



INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 04 – HTML/CSS

O QUE IREMOS APRENDER

01

REVISÃO

02

PRINCÍPIOS DO CSS

03

SELETORES

04

ESPECIFICIDADE

05

PROPRIEDADE

Revisão

HTML (HyperText Markup Language) é a espinha dorsal da criação de páginas web. Entre seus diversos elementos, as tabelas são fundamentais para a organização e apresentação de dados em formato de grade.

Uma tabela em HTML é criada usando a tag `<table>`, e cada linha é definida dentro de uma tag `<tr>`. Dentro das linhas, podemos ter células de cabeçalho (`<th>`) ou células de dados (`<td>`).

O agrupamento de conteúdo em HTML é realizado usando tags como `<div>` e ``. `<div>` é usado para agrupar blocos de conteúdo, enquanto `` é mais adequado para agrupar conteúdo inline. Esses elementos são essenciais para a estruturação da página, permitindo a aplicação de estilos CSS e scripts JavaScript de maneira eficiente.

Revisão

Formulários em HTML

Formulários são ferramentas poderosas para interação com o usuário, permitindo a coleta de dados e informações.

Estrutura Básica: Um formulário é definido pela tag `<form>` e pode conter diversos tipos de campos de entrada.

Campos de Entrada Comuns:

- Texto (`<input type="text">`): Para entrada de texto simples.
- Email (`<input type="email">`): Campo específico para endereços de email.
- Senha (`<input type="password">`): Para entrada de senhas.
- Botões de Rádio (`<input type="radio">`): Para escolhas múltiplas, mas com seleção única.
- Caixas de Seleção (`<input type="checkbox">`): Para seleções múltiplas.
- Botões (`<input type="submit">`, `<button>`): Para enviar o formulário.
- Áreas de Texto (`<textarea>`): Para entrada de texto de múltiplas linhas.
- Label (Rótulo): usado para nomear ou marcar elementos para facilitar a organização



ATIVIDADE PRÁTICA

Crie uma tabela HTML simples com 3 colunas e 2 linhas. Na primeira linha, use células de cabeçalho para nomear as colunas como "Nome", "Idade" e "Cidade". Na segunda linha, adicione alguns dados fictícios correspondentes a essas colunas.

Princípios do CSS

CSS, ou Cascading Style Sheets, é uma linguagem de estilo usada para descrever a aparência de um documento escrito em HTML ou XML.

Em outras palavras, o CSS é responsável pela apresentação visual do conteúdo na web.

Ele controla como os elementos HTML são exibidos na tela, no papel, na fala ou em outras mídias. Isso inclui layout, cores, fontes e até mesmo animações.



O que é CSS

CSS é uma ferramenta poderosa que capacita os desenvolvedores a criar sites atrativos e consistentes.

Sem o CSS, a web seria um lugar muito mais monótono e menos dinâmico.

Ele possibilita que os desenvolvedores controlem a aparência de várias páginas simultaneamente, o que pode economizar bastante tempo.



Tipos de CSS

Existem três tipos principais de CSS:

- Inline
- Interno
- Externo



Tipos de CSS - CSS InLine

CSS Inline:

É usado diretamente na tag HTML individual. Isso é particularmente útil para estilizações únicas que não serão reutilizadas em outro lugar. Isso significa, por exemplo, que se você quiser modificar a aparência de um único elemento em sua página, você pode usar CSS Inline. No entanto, usar este tipo de CSS pode rapidamente tornar o código difícil de ler e manter, especialmente para grandes documentos HTML.

Aqui está um exemplo de como usá-lo:

```
<p style="propriedade: valor;">Este é um parágrafo com CSS inline.</p>
```

Neste exemplo, "propriedade" poderia ser algo como color ou font-size, e "valor" poderia ser red ou 16px, respectivamente.

Tipos de CSS - CSS Interno

CSS Interno

É usado no cabeçalho da página HTML. Isso permite que os estilos sejam compartilhados por toda a página, mas não além dela.

Isso é útil quando você tem estilos específicos que se aplicam a apenas uma única página em seu site. Aqui está um exemplo de como usá-lo:

```
<head>  
<style>  
seletor {  
  propriedade: valor; }  
</style>  
</head>
```

Neste exemplo, "seletor" poderia ser uma tag HTML como p, div, h1, etc., "propriedade" poderia ser qualquer propriedade CSS como color, font-size, etc., e "valor" seria o valor que você deseja aplicar a essa propriedade.

Tipos de CSS - CSS Externo

CSS Externo

É usado em um arquivo separado que é então vinculado à página HTML.

Isso permite que os estilos sejam reutilizados em vários documentos, promovendo consistência e reduzindo a repetição. Isso é útil quando você tem muitas páginas que compartilham os mesmos estilos. Aqui está um exemplo de como usá-lo:

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

E no arquivo estilos.css:

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

Novamente, "seletor" poderia ser uma tag HTML, uma classe ou um ID, "propriedade" seria qualquer propriedade CSS, e "valor" seria o valor que você deseja aplicar a essa propriedade.

Sintaxe

A sintaxe do CSS é relativamente simples.

Consiste em um seletor e um bloco de declaração. O seletor aponta para o elemento HTML que você deseja estilizar.

O bloco de declaração contém uma ou mais declarações separadas por ponto e vírgula. Cada declaração inclui uma propriedade CSS e um valor associado.



Sintaxe

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

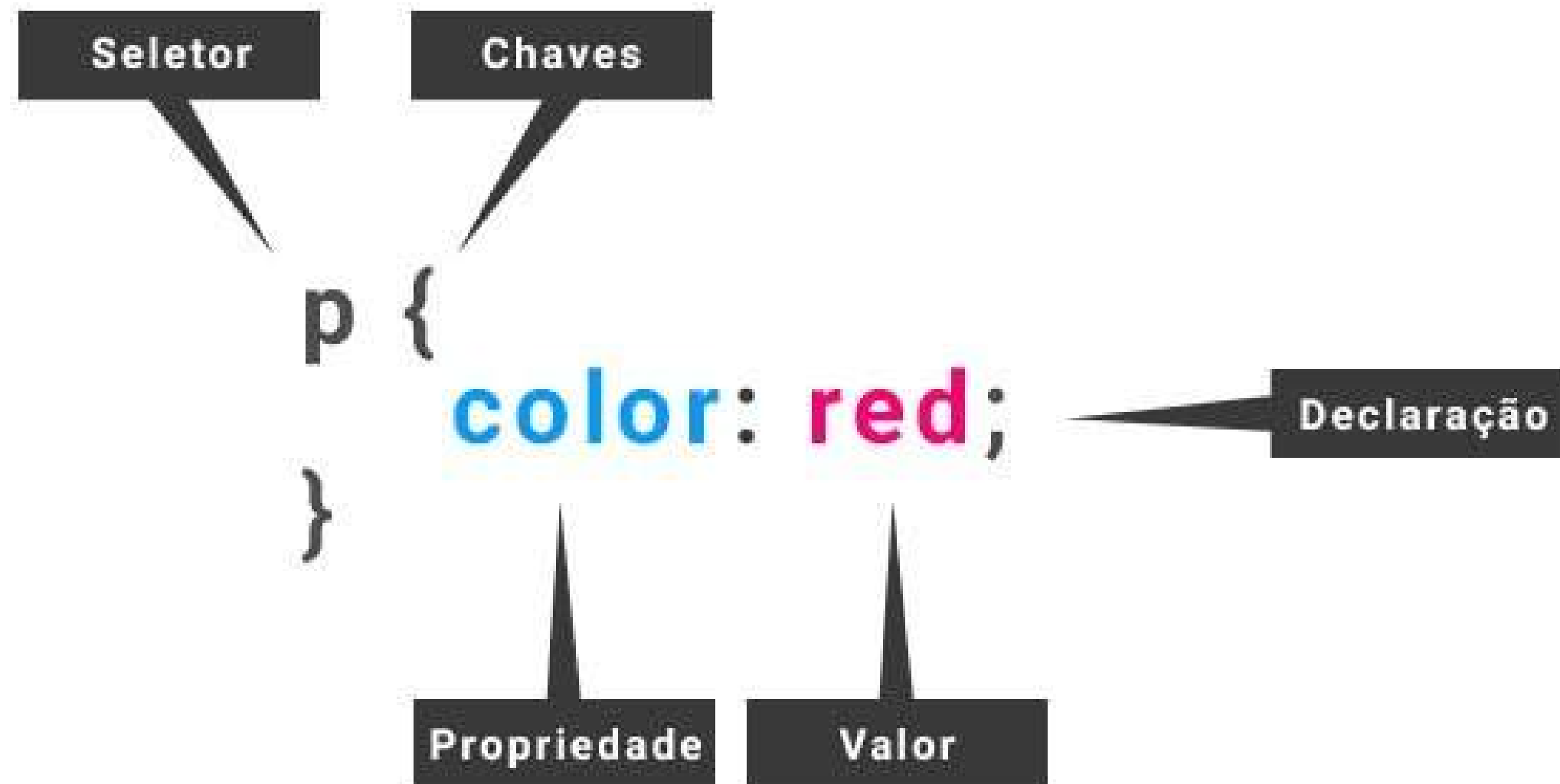
Neste exemplo, `seletor` pode ser qualquer seletor CSS válido, como um nome de tag HTML (por exemplo, 'div', 'p', 'a'), uma classe (indicada por um ponto, como '.container'), ou um ID (indicado por um símbolo de hashtag, como '#coisa').

`propriedade` é qualquer propriedade CSS válida, como 'color', 'font-size', 'width', 'background-color', etc.

`valor` é o valor que você deseja aplicar a essa propriedade. Por exemplo, se a propriedade for 'color', o valor pode ser uma cor como 'red', 'blue', '#FF0000', etc.



Sintaxe



Seletores simples

Os seletores de CSS permitem que você selecione e manipule elementos HTML. Existem vários tipos de seletores, incluindo seletores de tag, classe e ID.

- O seletor de tag seleciona elementos com base em seu nome de tag (por exemplo, p, div, h1).
- O seletor de classe seleciona elementos com base em sua classe (por exemplo, .intro, .header).
- O seletor de ID seleciona um elemento com base em seu ID único (por exemplo, #navbar).

A especificidade é um conceito importante em CSS que determina quais estilos se aplicam a um elemento quando há vários estilos em conflito. Regras mais específicas têm precedência sobre regras menos específicas.

Seletores simples

- O seletor de tag

```
div{  
  background-color:red;  
}
```

Essa estrutura aplicará a formatação à todas as tags div que existirem no seu documento HTML, sem exceção.



Seletores simples

- O seletor de classe

```
.conteudo{  
  background-color:red;  
}
```

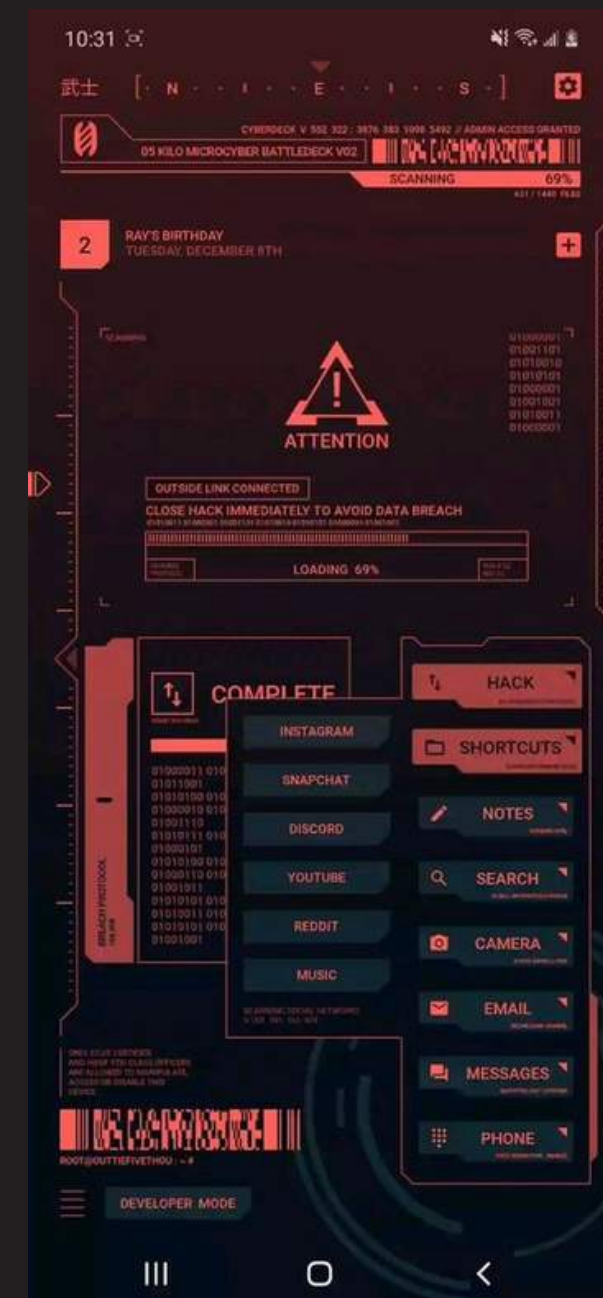
Assim, a formatação definida nessa classe será aplicada à todas as tags que possuírem essa classe, sendo elas do mesmo tipo, ou não.

Seletores simples

- O seletor de id

```
#conteudo{  
  background-color:red;  
}
```

O estilo mostrado no exemplo será aplicado à tag que possui o id conteudo, não importando qual tag seja essa.



ATIVIDADE PRÁTICA

Estilizando uma Página Simples com CSS

Objetivo: Utilizar seletores simples de CSS para aplicar estilos específicos a uma página HTML.

Estilizar o Cabeçalho (<header>):

Defina a cor de fundo do cabeçalho para cinza claro.

Centralize o texto do título (<h1>).

Estilizar a Navegação (<nav>):

Remova os marcadores da lista () no menu de navegação.

Faça com que cada item da lista () seja exibido em linha (horizontalmente).

Estilizar os Artigos (<article>):

Adicione uma borda sólida de 1px e cor preta em torno de cada artigo.

Defina uma margem de 10px para cada artigo.

Estilizar o Rodapé (<footer>):

Defina a cor de fundo do rodapé para cinza escuro.

Altere a cor do texto para branco.

```
<!DOCTYPE html>
<html>
<head>
  <title>Exercício de CSS</title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
  <header>
    <h1>Bem-vindo ao Meu Site</h1>
  </header>
  <nav>
    <ul>
      <li><a href="#">Início</a></li>
      <li><a href="#">Sobre</a></li>
      <li><a href="#">Contato</a></li>
    </ul>
  </nav>
  <section>
    <article>
      <h2>Artigo 1</h2>
      <p>Este é o primeiro artigo.</p>
    </article>
    <article>
      <h2>Artigo 2</h2>
      <p>Este é o segundo artigo.</p>
    </article>
  </section>
  <footer>
    <p>Direitos autorais © 2023</p>
  </footer>
</body>
</html>
```

CÓDIGO BASE HTML DO EXERCÍCIO

```
header {  
  background-color: lightgray;  
  text-align: center;  
}
```

```
nav ul {  
  list-style-type: none;  
  padding: 0;  
}
```

```
nav li {  
  display: inline;  
  margin-right: 10px;  
}
```

```
article {  
  border: 1px solid black;  
  margin: 10px;  
}
```

```
footer {  
  background-color: darkgray;  
  color: white;  
  text-align: center;  
}
```

CÓDIGO BASE CSS DO EXERCÍCIO

Crie um arquivo HTML e outro CSS com o conteúdo fornecido.

Salve os arquivos e abra o arquivo HTML em um navegador para ver o resultado.

Modifique o CSS conforme as tarefas descritas e observe as mudanças no navegador.

Seletores combinados

Os seletores combinados permitem que você seja mais específico com as suas regras CSS, atingindo elementos HTML de maneira mais precisa.

Eles são uma combinação de dois ou mais seletores diferentes que identificam elementos baseados numa relação específica entre eles.



Seletores combinados

Existem quatro tipos principais de seletores combinados:

- Descendente (espaço): Seleciona todos os elementos que são descendentes de um elemento específico. Por exemplo, `div p` selecionará todos os elementos `p` que estão dentro de um `div`.
- Filho (`>`): Seleciona todos os elementos que são filhos diretos de um elemento específico. Por exemplo, `div > p` selecionará todos os elementos `p` que são filhos diretos de um `div`.

Adjacente direto (`+`): Seleciona o elemento que está imediatamente após um elemento específico. Por exemplo, `div + p` selecionará o elemento `p` que está imediatamente após o `div`.

- Irmão geral (`~`): Seleciona todos os elementos que são irmãos de um elemento específico. Por exemplo, `div ~ p` selecionará todos os elementos `p` que são irmãos de um `div`.

Esses seletores oferecem maior flexibilidade e controle na aplicação de estilos CSS, permitindo que você crie designs mais complexos e detalhados.

Seletores combinados

Descendente (espaço):

```
p span{  
  background-color:red;  
}
```

O estilo deste exemplo se aplicará à todas as tags span que estejam dentro de uma tag p, mesmo que a tag span não seja filha direta da tag p.



Seletores combinados

Filho (>):

```
p > span{  
  background-color:red;  
}
```

O estilo deste exemplo se aplicará apenas às tags span que forem filhas diretas de uma tag p.



Seletores combinados

Adjacente direto (+):

```
p + span{  
  background-color: red;  
}
```

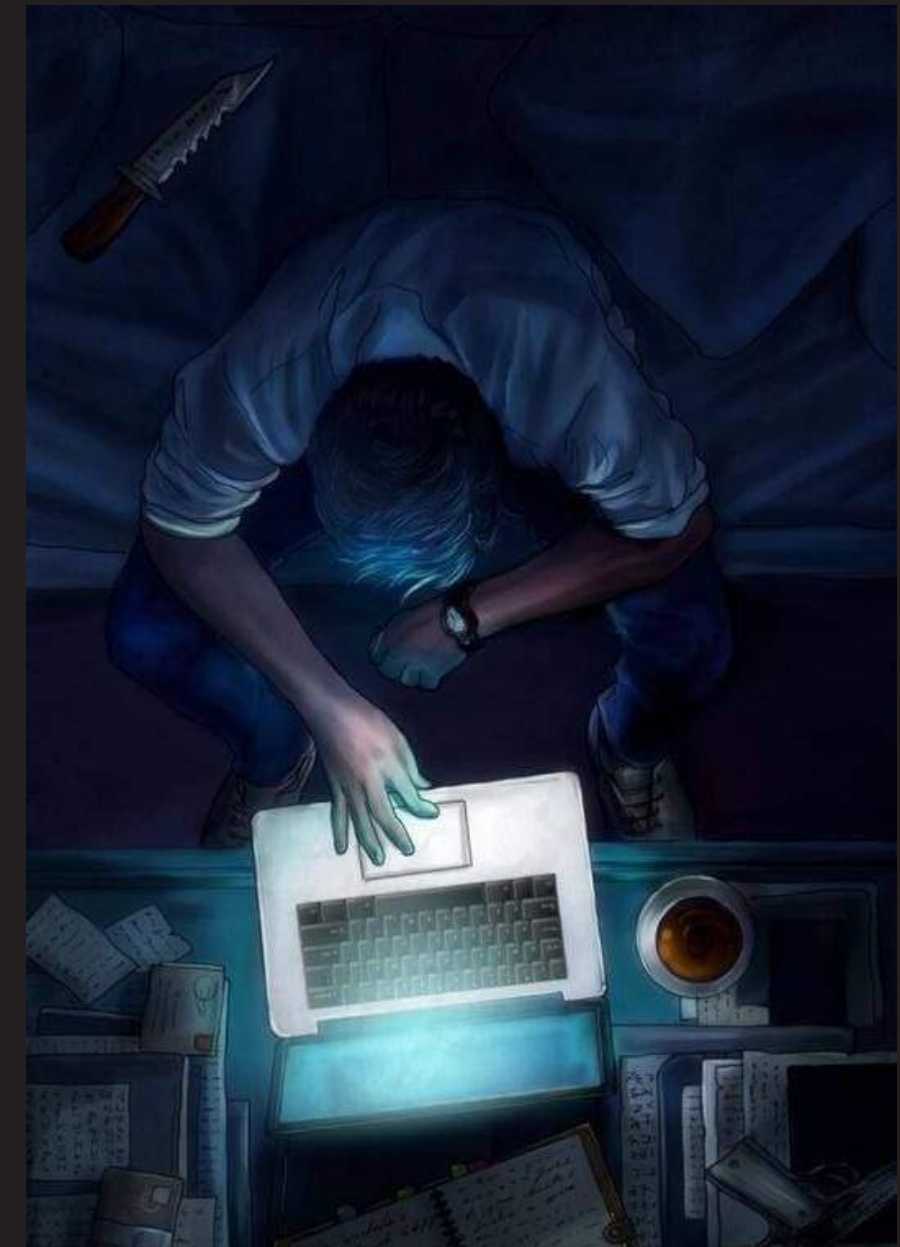
O estilo deste exemplo será aplicado à todas as tags span que forem imediatamente precedidas de uma tag p.

Seletores combinados

Irmão geral (~):

```
p ~ span{  
  background-color:red;  
}
```

O estilo deste exemplo será aplicado à todas as tags span que forem precedidas de uma tag p dentro de um mesmo elemento pai.



ATIVIDADE PRÁTICA

Utilizando Seletores Combinados em CSS

Objetivo: Usar seletores combinados de CSS para aplicar estilos específicos a elementos em uma estrutura HTML simples.

Estilizar Parágrafos Dentro do Container:

Use um seletor descendente para alterar a cor do texto de todos os parágrafos (<p>) dentro do .container para azul.

Estilizar o Primeiro Parágrafo Dentro de Cada Container:

Use o seletor de filho direto (>) para alterar a cor de fundo do primeiro parágrafo dentro de .container e .sub-container para amarelo claro.

Estilizar Parágrafos no Sub-Container:

Use um seletor descendente para alterar a fonte dos parágrafos dentro de .sub-container para itálico.

```
<!DOCTYPE html>
<html>
<head>
  <title>Exercício de Seletores Combinados</title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
  <div class="container">
    <h1>Título Principal</h1>
    <p>Este é um parágrafo no container.</p>
    <p>Outro parágrafo no mesmo container.</p>
    <div class="sub-container">
      <p>Parágrafo no sub-container.</p>
    </div>
  </div>
  <p>Parágrafo fora do container.</p>
</body>
</html>
```

CÓDIGO BASE HTML DO EXERCÍCIO


```
.container p {  
  color: blue;  
}
```

```
.container > p:first-child, .sub-container > p:first-child {  
  background-color: lightyellow;  
}
```

```
.sub-container p {  
  font-style: italic;  
}
```

CÓDIGO BASE CSS DO EXERCÍCIO

Crie um arquivo HTML e outro CSS com o conteúdo fornecido.

Salve os arquivos e abra o arquivo HTML em um navegador para ver o resultado.

Modifique o CSS conforme as tarefas descritas e observe as mudanças no navegador.

Especificidade

A especificidade é um conceito fundamental em CSS que determina quais estilos são aplicados a um elemento.

Em situações em que diferentes regras podem se aplicar ao mesmo elemento, a regra mais específica terá precedência.

A especificidade é calculada com base no tipo de seletor usado na regra CSS:

Seletor de tipo ou seletor universal tem especificidade 0,0,0,1.

- Seletor de classe, seletor de atributo e seletor pseudo-classe tem especificidade 0,0,1,0.

Seletor de ID tem especificidade 0,1,0,0.

Estilo inline tem especificidade 1,0,0,0.

Especificidade

Se a especificidade for a mesma, a última regra declarada no CSS será aplicada. Se uma declaração `!important` for usada, ela terá a maior especificidade, independentemente dos outros seletores utilizados na regra. No entanto, o uso de `!important` deve ser evitado sempre que possível, pois pode tornar o código difícil de gerenciar e levar a resultados inesperados.

Entender a especificidade em CSS é essencial para escrever código eficiente e gerenciável, e para garantir que seus estilos sejam aplicados como esperado.

Especificidade

Id é maior que classe

Bem, todos sabemos que o ID pode ser usado uma vez em uma estrutura de documento. No entanto, o ID em CSS é um exército de um homem que pode derrotar 1000 classes. Quando você tem ID (s) no seu seletor, o número de classes se torna irrelevante.

```
<p id="idTeste" class="barra1 barra 2 barra3 barra4">Um ID contra quatro classes</p>
```

```
#idTeste{Resultado: A cor do parágrafo é "red" (vermelho).  
color: red;
```

```
}Embora tenhamos quatro classes em um parágrafo,  
mas um único ID o precedeu.
```

```
.barra1.barra2.barra3.barra4 {O navegador não respeitará a posição do seletor,  
color: steelblue;seja ela mais recente ou em algum lugar entre as  
}linhas superiores quando for ID.
```

Especificidade

Quem será o vencedor?

E se dois seletores se aplicarem ao mesmo elemento?
Vamos dar um exemplo e depois prosseguir.

```
<p class="barra1 barra2 barra3 barr4">Temos quatro classes</p>
.barra1.barra2.barra3.barr4 {
color: green;
}
```

A cor do parágrafo é "green" (Verde)

```
p.barra3.barra4 {
color: #2e8dcd;
}
```

O vencedor é aquele com especificidade sempre maior. Neste código, a especificidade do primeiro seletor é 0-4-0, enquanto a última regra tem especificidade 0-2-1. Não se preocupe com esse valor de especificidade; já já explicou com mais detalhes. Neste ponto, lembre-se de que a esquerda está sempre certa.

Propriedades

As propriedades de CSS são usadas para alterar a aparência dos elementos. Por exemplo, você pode alterar a fonte, o tamanho do texto, a cor do texto, o alinhamento do texto, a altura e a largura dos elementos e muito mais..



Propriedades

Propriedades Básicas do CSS

color: Esta propriedade é usada para alterar a cor do texto de um elemento. Você pode usar nomes de cores, valores de cores hexadecimais ou valores de cores RGB.

font-size: Esta propriedade é usada para alterar o tamanho da fonte de um elemento. Você pode usar unidades absolutas como pixels (px) ou unidades relativas como em, que são relativas ao tamanho da fonte do elemento pai.

font-family: Esta propriedade é usada para alterar a fonte de um elemento. Você pode especificar uma lista de fontes separadas por vírgulas, que serão utilizadas em ordem, caso a anterior não esteja disponível.

font-weight: Esta propriedade é usada para alterar a espessura da fonte de um elemento. Você pode usar palavras-chave como 'normal' ou 'bold', ou valores numéricos entre 100 e 900.

font-style: Esta propriedade é usada para alterar o estilo da fonte de um elemento. Você pode usar palavras-chave como 'normal', 'italic' ou 'oblique'.

Propriedades

`text-align`: Esta propriedade é usada para alterar o alinhamento do texto de um elemento. Você pode usar palavras-chave como 'left', 'right', 'center' ou 'justify'.

`text-decoration`: Esta propriedade é usada para adicionar ou remover decorações de um texto, como sublinhados, sobrescritos ou tachados. Você pode usar palavras-chave como 'underline', 'overline', 'line-through' ou 'none'.

`width` e `height`: Estas propriedades são usadas para alterar a largura e a altura de um elemento, respectivamente. Você pode usar unidades absolutas como pixels (px) ou unidades relativas como porcentagens (%).

`background-color`: Esta propriedade é usada para alterar a cor de fundo de um elemento. Você pode usar nomes de cores, valores de cores hexadecimais ou valores de cores RGB.

`background-image`: Esta propriedade é usada para adicionar uma imagem de fundo a um elemento. Você pode usar o valor 'url()' para especificar o caminho da imagem.

ATIVIDADE PRÁTICA

Objetivo: Praticar o uso de diferentes propriedades CSS para estilizar uma página HTML simples.

Centralizar o Título (<h1>):

Use a propriedade text-align para centralizar o título.

- Adicionar Sombra ao Texto Destacado:

Use a propriedade text-shadow para adicionar uma sombra ao texto com a classe .destaque

- Alterar o Espaçamento da Linha do Parágrafo:

Use a propriedade line-height para aumentar o espaçamento da linha no segundo parágrafo (<p>).

- Estilizar o Link:

Use as propriedades color e text-decoration para alterar a cor do link e remover o sublinhado, respectivamente.

ARQUIVO HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Exercício de Propriedades CSS</title>
  <link rel="stylesheet" type="text/css" href="estilos.css">
</head>
<body>
  <div id="bloco">
    <h1>Titulo Centralizado</h1>
    <p class="destaque">Texto destacado com sombra</p>
    <p>Parágrafo com espaçamento de linha.</p>
    <a href="#" class="link">Link Estilizado</a>
  </div>
</body>
</html>
```

ARQUIVO CSS

```
#bloco h1 {
  text-align: center;
}

.destaque {
  text-shadow: 2px 2px 2px gray;
}

#bloco p {
  line-height: 1.5;
}

.link {
  color: green;
  text-decoration: none;
}
```

Crie um arquivo HTML e outro CSS com o conteúdo fornecido. Salve os arquivos e abra o arquivo HTML em um navegador para ver o resultado. Modifique o CSS conforme as tarefas descritas e observe as mudanças no navegador.



PROJETO CCS

PARTE I

INÍCIO DO PROJETO

EM SALA DE AULA

SE LIGA NO CONTEÚDO DA PRÓXIMA AULA!

AVANÇADO CSS.
DISPLAY – INLINE – BLOCK – INLINE BLOCK

The logo consists of the letters 'IN' in a white, bold, sans-serif font, centered within a solid red square. In the top right corner of the overall image, there are several thin, concentric red curved lines.

INFINITY SCHOOL
VISUAL ART CREATIVE CENTER

CSS

Em CSS, a propriedade `display` é crucial para determinar como um elemento é exibido na página. Existem vários valores para essa propriedade, mas três dos mais comuns são `inline`, `block` e `inline-block`.

- **Inline:** Elementos com `display: inline`; não começam em uma nova linha e só ocupam o espaço necessário para o conteúdo. Exemplos típicos de elementos inline são ``, `<a>`, e ``. Eles não respeitam propriedades de largura (`width`) e altura (`height`), e as margens verticais e preenchimentos não causam impacto no layout.
- **Block:** Elementos com `display: block`; começam em uma nova linha e ocupam toda a largura disponível, estendendo-se de uma borda lateral da tela à outra. Exemplos comuns incluem `<div>`, `<p>`, e `<h1>`. Eles respeitam todas as propriedades de largura, altura, margem e preenchimento.
- **Inline-Block:** Combina características dos dois anteriores. Elementos com `display: inline-block`; são colocados na mesma linha quando possível, mas também respeitam propriedades de largura e altura. Isso é útil para elementos que devem se comportar como blocos, mas se alinhar horizontalmente, como botões em uma barra de navegação.

```
<!DOCTYPE html>
<html>
<head>
  <title>Página com Figura</title>
<style>
.imagem-centro {
display: block;
margin-left: auto;
margin-right: auto;
width: 50%;
}
</style>
</head>
<body>
  <h1>Exemplo de Figura</h1>
  
  <p>Descrição sobre a imagem
acima.</p>
</body>
</html>
```

HTML:

``: Este é o elemento de imagem. Substitua "url_da_imagem.jpg" pelo URL da imagem que deseja exibir. O atributo alt é usado para descrever a imagem, importante para acessibilidade e SEO.

CSS:

- .imagem-centro: Esta classe CSS é aplicada à imagem. Ela centraliza a imagem na página e define sua largura para 50% do contêiner pai (neste caso, o corpo da página).

Notas:

Certifique-se de substituir "url_da_imagem.jpg" pelo caminho correto da imagem que você deseja exibir.





INFINITY SCHOOL

VISUAL ART CREATIVE CENTER

AULA 04 – HTML/CSS