vermelho-escuro.



# CURSO TÉCNICO EM INFORMÁTICA

## Sintaxe do CSS

### Resumo da Sintaxe CSS:

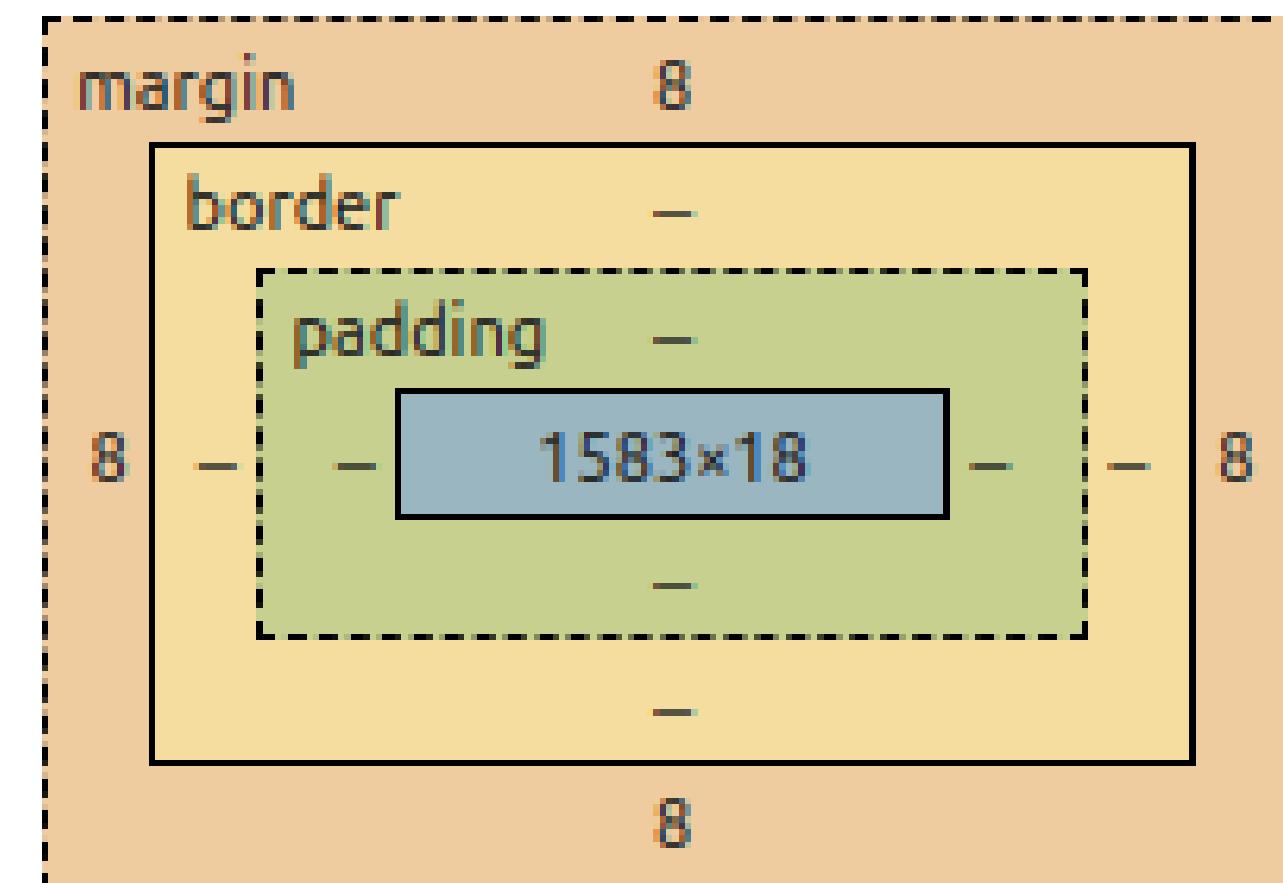
Parte	Função	Exemplo
Seletor	Escolhe o elemento HTML	p , h1 , .classe , #id
Propriedade	Define o que mudar	color , font-size , background-color
Valor	Define o novo estilo	blue , 18px , center
Bloco de Declarações	Agrupa propriedades e valores	{ propriedade: valor; }



## Border, Margin e Padding no CSS

Esses três conceitos são fundamentais no CSS porque eles controlam o **espaço ao redor dos elementos** HTML, como caixas.

Pense em cada elemento como uma caixa composta por 4 camadas:



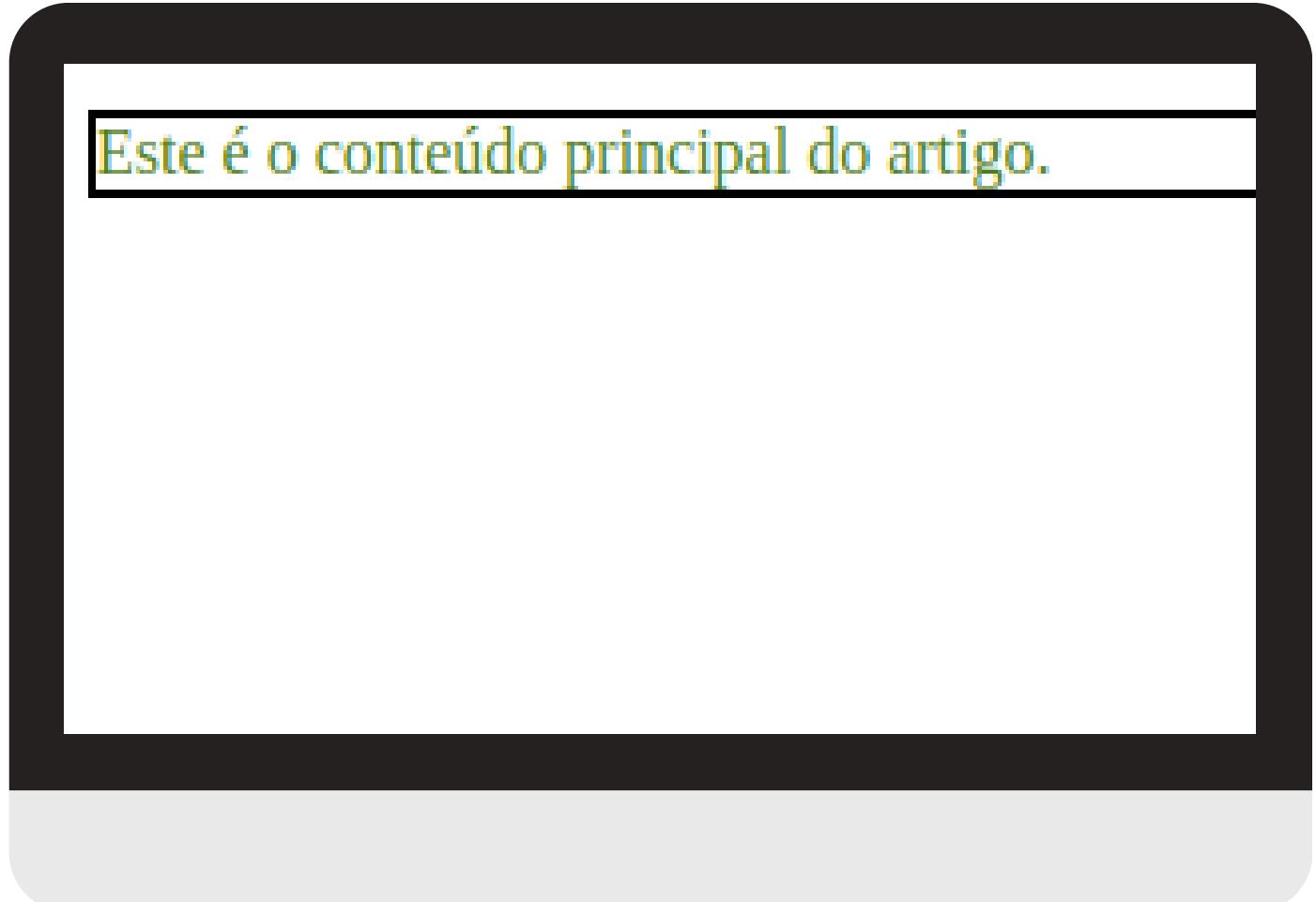
## Border, Margin e Padding no CSS

### Border (Borda)

- A **border** é a linha que envolve o elemento.
- Ela delimita visualmente o espaço ocupado pelo conteúdo e pelo **padding**.

### Arquivo estilo.css

```
P {  
    color: green;  
    border: 2px solid black;  
}
```



### Explicação:

- **2px** → espessura da borda.
- **solid** → estilo da borda (pode ser solid, dashed, dotted, etc.).
- **black** → cor da borda.

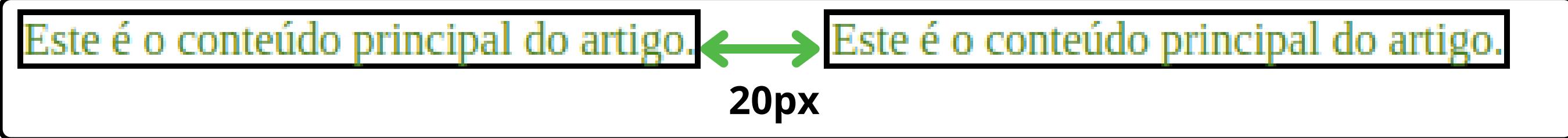


## Border, Margin e Padding no CSS

### Margin (Margem)

- A **margin** é o espaço externo entre o elemento e o que estiver fora dele (outros elementos).
- **Ela afasta o elemento dos seus vizinhos.**

### EXEMPLO:



estilo.css > ...

```
1 p {  
2   color: green;  
3   display: inline;  
4   border: 2px solid black;  
5   margin: 20px;  
6 }
```

### EXPLICAÇÃO:

20px → adiciona 20 pixels de espaço fora da borda em todas as direções (topo, direita, baixo e esquerda).



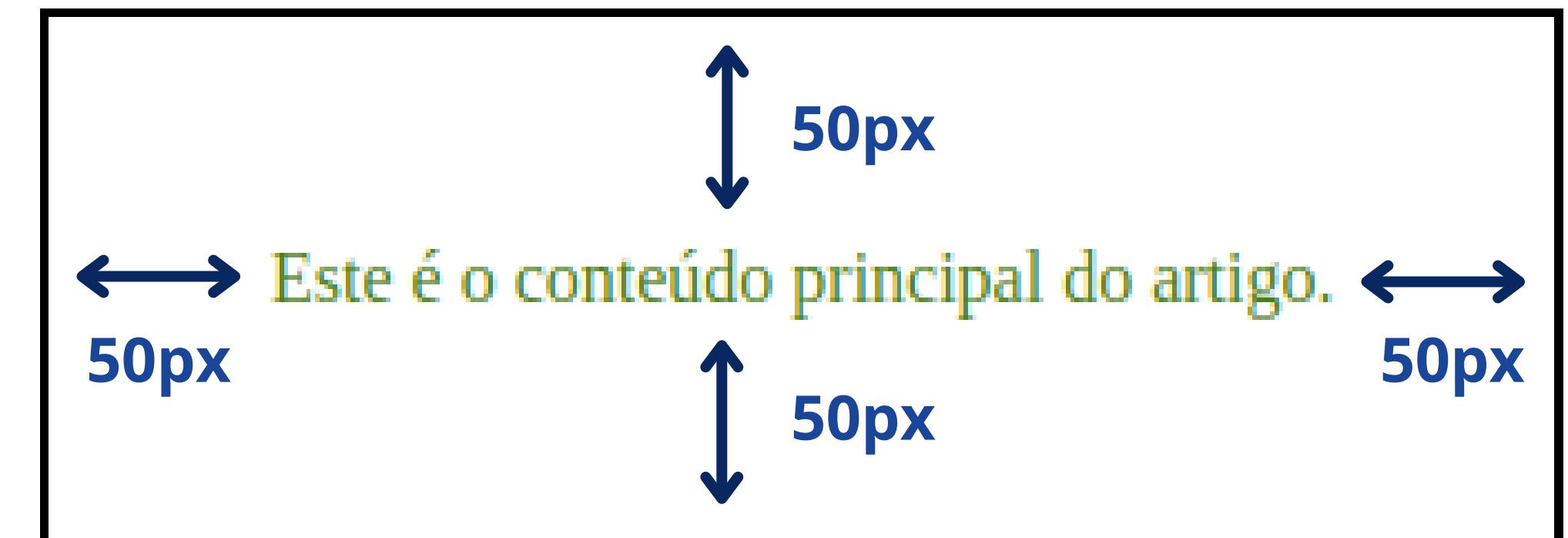
## Border, Margin e Padding no CSS

**Padding** (Preenchimento Interno)

O **padding** é o espaço interno entre o conteúdo do elemento e sua borda.

Ele afasta o conteúdo da borda.

```
estilo.css > ...
1 p {
2     color: green;
3     display: inline;
4     border: 2px solid black;
5     margin: 20px;
6     padding: 50px;
7 }
8
```



**Explicação:**

- 15px → adiciona 15 pixels de espaço dentro da borda, ao redor do texto.



# CURSO TÉCNICO EM INFORMÁTICA

## EXEMPLO COMPLETO:

```
<div style="margin: 20px; border: 2px solid black; padding: 15px;">
  <p>Este é um exemplo de caixa com margin, border e padding!</p>
</div>
```

Este é um exemplo de caixa com margin, border e padding!

**margin: 20px;** → 20px de espaço entre a `<div>` e outros elementos.

**border: 2px solid black;** → borda preta sólida ao redor da `<div>`.

**padding: 15px;** → 15px de espaço interno entre a borda e o conteúdo (`<p>`).



# CURSO TÉCNICO EM INFORMÁTICA

## A PROPRIEDADE DISPLAY NO CSS

A propriedade **display** é uma das mais importantes no CSS.

Ela define como um elemento HTML será exibido na página – ou seja, determina o tipo de caixa que o elemento vai se comportar.

Com **display**, você pode fazer um elemento se comportar como:

- Um **block**
- Um item **inline**
- Um elemento **flex**
- Um **grid**
- Ou até escondê-lo completamente! (**hidden**)



## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

### block

- O elemento ocupa toda a largura disponível.
- Sempre começa em uma nova linha

```
div {  
    display: block;  
}
```

Elementos que são block por **padrão**: <div>, <p>, <h1>–<h6>, <section>, <article>, etc.

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Document</title>  
    <link rel="stylesheet" href="estilo.css">  
  </head>  
  <body>  
  
    <div style="display: block; background-color: #blueviolet; padding: 20px;">  
      <p>Este é um exemplo de caixa com margin, border e padding!</p>  
    </div>  
  </body>  
</html>
```

Este é um exemplo de caixa com margin, border e padding!

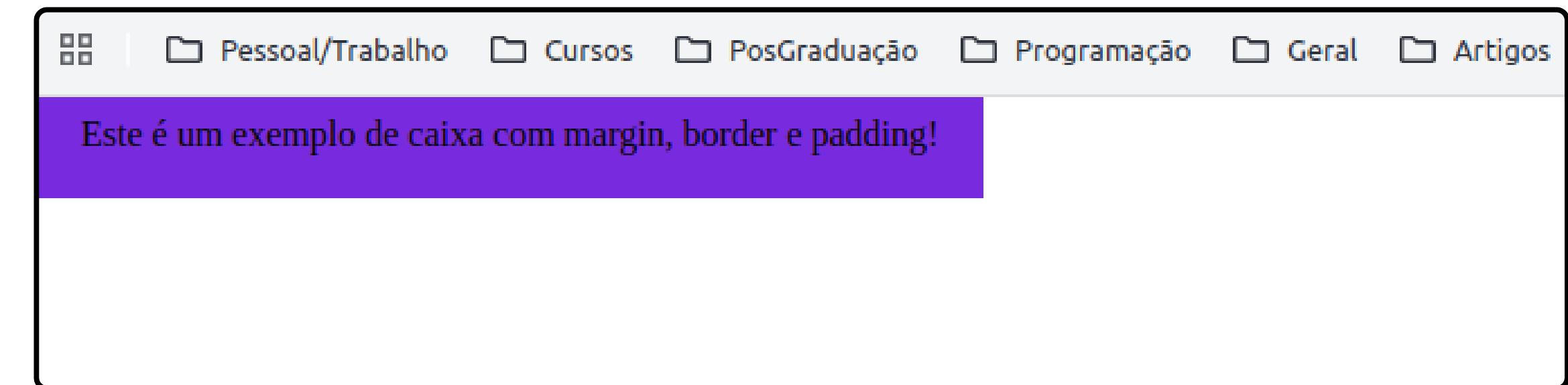


## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

### inline

- O elemento não quebra linha.
- Ocupa apenas o espaço necessário ao redor do seu conteúdo.
- Não aceita propriedades de largura e altura facilmente.

```
p{  
    display: inline;  
    background-color: blueviolet;  
    padding: 20px;  
}
```



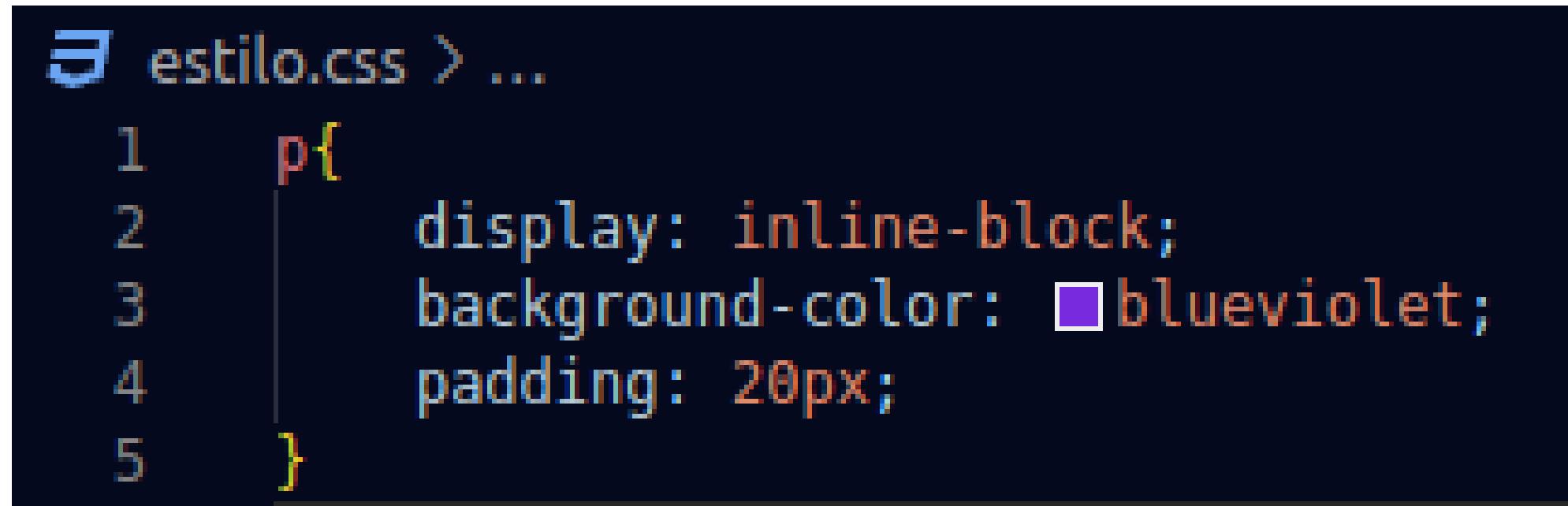
Elementos que são inline por padrão: **<span>**, **<a>**, **<strong>**, **<em>**, etc.



## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

### inline-block

- Parece um elemento inline (fica na mesma linha dos outros).
- Se comporta como bloco internamente (permite definir largura e altura).



```
estilo.css > ...
1 p{
2     display: inline-block;
3     background-color: blueviolet;
4     padding: 20px;
5 }
```

Este é um exemplo de caixa com margin, border e padding!

Este é um exemplo de caixa com margin, border e padding!



# CURSO TÉCNICO EM INFORMÁTICA

## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

### **none**

- Remove o elemento da página – ele não é exibido e não ocupa espaço.

```
p {  
    display: none;  
}
```



# CURSO TÉCNICO EM INFORMÁTICA

## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

Outros valores modernos

**flex** → Transforma o elemento em um flex container (para layouts mais dinâmicos).

**grid** → Transforma o elemento em um grid container (para criar grades de layout).

**inline-flex, inline-grid** → Versões inline de flex e grid.



# CURSO TÉCNICO EM INFORMÁTICA

## PRINCIPAIS VALORES DA PROPRIEDADE DISPLAY

### Resumo da propriedade `display`

Valor	Comportamento
<code>block</code>	Nova linha + ocupa toda a largura
<code>inline</code>	Mesma linha + ocupa só o necessário
<code>inline-block</code>	Mesma linha + aceita largura e altura
<code>none</code>	Some da página
<code>flex</code>	Organiza filhos em formato flexível
<code>grid</code>	Organiza filhos em uma grade

