

TTI107 – Seletor CSS

1. Style e CSS

Até agora você estava vendo sobre o elemento **style** da tag, que pode modificar a forma de como o elemento vai ser exibido, e descobriu que pode declará-lo tanto no corpo da tag (inline), no elemento head (internal) e/ou em um arquivo separado **.css** (external).

Apenas uma rápida lembrada, um arquivo **.css** deve ser semelhante a isso:

```
body {}
h1 {
  color: blue;
  background-color: white;
  border-style: solid;
  border-radius: 10px;
  text-align: center;
  font-family:'Courier New', Courier, monospace;
}
p {
  text-align: justify;
  border-left: 10px;
  border-left-style: solid;
  border-left-color: blue;
}
```

CódigoCSS1.1

Você já deve ter percebido que o mesmo nome de tag que usamos no arquivo **.html**, também usamos para definir o elemento que queremos mudar a aparência. Então se não gostamos do estilo “padrão do html”, ou se quisermos mudar para deixar nossa página mais bonita, devemos mudar seu estilo (**style**). Logo passamos de algo assim:

Página super legal

Página muito bonita e estilizada.

Imagem1.1

Para algo assim:

Página super legal

■ Página muito bonita e estilizada.

Imagem1.2

Mas, e se quisermos vários elementos diferentes? Cada título com uma formatação diferente de acordo com o tópico, ou parágrafos diferentes um do outro, para destacar um que seja mais importante que o outro. Enfim, continuar usando os mesmos elementos, porém com estilos diferentes?

2. Seletores CSS

No caso dos títulos/tópicos pode-se sugerir mudar o estilo de h2 para ficar parecido com h1 e depois mudar uma coisa ou outra, mas e se eu tiver mais de 6 títulos/tópicos com estilos diferentes? Ou quiser usar um parágrafo com texto grande e outro com texto pequeno?

2.1 Seletor CSS por tipo

Como citado antes, já deve ter reparado que usamos o nome da tag no arquivo `.css` para definir o elemento que queremos alterar sua aparência, vamos ver mais afundo alguns aspectos interessantes antes de avançar no assunto.

Já percebeu que mudar o elemento body acaba por afetar a página toda? E os elementos presentes nela?

Faça um teste, usando a tag div (`<div></div>`), dentro do elemento body, crie um elemento parágrafo (`<p></p>`) dentro do elemento div e um parágrafo fora do elemento div, no arquivo `.css`, altere o estilo do elemento div. Deve ter acontecido algo como isso:

Parágrafo fora do elemento div

Parágrafo dentro do elemento div

Imagem2.1.1

```
body {}
div {
  color: blue;
  border-style: solid;
  border-color: red;
}
p {}
```

Mesmo não declarando uma mudança de estilo do elemento parágrafo, ele acabou por mudar o estilo dele quando este está dentro do elemento div, o mesmo ocorre ao se declarar uma mudança no elemento body, todos os elementos dentro de body vão sofrer uma mudança. Note que títulos/tópicos (e qualquer outro elemento) vão sofrer alteração por estar dentro de outro elemento que teve seu estilo trocado.

[...]
<pre> body { div { color: blue; border-style: solid; border-color: red; } p { color: black; border-style: solid; border-color: black; } </pre>

Usando o cenário anterior (CódigoHTML2.1.1), qual será o efeito de declarar um estilo para o parágrafo?

O resultado é semelhante ao se mudar o elemento body para um efeito mais geral na página, e mudar o parágrafo para um estilo mais específico.

Parágrafo fora do elemento div
<div> <p>titulo</p> <p>Parágrafo dentro do elemento div</p> </div>

Há outros meios de seleção por tipo que usa combinações, mas esses métodos fogem do escopo do curso, para saber mais sobre eles: [CSS Combinators \(w3schools.com\)](https://www.w3schools.com/css/css-combinators.php)

2.2 Seletor CSS por classe

O tópico anterior abordou um assunto intuitivo: mudar estilo conforme o nome da tag e que uma tag dentro de outra herda seus estilos, como a tag pai (mais externa) e a tag filha (tag interna). Mas apenas isso não nos dá uma liberdade de mudar o estilo como o **css inline**, para poder fazer uma seleção que não se prenda ao tipo e ainda assim pode ser usada em outros elementos é o atributo **classe**.

O elemento classe é declarado no arquivo **.css** como **‘.nomeClasse’**, e no arquivo **.html** é declarado como o elemento **style** na tag: **‘<p class="nomeClasse"></p>’**.

```
body {  
  .bonito {  
    color: blueviolet;  
    font-family: 'Courier New', Courier, monospace;  
  }  
}
```

```
<body>  
  <h1 class="bonito">Um título bonito</h1>  
  <p class="bonito">Um texto bonito</p>  
  <h1>Um título normal</h1>  
  <p>Um texto normal</p>  
</body>
```

CódigoCSS2.2.1

CódigoHTML2.2.1

Um título bonito

Um texto bonito

Um título normal

Um texto normal

Imagem2.2.1

2.3 Seletor CSS por ID

Usar uma classe pode ajudar bastante na hora de definir os estilos e usá-los, mas é possível usar algo ainda mais específico que o tipo de uma tag e sua classe, que é o ID. O ID é declarado no arquivo **.css** como **‘#nomeId’**, e no arquivo **.html** é declarado no nome da tag como o **style**: **‘<p id="nomeId"></p>’**.

```
.bonito {  
  color: blueviolet;  
  font-family: 'Courier New', Courier, monospace;  
}  
#especial {  
  color: aliceblue;  
  background-color: darkslategray;
```

```
<body>
  <h1 class="bonito">Um título bonito</h1>
  <p id="especial" class="bonito">Um texto bonito e especial</p>
  <h1>Um título normal</h1>
  <p>Um texto normal</p>
</body>
```

Um título bonito

Um texto bonito e especial

Um título normal

Um texto normal

Imagem2.3.1

Como pode ver no texto bonito e especial, é possível usar classe e id junto, e isso se aplica aos tipos também. Note que a cor de fonte que permanece é branca, declarado no id, porém a família da fonte (font-family) não é a padrão, é a declarada na classe.

Como pode perceber, há um grau de importância entre os seletores, aonde o que for mais específico sobrescreve os outros, e o que não for declarado no mais específico é buscado no menos específico, até chegar no padrão.

3 Maneiras de agrupar

Os seletores CSS podem ser agrupados para ter parâmetros mais específicos ou apenas juntar aqueles parâmetros que tem um estilo semelhante.

No caso de juntar seletores por ter um estilo semelhante, pode nos ajudar a economizar tempo e linhas, além de deixar o arquivo **.css** mais organizado.

```
body {}
.bonito, h1 {
  color: blueviolet;
  font-family: 'Courier New', Courier, monospace;
}
#textoEspecial, #tituloEspecial {
  color: aliceblue;
  background-color: darkslategray;
```

Como pode ver é possível agrupar qualquer tipo de seletor, basta separá-los por vírgula.

Também é possível juntar esses seletores para ter estilos mais específicos, pode-se limitar o funcionamento de uma classe/id apenas para certo tipo, ou ainda ter um comportamento polimórfico por parte das classes.

```
.centro#borda {  
  border-style: dotted;  
  border-color: gold;  
  text-align: left;  
  color: black;  
}  
p.centro {  
  text-align: center;  
  color: red;  
}  
h1.centro {  
  text-align: center;  
  color: blue;  
}
```

Algo que pode ser feito também que não fica no arquivo **.css** é selecionar mais de uma classe para os elementos no arquivo **.html**, assim pode fazer classes que mudam algo bem específico e juntar elas quando declarar o elemento.

```
<h1 class="centro bonito">Um título bonito</h1>  
<p class="centro textoGrande bordaComum" id="perigo"> Cuidado</p>
```