



# Introdução ao CSS



## que é CSS?

Nesta primeira parte da nossa aula sobre CSS, introduziremos e explorar essa linguagem essencial para o desenvolvimento web. O CSS, que significa Cascading Style Sheets (Folhas de Estilo em Cascata), é uma linguagem de estilo utilizada para controlar a apresentação visual das páginas web. Em outras palavras, enquanto o HTML estrutura o conteúdo de uma página, o CSS cuida da aparência desse conteúdo, tornando as páginas mais atraentes e fáceis de usar.

Quando aplicamos o CSS a uma página, ele transforma o HTML, que sem estilo algum pode se assemelhar a um bloco de notas com texto corrido, em algo visualmente agradável e esteticamente organizado. Com o CSS, é possível personalizar cores, fontes, tamanhos, espaçamentos, e muito mais, garantindo que a página tenha um design que seja tanto visualmente atrativo quanto funcional.

A principal função do CSS é justamente essa: separar o conteúdo (estrutura e semântica) da apresentação visual. Essa separação traz inúmeras vantagens, como a facilidade de manutenção e o reaproveitamento do código. Em vez de misturar o estilo diretamente no HTML, o CSS permite que o estilo seja aplicado separadamente, facilitando tanto o desenvolvimento quanto a manutenção do site.

Além disso, o CSS permite que diferentes estilos sejam aplicados a elementos específicos da página, possibilitando personalizar cada detalhe visual conforme as necessidades do projeto. Essa flexibilidade faz com que o CSS seja uma ferramenta indispensável para qualquer desenvolvedor web.

Por fim, é importante destacar que a utilização do CSS melhora significativamente a experiência do usuário ao interagir com uma página web. Com ele, é possível garantir que o conteúdo se ajuste corretamente ao tamanho da tela, seja ela de um desktop, tablet ou smartphone, proporcionando uma navegação mais fluida e agradável.

Começamos a compreender o que é o CSS, como ele funciona e por que ele é tão importante no desenvolvimento de páginas web. Na próxima parte, exploraremos os fundamentos do CSS, onde entenderemos as regras básicas e os conceitos necessários para aplicá-lo de maneira eficiente em nossos projetos.

## Fundamentos do CSS

Nesta segunda parte, vamos aprofundar o entendimento sobre os fundamentos dessa linguagem. Compreender os fundamentos do CSS é essencial para utilizá-lo de forma eficaz e otimizada no desenvolvimento de páginas web.

O CSS opera por meio de uma série de regras que determinam como os elementos de uma página serão exibidos no navegador. Cada uma dessas regras é composta por seletores e declarações. Os seletores indicam quais elementos do HTML serão estilizados, enquanto as declarações especificam quais propriedades visuais serão aplicadas a esses elementos.

Para o CSS funcionar corretamente, é importante seguir algumas diretrizes fundamentais. Primeiro, é necessário respeitar a hierarquia das regras. No CSS, as regras podem se sobrepor, e a ordem em que elas são aplicadas pode afetar o resultado final. Por isso, entender o conceito de cascata é crucial. A cascata determina a prioridade das regras, considerando a origem do estilo (externo, interno ou inline), a especificidade dos seletores e a ordem na qual as regras aparecem no código.

Outro ponto fundamental é a herança. Algumas propriedades no CSS são herdadas pelos elementos filhos a partir de seus elementos pai. Por exemplo, se você der a cor do texto para um elemento pai, os elementos filhos herdarão essa cor, a menos que seja especificado um estilo diferente para eles. Compreender como a herança funciona permite evitar redundâncias e escrever códigos mais limpos e eficientes.


Além disso, o conceito de box model é central para o entendimento do CSS. Todo elemento em uma página web é representado como uma caixa retangular que envolve o conteúdo, padding, borda e margem. Saber como essas partes se relacionam ajuda a controlar o layout da página com precisão, permitindo ajustes finos na disposição dos elementos.

Também é importante ter em mente a responsividade das páginas. Com a diversidade de dispositivos e tamanhos de tela disponíveis hoje, o CSS deve ser utilizado para garantir que a página se adapte bem a diferentes ambientes. Isso envolve o uso de unidades flexíveis, como porcentagens e em, e de media queries para aplicar estilos específicos dependendo das características do dispositivo, como a largura da tela.

Por fim, para otimizar o desempenho das páginas, é essencial minimizar a quantidade de regras CSS desnecessárias e garantir que o código seja o mais limpo e conciso possível. Isso ajuda não só na manutenção do site, mas também no tempo de carregamento, proporcionando uma melhor experiência para o usuário.

Neste tópico, abordamos os fundamentos do CSS, explorando conceitos como regras, hierarquia, herança, box model e responsividade. Compreender esses fundamentos é o primeiro passo para dominar o CSS e utilizá-lo de forma eficiente no desenvolvimento web.

## **Como aplicar CSS na página**


Nesta terceira parte da nossa aula sobre CSS, exploraremos as diferentes formas de aplicar CSS em uma página HTML. Entender essas abordagens é fundamental para escolher a mais adequada conforme o projeto em desenvolvimento. 

Existem três métodos principais para aplicar CSS em um documento HTML: o estilo inline, o estilo interno e o estilo externo. Cada um desses métodos tem suas características, vantagens e desvantagens, que analisamos a seguir.

A primeira forma é o estilo inline. Neste método, o CSS é aplicado diretamente no elemento HTML, utilizando o atributo `style`. Por exemplo, podemos aplicar uma borda ou definir o espaçamento interno de um elemento diretamente na tag HTML correspondente. Embora essa abordagem seja útil para personalizações rápidas e específicas, não é recomendada para páginas grandes ou complexas. Isso porque o código pode se tornar difícil de manter e atualizar, além de perder a organização, pois cada elemento terá suas próprias regras CSS. Portanto, o uso de estilos inline deve ser limitado a casos excepcionais, onde é necessário estilizar um único elemento de maneira rápida e isolada.

O segundo método é o estilo interno. Neste caso, o CSS é inserido dentro de uma tag `style` localizada na seção head do documento HTML. Essa abordagem é útil quando se deseja aplicar estilos a uma única página, sem a necessidade de um arquivo CSS separado. Ao colocar as regras CSS dentro do head, o navegador é instruído a aplicar esses estilos a todos os elementos da página, conforme as regras definidas. Esse método é mais organizado do que o inline, por centralizar os estilos em um único local dentro do HTML. No entanto, para projetos maiores ou com múltiplas páginas, pode ainda ser menos eficiente do que o método externo, pois os estilos internos não são reaproveitados em outras páginas.

Por fim, temos o estilo externo, considerado a melhor prática para a maioria dos projetos. Aqui, o CSS é colocado em um arquivo separado, com a extensão `.css`, que é então vinculado ao documento HTML através de uma tag `link` também no head. Esse método oferece uma clara separação entre a estrutura (HTML) e a apresentação (CSS), facilitando a manutenção e o reaproveitamento de estilos em

várias páginas. Além disso, o uso de arquivos CSS externos contribui para um código HTML mais limpo e organizado, e permite que as mudanças nos est. sejam feitas de forma centralizada, afetando todas as páginas que utilizam o mesmo arquivo CSS.

Agora, para colocar em prática o que aprendemos, vamos ver um exemplo simples. Suponha que estamos trabalhando em um arquivo HTML chamado `estoque.html`. Se quisermos aplicar estilos utilizando o método interno, começamos inserindo a tag `style` dentro do head do documento. Dentro dessa tag, podemos definir regras como o tamanho da fonte (`font-size`), a espessura da fonte (`font-weight`) e a cor de fundo (`background-color`). Por exemplo:

```
<style>
```

```
body {
```

```
font-size: 10px;
```

```
font-weight: bold;
```

```
background-color: gray;
```

```
}
```

```
</style>
```

Essas regras serão aplicadas a todos os elementos dentro do body da página, alterando o tamanho da fonte para 10 píxeis, tornando o texto em negrito e mudando a cor de fundo para cinza. Essa abordagem mostra como o CSS pode ser usado para estilizar uma página inteira de maneira eficaz e organizada, sem a necessidade de editar cada elemento individualmente no HTML.



## Sintaxe básica do CSS

Neste quarto e último tópico, vamos explorar a sintaxe básica dessa linguagem, essencial para aplicarmos estilos de forma eficiente e organizada em nossas páginas web. A compreensão da sintaxe é fundamental para utilizar o CSS de maneira correta e maximizar seu potencial no desenvolvimento de projetos.

A sintaxe básica do CSS é composta por três elementos principais: seletores, propriedades e valores. Esses três componentes são a base sobre a qual o CSS é construído, sendo responsáveis por definir como os estilos serão aplicados aos elementos HTML.

Seletores são usados para identificar os elementos HTML que queremos estilizar. Existem vários tipos de seletores, cada um com uma função específica. O seletor mais simples é o seletor de tipo, que se refere diretamente às tags HTML, como `body`, `p`, `div`, etc. Por exemplo, ao usar o seletor `body`, estamos direcionando o CSS para aplicar estilos a todo o conteúdo dentro da tag `body` da página.

Outro tipo importante é o seletor de classe, que permite aplicar estilos a um grupo de elementos que compartilham a mesma classe. As classes são definidas no HTML com o atributo `class` e são precedidas por um ponto (.) no CSS. Isso nos permite estilizar múltiplos elementos de forma consistente e eficiente. Já o seletor de ID, que usa o símbolo `#`, é mais específico e aplica estilos a um único elemento identificado por um ID único no HTML. Este seletor é útil quando precisamos estilizar um elemento específico de forma única.

Além disso, temos o seletor universal, representado pelo asterisco (\*), que aplica o estilo a todos os elementos da página. Embora poderoso, deve ser usado com cautela, pois pode gerar conflitos com outros seletores. Também existe o seletor de atributo, que permite estilizar elementos com base em atributos específicos, como o tipo de `input`, por exemplo.

Propriedades são as características que queremos alterar nos elementos selecionados. As propriedades podem abranger uma ampla gama de aspectos visuais, como cor, tamanho da fonte, bordas, espaçamento, alinhamento, entre outros. Algumas propriedades comuns incluem `color` para definir a cor do texto, `background-color` para a cor de fundo, `font-size` para o tamanho da fonte, e `margin` e `padding` para controlar os espaçamentos externos e internos dos elementos, respectivamente. Cada propriedade define um aspecto visual específico que pode ser ajustado conforme as necessidades do design.

Valores são as especificações dadas às propriedades. Por exemplo, ao definir `color: blue;`, estamos dizendo ao CSS para aplicar a cor azul ao texto. Os valores podem ser expressos de várias maneiras: nomes de cores em inglês, valores hexadecimais, unidades de medida como píxeis (px), em, rem, e porcentagens, entre outros. Cada valor precisa ser adequado à propriedade que está sendo definida para garantir que o estilo seja aplicado corretamente.

Ao escrever uma regra CSS, seguimos a seguinte estrutura: primeiro, especificamos o seletor, seguido por chaves `{ }` que contêm as propriedades e seus respectivos valores. Por exemplo:

```
body {  
  
    font-size: 20px;  
  
    font-weight: bold;  
  
    background-color: antiquewhite;  
  
}
```

Nesse exemplo, todos os textos dentro do body terão um tamanho de fonte de 20 píxeis, estarão em negrito, e o fundo será de cor antiquewhite.



Na prática, ao aplicar o CSS em uma página, é possível estilizar elementos de maneira eficaz, alterando as propriedades de acordo com as necessidades do design. Se, por exemplo, quisermos estilizar uma tabela específica, podemos usar um seletor como `body > table` para aplicar estilos apenas àquela tabela que está diretamente associada ao body. Isso nos permite uma personalização detalhada e evita conflitos com outros elementos da página. Como neste exemplo:

```
body > table {  
  
    font-family: sans-serif;  
  
    border: 2px solid;  
  
    padding: 10px;  
  
}
```

Onde a tabela (table) que é um filho direto do body utilizará a fonte Sans-Serif, terá uma borda sólida de 2 píxeis e um preenchimento (padding) de 10 píxeis.

Como vimos, a sintaxe do CSS oferece uma estrutura clara e flexível para definir e aplicar estilos às páginas web. Com o conhecimento desses conceitos fundamentais, vocês estão preparados para explorar ainda mais as possibilidades que o CSS oferece, seja no ajuste fino de elementos individuais ou na criação de layouts complexos e responsivos.

Encerramos aqui esta aula sobre a Introdução ao CSS. Espero que tenham compreendido bem os conceitos abordados e estejam prontos para continuar



explorando essa poderosa linguagem. O CSS abre um vasto leque de possibilidades para quem deseja criar páginas web visualmente atraentes e funcionais.



## Conteúdo Bônus

Para aprofundar seu conhecimento sobre CSS e revisar os conceitos iniciais abordados em aula, recomendo assistir ao vídeo **“TUTORIAL BÁSICO HTML e CSS: Introdução a código para não-programadores”**. Este vídeo é apresentado por Luiz Ludwig no YouTube e oferece uma introdução clara e prática ao uso de HTML e CSS, ideal para reforçar os fundamentos que discutimos.

## Referência Bibliográfica

ASCENCIO, A. F. G.; CAMPOS, E. A. V. de. **Fundamentos da programação de computadores**. 3.ed. Pearson: 2012.

BRAGA, P. H. **Teste de software**. Pearson: 2016.

GALLOTTI, G. M. A. (Org.). **Arquitetura de software**. Pearson: 2017.

GALLOTTI, G. M. A. **Qualidade de software**. Pearson: 2015.

MEDEIROS, E. **Desenvolvendo software com UML 2.0 definitivo**. Pearson: 2004.

PFLEEGER, S. L. **Engenharia de software: teoria e prática**. 2.ed. Pearson: 2003.

SOMMERVILLE, I. **Engenharia de software**. 10.ed. Pearson: 2019.

## Prática Integradora Desenvolvimento de Software

### Atividade Prática 5 – Explorando e Aplicando CSS

## Objetivos:



- Compreender os fundamentos do CSS e sua importância no desenvolvimento web.
- Aplicar diferentes métodos de inserção de CSS em uma página HTML.
- Utilizar a sintaxe básica do CSS para estilizar elementos de uma página web.

## Materiais, Métodos e Ferramentas:

- Computador com editor de código (Visual Studio Code, Sublime Text, etc.)
- Navegador web (Chrome, Firefox, etc.)
- Arquivo HTML e CSS para exercícios práticos

## Atividade Prática

Primeiramente, leia atentamente o texto a seguir:

Nesta atividade prática, você será responsável por estilizar uma página HTML utilizando CSS. Você começará aplicando estilos diretamente na página e, em seguida, criará um arquivo CSS externo para organizar os estilos de maneira eficiente. A atividade ajudará a consolidar sua compreensão dos fundamentos do CSS e das diferentes formas de aplicação dos estilos.

Agora, vamos praticar!

- **Criação do Arquivo HTML:**

a. Crie um novo arquivo HTML chamado atividade.html.



b. Insira o seguinte conteúdo no arquivo HTML:

```
<!DOCTYPE html>
```

```
<html lang="pt-BR">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Atividade de CSS</title>
```

```
</head>
```

```
<body>
```

```
  <header>
```

```
    <h1>Bem-vindo à Atividade de CSS</h1>
```

```
  </header>
```

```
  <main>
```

```
    <p class="intro">Esta é uma página para praticar a aplicação de CSS.
```

```
</p>
```

```
    <p id="especial">Estilize este parágrafo de forma única.</p>
```

```
  <table>
```

```
    <tr>
```

<th>Nome</th>



<th>Idade</th>

</tr>

<tr>

<td>João</td>

<td>25</td>

</tr>

<tr>

<td>Maria</td>

<td>30</td>

</tr>

</table>

</main>

<footer>

<p>© 2024 Prática de CSS</p>

</footer>

</body>

</html>

- **Aplicação de Estilos Inline:**

a. Adicione um estilo inline ao <header> para alterar a cor do texto para azul e definir a cor de fundo como cinza claro.



b. Adicione um estilo inline ao <footer> para centralizar o texto e definir a cor de fundo como preto com texto branco.

- **Aplicação de Estilos Internos:**

a. No arquivo atividade.html, adicione uma tag <style> dentro da seção <head>. Dentro dessa tag, adicione regras CSS para estilizar os seguintes elementos:

- Aumentar o tamanho da fonte do parágrafo com a classe intro para 18px e definir a cor do texto como verde.
- Aplicar uma borda de 2px sólida preta à tabela.

b. Seu código CSS interno deve se parecer com isto:

```
<style>
```

```
.intro {
```

```
    font-size: 18px;
```

```
    color: green;
```

```
}
```

```
table {
```

```
    border: 2px solid black;
```

```
    width: 100%;
```

}



</style>

- **Criação e Aplicação de Estilos Externos:**

a. Crie um novo arquivo chamado estilos.css.

b. Adicione regras CSS no arquivo estilos.css para estilizar os seguintes elementos:

- Definir o fundo da página como cor de antiquewhite.
- Estilizar o parágrafo com o ID especial para ter um fundo amarelo e texto em negrito.

c. No arquivo atividade.html, vincule o arquivo estilos.css usando a tag <link> na seção <head>.

d. Seu arquivo estilos.css deve conter:

```
body {
```

```
    background-color: antiquewhite;
```

```
}
```

```
#especial {
```

```
    background-color: yellow;
```

```
    font-weight: bold;
```

```
}
```

## Gabarito Esperado



- **Estilos Inline:** O <header> deve ter a cor do texto em azul e fundo cinza claro, e o <footer> deve estar centralizado com fundo preto e texto branco.
- **Estilos Internos:** O parágrafo com a classe intro deve ter a fonte de 18px e cor verde, e a tabela deve ter uma borda de 2px sólida preta.
- **Estilos Externos:** O fundo da página deve ser antiquewhite, e o parágrafo com o ID especial deve ter um fundo amarelo e texto em negrito.

Essa atividade cobre a criação e aplicação de CSS de várias formas, ajudando os alunos a solidificar seus conhecimentos sobre os fundamentos e métodos de estilização.

**Ir para exercício**