



CSS

Introdução

Hoje, vamos iniciar nossa jornada pelo universo do CSS, uma linguagem fundamental para estilizar e tornar visualmente atraentes as páginas HTML. O CSS, sigla para Cascading Style Sheets, permite definir cores, layouts e diversos estilos que transformam uma estrutura simples em um design sofisticado. Começaremos explorando as definições básicas do CSS, entendendo sua função essencial no desenvolvimento web. Ao longo desta aula, veremos como aplicar seletores, propriedades e valores para estilizar elementos HTML, além de conceitos importantes como o Box Model e as categorias de valores que podem ser utilizados.

Definições

Nesta primeira parte, vamos explorar as definições e conceitos básicos dessa linguagem essencial para o desenvolvimento web. O CSS, ou Cascading Style Sheets (Folhas de Estilo em Cascata), é uma linguagem utilizada para estilizar páginas HTML. No HTML que desenvolvemos, inserimos um código CSS no cabeçalho para definir as propriedades, criar seletores e especificar valores para cada seletor. Esta abordagem faz parte do CSS e será o foco do nosso estudo.

Primeiramente, é fundamental entender o que é o CSS e qual é sua função. O CSS é responsável pela estilização das páginas HTML, incluindo a definição de estilos, cores e outras características visuais. Embora seja possível incorporar o CSS diretamente no HTML, a prática recomendada é

criar um arquivo separado para o CSS. Isso garante uma organização mais eficiente e facilita a manutenção do código.

Veja a seguir um exemplo básico de como o CSS é usado para estilizar um parágrafo em HTML:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Exemplo de CSS</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <p class="paragrafo-destaque">Este é um parágrafo com destaque.</p>
</body>
</html>
```

```
.paragrafo-destaque {
  background-color: yellow;
  padding: 10px;
  border: 2px solid black;
  font-size: 18px;
}
```

O CSS funciona com base em uma estrutura básica que inclui seletores, propriedades e valores. Os seletores são usados para identificar os elementos HTML que serão estilizados. Dentro dos seletores, especificamos as propriedades que desejamos alterar e atribuímos valores a essas propriedades. A hierarquia e a cascata no CSS determinam como as regras de estilo são aplicadas, com a cascata permitindo que as regras sejam sobrepostas conforme necessário.

Para ilustrar o uso do CSS, vamos considerar um exemplo prático. Suponha que temos um arquivo HTML com um parágrafo ao qual aplicamos uma classe chamada “parágrafo-destaque”. No arquivo CSS, criamos uma regra para essa classe, definindo, por exemplo, a cor de fundo como amarelo e

ajustando o padding e as bordas. Quando o CSS é aplicado ao HTML, o parágrafo com a classe “parágrafo-destaque” exibe as características definidas no arquivo CSS. Se modificarmos as propriedades no CSS, essas mudanças são refletidas imediatamente na página.

Ao trabalhar com CSS, é essencial praticar e experimentar diferentes propriedades e valores para entender como eles afetam a aparência dos elementos HTML. O conhecimento da estrutura básica e das regras do CSS, como o uso de seletores, propriedades e valores, é crucial para a criação de páginas web visualmente atraentes e bem estruturadas.

Modelo de formatação

Agora, vamos aprofundar o conceito de modelo de formatação. O CSS é uma ferramenta fundamental para estilizar e embelezar páginas web, complementando o HTML, que define a estrutura da página. Enquanto o HTML estabelece a organização dos elementos, o CSS é responsável por aplicar estilos, como cores e formatações, para tornar a página visualmente atraente.

Como discutido anteriormente, o CSS utiliza seletores, propriedades e valores para definir estilos. Neste tópico, vamos explorar como aplicar esses conceitos para criar diferentes modelos de formatação. Veja um exemplo prático de como usar o CSS para estilizar divs com classes específicas:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Modelo de Formatação</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container">
    <div class="box">Caixa 1</div>
    <div class="box">Caixa 2</div>
  </div>
</body>
</html>
```

```
.container {
  background-color: #f0f0f0;
  padding: 20px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

.box {
  background-color: blue;
  color: white;
  padding: 15px;
  margin: 10px 0;
  border-radius: 5px;
  text-align: center;
}
```

Em nossa prática, trabalharemos com dois arquivos essenciais: o arquivo HTML e o arquivo CSS. O arquivo HTML define a estrutura da página, enquanto o CSS é utilizado para a estilização. No exemplo, temos um arquivo HTML chamado index.html, que contém a estrutura básica do documento, incluindo a tag <header> e <body>, e um link para o arquivo CSS que contém os estilos aplicados.

No arquivo HTML, encontramos um novo elemento chamado <div>, que serve para agrupar conteúdo. Utilizamos a <div> para criar seções distintas na página e aplicar classes para estilização. Por exemplo, adicionamos uma classe chamada "box" a um elemento <div>, que será estilizado no CSS. O arquivo CSS é onde definimos os seletores, propriedades e valores. Por exemplo, o seletor 'container' pode ser utilizado para estilizar um contêiner

com um fundo branco e aplicar sombras. O seletor 'box' pode ser usado para definir um retângulo azul com padding e margens específicas.

Para visualizar essas alterações, abrimos o arquivo HTML em um navegador e aplicamos estilos no CSS. Por exemplo, ao modificar a cor de fundo do seletor .container, a mudança será refletida na página renderizada. Alterações no seletor .box, como mudar a cor de fundo ou o padding, também serão imediatamente visíveis. Testar e ajustar essas propriedades no CSS é uma maneira eficaz de aprender e entender como o CSS afeta a aparência da página.

Box Model

Agora, vamos focar no conceito de Box Model, que é fundamental para entender como o CSS organiza e exibe os elementos na página. O Box Model descreve como cada elemento HTML é tratado e exibido, e inclui componentes essenciais como o conteúdo, o padding (preenchimento), a borda e a margem.

O Box Model é composto por quatro partes principais: o conteúdo, o padding, a borda e a margem. O conteúdo refere-se ao texto ou elementos dentro da caixa. Observe um exemplo prático de como essas propriedades podem ser aplicadas em um arquivo CSS para visualizar o Box Model em ação:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Box Model</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="box">Conteúdo da Caixa</div>
</body>
</html>
```

```
.box {
  width: 300px;
  padding: 20px;
  border: 5px solid black;
  margin: 20px;
  background-color: lightblue;
}
```

O padding é o espaço entre o conteúdo e a borda da caixa. A borda é a linha ao redor do padding, e a margem é o espaço fora da borda, que separa o elemento de outros elementos na página.

Vamos explorar o Box Model na prática. Suponha que temos dois arquivos: index.html e styles.css. No arquivo HTML, definimos uma estrutura básica com um título e uma 'div' com a classe box. No CSS, estilizamos essa 'div' para visualizar como o Box Model afeta o layout. O seletor .box define o tamanho da caixa, a cor da borda, o padding e a margem. Por exemplo, podemos definir uma largura de 300 pixels, um padding de 20 pixels, uma borda sólida de 5 pixels e uma margem de 20 pixels. Além disso, podemos alterar a cor de fundo para ilustrar como cada componente do Box Model influencia o design da caixa.

Essas propriedades e valores permitem personalizar a aparência dos elementos na página de forma precisa. Alterar o padding ajusta o espaço interno da caixa, enquanto a borda define a espessura e o estilo da linha ao redor da caixa. A margem ajusta o espaço externo, controlando a separação entre a caixa e outros elementos. Compreender e manipular

essas propriedades é crucial para criar layouts eficientes e visualmente agradáveis em suas páginas web.

Categorias e Valores

Nesta última parte da nossa aula sobre CSS, vamos explorar as categorias e valores que podem ser aplicados ao estilizar elementos HTML. O CSS é essencial para o desenvolvimento web, complementando o HTML ao fornecer estilos e cores às páginas. Revisando o que discutimos anteriormente, vimos que o HTML cria a estrutura da página, enquanto o CSS define sua aparência. Falamos sobre seletores, propriedades, valores e o conceito de Box Model, que abrange elementos como background, padding, margem e conteúdo.

Agora, vamos nos concentrar nas categorias de propriedades que podem ser aplicadas aos elementos e nos diferentes valores que essas propriedades podem ter. Veja abaixo está um exemplo de como essas propriedades podem ser aplicadas a um título, parágrafo e botão em HTML:

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Categorias e Valores</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Exemplo de Título</h1>
  <p class="texto">Este é um parágrafo estilizado com CSS.</p>
  <button class="botao">Clique Aqui</button>
</body>
</html>
```

```
h1 {  
  color: darkred;  
  font-family: Arial, sans-serif;  
  text-align: center;  
}  
  
.texto {  
  font-size: 16px;  
  line-height: 1.5;  
  color: #333;  
}
```

```
.botao {  
  background-color: green;  
  color: white;  
  padding: 10px 20px;  
  border: none;  
  border-radius: 5px;  
  cursor: pointer;  
}  
  
.botao:hover {  
  background-color: darkgreen;  
}
```

As categorias incluem estilização de texto, imagens, figuras e botões, entre outros. Cada uma dessas categorias possui diversas propriedades que podem ser ajustadas para atingir o estilo desejado. Por exemplo, podemos controlar a cor, a fonte e o alinhamento do texto, além de aplicar estilos às bordas e margens de elementos.


Para ilustrar, considere um exemplo prático. Temos um arquivo HTML com um título, um parágrafo e uma div com a classe box. No CSS, podemos definir propriedades como a cor de fundo, a cor da borda, o padding e a margem dessa div. Podemos utilizar valores de cor em diferentes formatos, como nomes de cores (por exemplo, "blue"), códigos hexadecimais (por exemplo, "#0000FF") ou valores RGB (por exemplo, "rgb(0,0,255)"). Todas essas opções produzem o mesmo resultado visual, permitindo flexibilidade na estilização.

As propriedades de texto controlam a aparência do conteúdo textual, permitindo ajustar a cor, a fonte, o tamanho e o espaçamento entre as linhas. Por exemplo, podemos definir a cor do texto diretamente com o nome da cor ou usar um código hexadecimal para maior precisão. Além disso, as unidades de medida, como pixels, são amplamente utilizadas para definir tamanhos e espaçamentos no CSS.

Para uma prática mais aprofundada, abra seu arquivo HTML e o correspondente CSS no editor de código. Experimente alterar os valores das propriedades e observe como as mudanças são refletidas na página renderizada. Testar diferentes valores e formatos é fundamental para entender como cada propriedade afeta a aparência dos elementos.

Nesta aula, exploramos os fundamentos do CSS, abordando desde suas definições e conceitos básicos até o modelo de formatação e o Box Model. Entendemos a importância dos seletores, propriedades e valores na criação de estilos para elementos HTML. Vimos como diferentes categorias e valores podem ser aplicados para estilizar textos, imagens e outros elementos. Ao praticar e experimentar essas técnicas, você desenvolverá a habilidade de criar páginas web visualmente atraentes e funcionais.

Conteúdo Bônus

Para se aprofundar no tema desta aula, sugiro que assista ao vídeo intitulado “5 Projetos HTML + CSS para INICIANTES (Frontend) ”, que está disponível no canal Attekita Dev no YouTube.

Referência Bibliográfica

BONATTI, D. **Desenvolvimento de Sites Dinâmicos com Dreamweaver CC.**

Brasport: 2013.

BONATTI, D. **Desenvolvimento de Jogos em HTML5**. Brasport: 2014.

FLATSCHART, F. HTML 5 - **Embarque Imediato**. Brasport: 2011.

JOÃO, B. do N. (Org.). **Informática aplicada**. 2.ed. Pearson: 2019.

MARINHO, A. L.; CRUZ, J. L. da. **Desenvolvimento de aplicações para Internet**. 2.ed. Pearson: 2020

NEVES, M. C. B. de A. **Sites de Alta Performance**. Contentus: 2020

SOUSA, R. F. M. CANVAS HTML 5 - **Composição gráfica e interatividade na web**. Brasport: 2018.

TANENBAUM, A. S.; FEAMSTER, N.; WETHERALL, D. J. **Redes de computadores**. 6.ed. Pearson: 2021.

Atividade Prática 4 - CSS

Título da Prática: Entendendo HTML e CSS

Objetivos:

- Objetivo 1 – Aprendendo CSS.
- Objetivo 2 - Integração entre HTML e CSS.
- Objetivo 3 – Colocando em prática a programação entre HTML e CSS.

Materiais, Métodos e Ferramentas:

Ferramenta 1 – IDE.

Método 1 – Atividade Prática.

Materiais 1 – Acesso ao material complementar de aula.

Atividade Prática

HTML e CSS são duas das principais linguagens usadas para criar páginas da web. O HTML fornece a estrutura do conteúdo, enquanto o CSS é usado para estilizar esse conteúdo, definindo cores, fontes, tamanhos e layouts. Nesta atividade, você irá criar uma página HTML e aplicar estilo básico com CSS.

Tarefa:

Crie uma página HTML que exiba a frase “Bem-vindo ao mundo da programação!” e estilize-a usando CSS da seguinte maneira:

A frase deve estar centralizada na página.

A cor do texto deve ser azul.

O tamanho da fonte deve ser 24px.

Publique o arquivo no navegador para verificar o resultado.

Passos a seguir:

1 - Crie um arquivo HTML com a extensão .html (por exemplo, pagina_estilizada.html).

2 - Adicione a estrutura básica de um documento HTML.

3 - Escreva a frase dentro da tag <body>.

4 - Inclua um estilo CSS dentro da tag <style> para centralizar o texto, mudar a cor e o tamanho da fonte.

5 - Abra o arquivo no navegador para visualizar o resultado.

Gabarito Atividade Prática

Passo 1:

Você deve criar um arquivo HTML com a seguinte estrutura básica:

html

Copiar código

```
<!DOCTYPE html>
```

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

```
<head>
```

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

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

```
<title>Página Estilizada</title>
```

```
<style>
```

```
body {
```

```
text-align: center;
```

```
        color: blue;

        font-size: 24px;

    }

</style>

</head>

<body>

    <p>Bem-vindo ao mundo da programação!</p>

</body>

</html>
```

Explicação:

- **<!DOCTYPE html>**: Informa ao navegador que o documento é HTML5.
- **<html lang="pt-BR">**: Define o idioma do documento como português.
- **<meta charset="UTF-8">**: Define a codificação de caracteres para UTF-8.
- **<style>**: A tag onde o CSS está definido diretamente dentro do documento HTML.
 - **text-align: center**: Centraliza o texto no corpo da página.
 - **color: blue**: Define a cor do texto como azul.
 - **font-size: 24px**: Define o tamanho da fonte como 24 pixels.
- **<p>**: A tag de parágrafo onde a frase é inserida.

Passo 2:

Salve o arquivo com a extensão **.html**, por exemplo, `pagina_estilizada.html`.

Passo 3:

Abra o arquivo no navegador. Se você seguiu os passos corretamente, a frase **“Bem-vindo ao mundo da programação!”** estará centralizada, com a cor azul e tamanho de fonte 24px.

Erros Comuns:

- Esquecer de colocar o CSS dentro da tag `<style>`: Certifique-se de que as regras CSS estão entre as tags `<style></style>` no `<head>`.
- Esquecer de salvar o arquivo como `.html`: Isso é necessário para que o navegador reconheça o documento como uma página web.
- Não abrir o arquivo no navegador após salvar: Você deve visualizar o arquivo no navegador para verificar a aparência final.

Essa atividade prática ensina a combinar HTML e CSS, aplicando estilos básicos para centralizar, colorir e alterar o tamanho de texto em uma página web.

Ir para exercício