

# CSS

Cascading Style Sheets

Deixe suas páginas com estilo!

Programação de Aplicativos | Nível Iniciante | Curso Técnico SENAI

# 1. O que é CSS?

**CSS** significa **Cascading Style Sheets** — Folhas de Estilo em Cascata. É a linguagem que diz ao navegador **como os elementos HTML devem aparecer**: cor, tamanho, fonte, espaçamento, posição...

Sem CSS, toda página da internet seria apenas texto preto num fundo branco. O CSS é o que torna a web bonita e organizada!

## A analogia perfeita

HTML	CSS	JavaScript
Esqueleto / Estrutura	Roupa e Maquiagem	Músculos e Movimento
O que existe na página	Como fica bonito	O que acontece

## Como o CSS é incluído no HTML?

Existem 3 formas, mas a mais recomendada é o arquivo externo:

Forma	Como usar	Recomendado?
Externo (ideal)	HTML: <code>&lt;link rel="stylesheet" href="styles.css"&gt;</code>	Sim - separa o código
Interno	Dentro de <code>&lt;style&gt;</code> no <code>&lt;head&gt;</code> do HTML	As vezes
Inline	Direto na tag: <code>&lt;p style="color:red"&gt;</code>	Evitar - difícil manter

### Como conectar HTML e CSS

```

<!-- No arquivo HTML, dentro do <head>: -->
<link rel="stylesheet" href="styles.css">

/* No arquivo styles.css: */
h1 {
  color: purple;
  font-size: 32px;
}

```

## 2. Sintaxe do CSS

Uma regra CSS sempre tem a mesma estrutura: **seletor + chaves + propriedade + valor**.

### Estrutura de uma regra CSS

```
seletor {
  propriedade: valor;
  outra-propriedade: outro-valor;
}

/* Exemplo real: */
p {
  color: #333333;
  font-size: 16px;
  line-height: 1.5;
}
```

### Os 3 tipos de seletores essenciais

O seletor diz **qual elemento** vai receber o estilo. Veja os tres mais importantes:

Seletor de Elemento	Seletor de Classe (.)	Seletor de ID (#)
<pre>h1 {   color: red; }</pre>	<pre>.destaque {   background: blue; }</pre>	<pre>#menu {   border: 2px   solid green; }</pre>
<p>Título em vermelho</p>	<p>.destaque aplicado</p> <p>sem a classe</p>	<p>#menu com borda</p>

**Dica rápida:** Seletor de elemento (h1, p, div) afeta **todos** os elementos daquele tipo. Classe (.nome) afeta **qualquer elemento** que tiver aquela classe. ID (#nome) afeta apenas **um elemento único** na página, que é o que contém aquele ID.

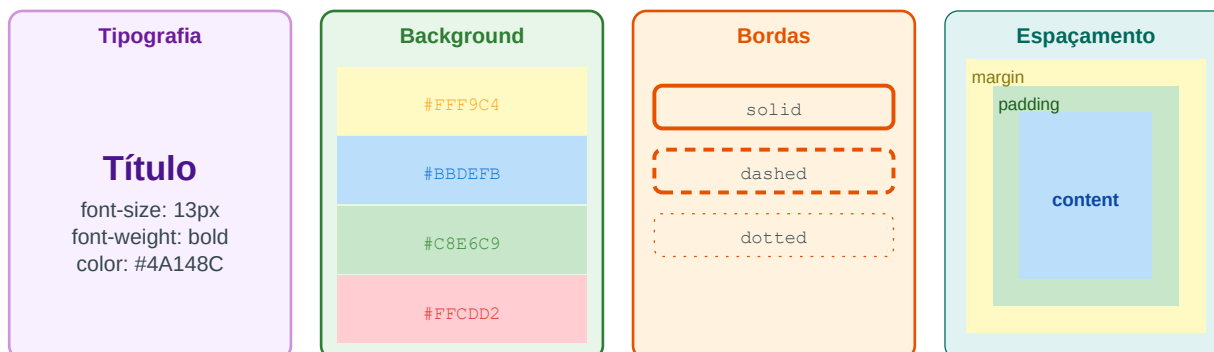
### Tabela de seletores

Seletor	Sintaxe CSS	Seleciona
Elemento	h1 { }	Todos os <h1> da página
Classe	.destaque { }	Todos com class="destaque"
ID	#menu { }	O elemento com id="menu"

Todos	* { }	Todos os elementos (cuidado!)
Filho direto	div > p { }	Somente os <p> filhos diretos de <div>
Descendente	div p { }	Todos os <p> dentro de qualquer <div>
Hover	a:hover { }	Um link quando o mouse passa por cima
Múltiplos	h1, h2, p { }	h1, h2 e p ao mesmo tempo

## 3. Propriedades Essenciais

Existem centenas de propriedades CSS, mas um pequeno grupo cobre a maioria dos casos. Veja os quatro grupos mais usados:



### Exemplos de código por grupo

#### Texto e Fontes

```
/* Tipografia */
p {
  font-family: Arial, sans-serif;
  font-size: 16px;
  font-weight: bold;
  color: #333333;
  text-align: center;
  line-height: 1.6;
}
```

#### Cores e Background

```
/* Cores e Fundo */
div {
  background-color: #F3E5F5;
  color: #6A1B9A;
  opacity: 0.9;
}
/* Cor transparente */
span {
  background: rgba(106,27,154,0.2);
}
```

#### Bordas

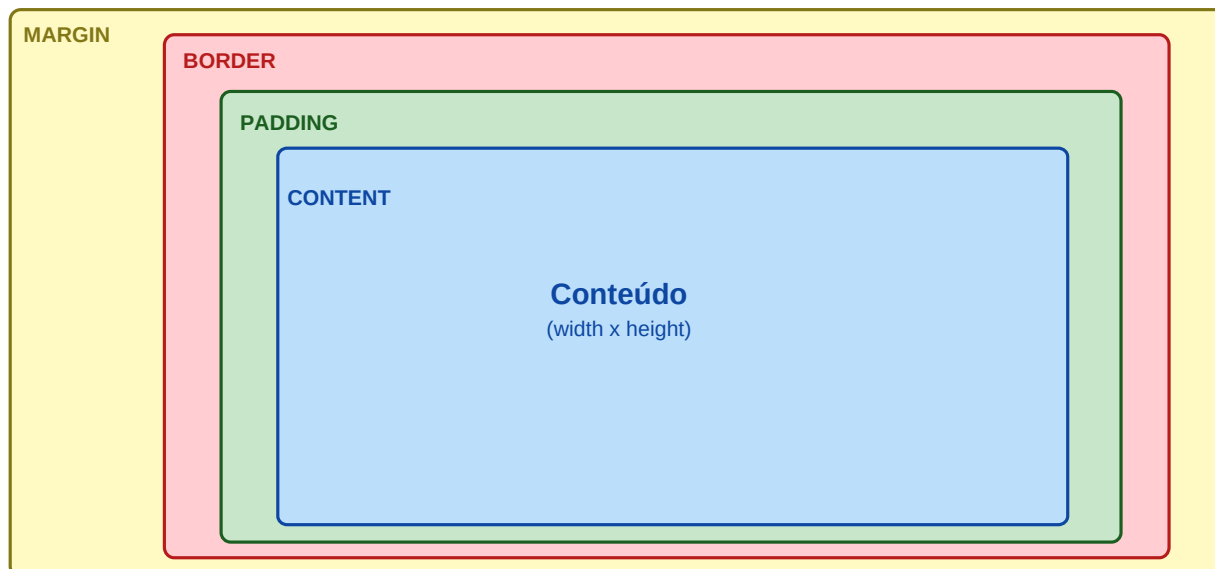
```
/* Bordas */
img {
  border: 2px solid #6A1B9A;
  border-radius: 8px;
}
/* So a borda inferior */
h2 {
  border-bottom: 3px solid purple;
}
```

#### Margin e Padding

```
/* Espaçamento */
.card {
  margin: 16px;
  padding: 24px;
}
/* Atalho: cima dir baixo esq */
.box {
  margin: 10px 20px 10px 20px;
}
```

## 4. O Box Model: como o CSS enxerga os elementos

No CSS, **todo elemento é tratado como uma caixa**. Essa caixa tem 4 camadas, de dentro para fora:



Camada	O que é	Propriedade CSS
CONTENT	O conteúdo em si (texto, imagem)	width, height
PADDING	Espaço interno entre conteúdo e borda	padding, padding-top...
BORDER	A linha que contorna o elemento	border, border-radius...
MARGIN	Espaço externo entre este e outros elementos	margin, margin-top...

### Box Model na prática

```
.card {
width: 300px; /* largura do conteudo */
padding: 20px; /* espaco interno */
border: 2px solid purple; /* borda */
margin: 16px; /* espaco externo */
}

/* Largura total real = 300 + 20+20 + 2+2 + 16+16 = 376px */
```

**Dica profissional:** adicione `box-sizing: border-box;` ao seu CSS. Isso faz o padding e a borda serem incluídos dentro do width, deixando os cálculos muito mais simples!

```
* { box-sizing: border-box; }
```

## 5. Display: como os elementos se comportam

A propriedade `display` controla como um elemento ocupa espaço na página. É uma das mais importantes do CSS!



Valor	Comportamento	Exemplo típico
block	Ocupa toda a largura, empilha na vertical	div, p, h1, section
inline	Ocupa só o necessário, fica na mesma linha	span, a, strong, em
inline-block	Fica na mesma linha, mas aceita width e height	Botoes, badges
flex	Distribui filhos em linha ou coluna, alinhados	Menus, cards lado a lado
grid	Cria grade com linhas e colunas	Layouts de página
none	Esconde o elemento completamente	Menus ocultos, modais

### Flexbox: o layout mais usado no dia a dia

#### Flexbox basico

```
.container {
  display: flex;
  justify-content: center; /* alinha horizontalmente */
  align-items: center; /* alinha verticalmente */
  gap: 16px; /* espaco entre os filhos */
}

/* Os filhos ficam lado a lado automaticamente: */
.container div {
  width: 100px;
  height: 100px;
  background-color: purple;
}
```

## 6. Cores no CSS

O CSS aceita cores de varias formas. Todas as cores abaixo são equivalentes (todas são o mesmo vermelho, por exemplo):

red	#E53935	rgb(229, 57, 53)	#1565C0	rgb(46, 125, 50)	#F9A825	#6A1B9A	#00695C
Nome	Hex	RGB	Hex	RGB	Hex	Hex	Hex

### Formas de escrever cores

Formato	Exemplo	Quando usar
Nome	color: red;	Para cores básicas e testes rápidos
Hexadecimal	color: #E53935;	O mais comum no mercado
RGB	color: rgb(229, 57, 53);	Quando quiser controle preciso
RGBA	color: rgba(229,57,53, 0.5);	Para transparência (0.0 a 1.0)
HSL	color: hsl(2, 76%, 55%);	Para criar paletas harmônicas

## 7. Exemplo Completo: cartão de perfil estilizado

Veja um exemplo real unindo HTML + CSS para criar um cartão de perfil bonito:

## perfil.html

```
<!-- perfil.html -->
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="estilo.c
</head>
<body>
<div class="card">

<h2 class="nome">Ana Silva</h2>
<p class="cargo">Desenvolvedora</p>
<button class="btn">Ver perfil</butto
</div>
</body>
</html>
```

## estilo.css

```
/* estilo.css */
.card {
width: 220px;
padding: 24px;
border-radius: 12px;
background: #F3E5F5;
text-align: center;
box-shadow: 0 4px 12px rgba(0,0,0,0.15);
}
.foto {
width: 80px;
border-radius: 50%;
border: 3px solid #6A1B9A;
}
.nome { color: #4A148C; }
.cargo { color: #78909C; }
.btn {
background: #6A1B9A;
color: white;
border: none;
padding: 8px 20px;
border-radius: 20px;
cursor: pointer;
}
```

## 8. Resumo Final

Sintaxe	Seletores	Box Model	Layout
<pre>seletor {   prop: valor; }</pre>	elemento: h1 classe: .nome ID: #id hover: a:hover	content padding border margin	block inline flex grid / none

### Próximos passos:

1. Crie o arquivo styles.css e conecte ao seu HTML com a tag link
2. Estilize o cartão de perfil do exemplo e veja funcionar no navegador
3. Experimente o display: flex para colocar cards lado a lado
4. Use o DevTools (F12 > Elements) para inspecionar e testar estilos ao vivo
5. Desafio: recrie o layout de um site que você gosta usando apenas HTML e CSS

**Lembre-se:** CSS é prática. Quanto mais você testar e errar, mais rápido aprende. Não tenha medo de mudar valores e ver o que acontece. O navegador mostra o resultado na hora!