



Universidade Federal de Santa Catarina
Curso de Tecnologias da Informação e Comunicação
Desenvolvimento de Sistemas Web
Professor: Adriano de Oliveira

Desenvolvimento de Sistemas Web

Prof.: Adriano de Oliveira
email: adriano.inovar@gmail.com
Turma: 05652 - 3-2020-2 e 5-1830-2

CSS

Imagem de Fundo

- ✓ A propriedade background-image permite indicar um arquivo de imagem para ser exibido ao fundo do elemento. Por exemplo:

```
body {  
    background-image: url("../images/ufsc.png");  
}
```

- ✓ Com essa declaração, o navegador vai requisitar um arquivo ufsc.png, que deve estar na pasta images no mesmo nível da pasta do arquivo CSS.

Seletor de Id


- ✓ É possível aplicar propriedades visuais a um elemento selecionado pelo valor de seu atributo id. Para isso, o seletor deve iniciar com o caractere “#” seguido do valor correspondente.

```
#titulo {  
    background-color: ☐ black;  
    color: ☐ white;  
    text-align: center;  
}
```

Seletor de Atributos

- ✓ Além dos seletores de tag, classe e id existe mais uma série de seletores avançados do CSS.
- ✓ Um dos seletores CSS mais versáteis é o seletor de atributo, com ele podemos verificar a presença ou valor de um atributo para selecioná-lo.

Seletor de Atributos

```
input[value] {  
  color:  #cc0000;  
}
```

```
<input type="text" value="Seletores CSS">
```



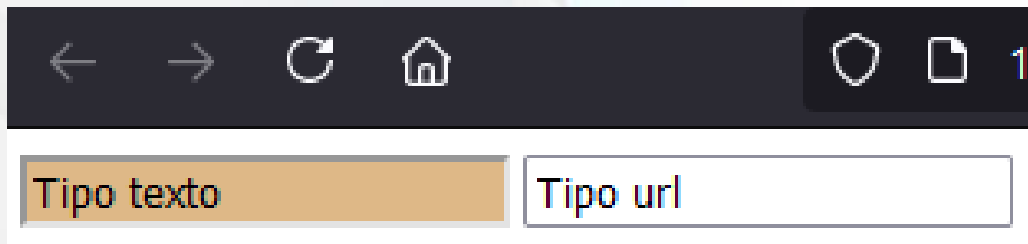
Seletores CSS

Seletor de Atributos

- ✓ Também é possível verificar se o atributo tem um valor específico:

```
input[type="text"] {  
    background-color: #burlywood;  
}
```

```
<input type="text" value="Tipo texto">  
<input type="url" value="Tipo url">
```



← → ↻ 🏠 🔒 📄 12

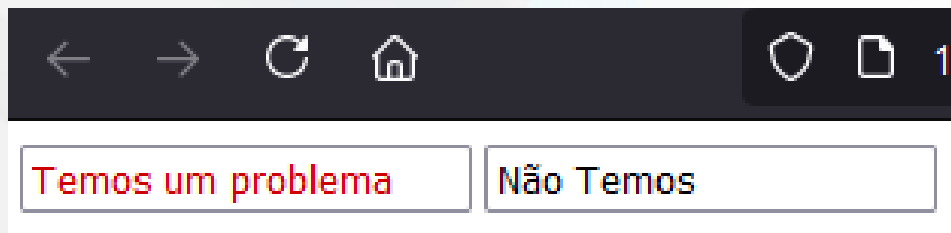
Tipo texto Tipo url

Seletor de Atributos

- ✓ Também é possível buscar por uma palavra específica no valor, não importando o valor completo do atributo. Por exemplo:

```
input[value~="problema"] {  
  color: #cc0000;  
}
```

```
<input type="text" value="Temos um problema">  
<input type="text" value="Não Temos">
```



← → ↺ 🏠 🔒 📄 12

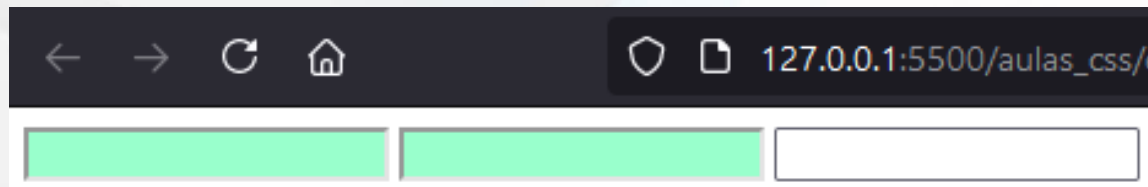
Temos um problema Não Temos

Seletor de Atributos

- ✓ Buscar por inputs com valor de "name" iniciando em "usuario".

```
input[name^="usuario"] {  
    background-color: #99ffcc;  
}
```

```
<input type="text" name="usuario01">  
<input type="text" name="usuario02">  
<input type="text" name="outrouusuario">
```

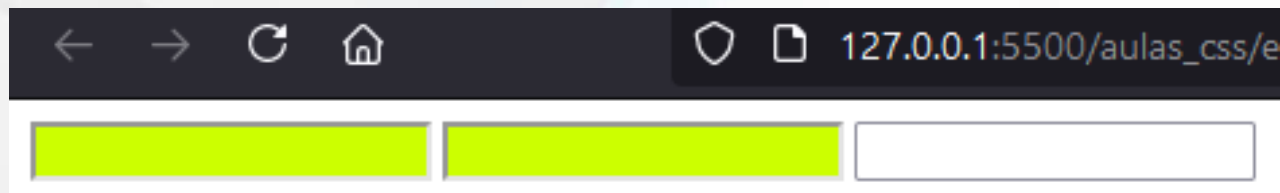


Seletor de Atributos

- ✓ Buscar por inputs com valor de "id" terminando em "teste".

```
input[id$="teste"] {  
    background-color: #ccff00;  
}
```

```
<input type="text" id="umteste">  
<input type="text" id="outroteste">  
<input type="text" id="testenovo">
```

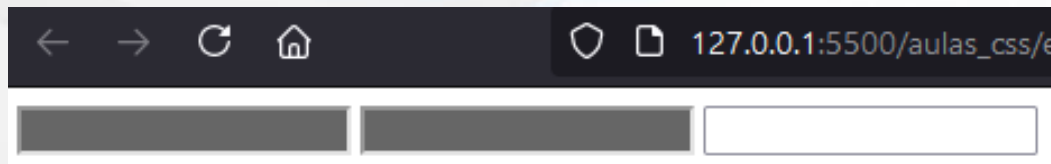


Seletor de Atributos

- ✓ Buscar por inputs com valor do atributo "name" contendo "tela" em qualquer posição.

```
input[name*="tela"] {  
    background-color: #666666;  
}
```

```
<input type="text" name="umatela">  
<input type="text" name="outrotela">  
<input type="text" name="essanao">
```



← → ↻ 🏠 127.0.0.1:5500/aulas_css/e

Seletor de Irmão

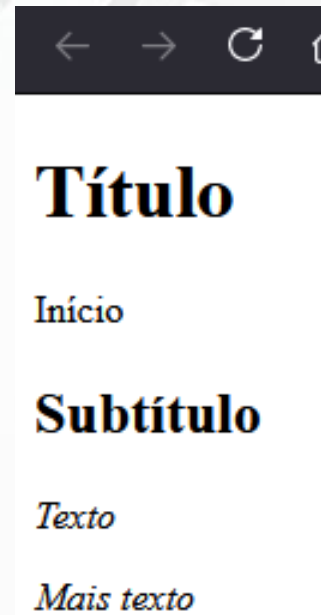
- ✓ Veja o seguinte HTML, que simula um texto com vários parágrafos, títulos e subtítulos no meio do documento:

```
<div>  
  <h1>Título</h1>  
  <p>Início</p>  
  
  <h2>Subtítulo</h2>  
  <p>Texto</p>  
  <p>Mais texto</p>  
</div>
```


Seletor de Irmão

- ✓ Isso indica que será selecionado todos os p que foram precedidos por algum h2 e estão sob a mesma tag pai.

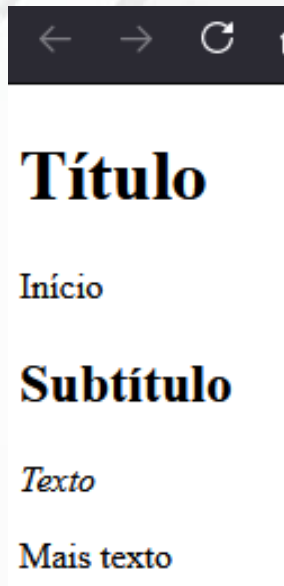
```
h2 ~ p {  
    font-style: italic;  
}
```



Seletor de Irmão Adjacente

- ✓ Será selecionado apenas o parágrafo imediatamente seguinte ao h2.

```
h2 + p {  
    font-style: italic;  
}
```



Negação

- ✓ Observe o seguinte o HTML com vários parágrafos simples:

```
<p>Texto</p>  
<p>Outro texto</p>  
<p>Texto especial</p>  
<p>Mais texto</p>
```

Negação

- ✓ Para fazer todos os parágrafos de cor cinza, exceto o que tem o texto especial. Precisamos destacá-lo de alguma forma no HTML para depois selecioná-lo no CSS. Uma classe ou ID resolve:

```
<p>Texto</p>  
<p>Outro texto</p>  
<p class="especial">Texto especial</p>  
<p>Mais texto</p>
```

Negação

```
p:not(.especial) {  
  color: gray;  
}
```



Texto

Outro texto

Texto especial

Mais texto

Pseudo-classe

- ✓ Pegue o seguinte HTML de uma lista de elementos, se quisermos estilizar elementos específicos dessa lista?

```
<ul>  
  <li>Primeiro item</li>  
  <li>Segundo item</li>  
  <li>Terceiro item</li>  
  <li>Quarto item</li>  
</ul>
```


Pseudo-classe

- ✓ O CSS tem um recurso chamado de pseudo-classes que são como classes CSS já pré-definidas.
- ✓ É como se o navegador já colocasse certas classes por padrão em certos elementos, cobrindo situações comuns como essa de selecionar o primeiro ou o último elemento.

Pseudo-classe

- ✓ Há duas pseudo-classes do CSS que representam exatamente o primeiro elemento filho de outro (first-child) e o último elemento filho (last-child).
- ✓ Essas classes já estão definidas, não precisam ser especificadas no HTML.

Pseudo-classe

```
li:first-child {  
    color: ■ red;  
}  
  
li:last-child {  
    color: ■ blue;  
}
```



- Primeiro item
- Segundo item
- Terceiro item
- Quarto item

Pseudo-classe de estados

- ✓ Para mudar a cor de foco de um link, o CSS possui a pseudo-classe `:hover` que representam estados dos elementos.
- ✓ É como se o navegador aplicasse uma classe chamada `hover` automaticamente quando o usuário passa o mouse em cima do elemento e depois retirasse a classe quando ele sai.

Pseudo-classe de estados

```
/*seleciona o link no exato momento em que passamos o mouse por cima dele*/  
a:hover {  
    background-color: #FF00FF;  
}  
  
/*seleciona todas as âncoras que têm o atributo "href", ou seja, links*/  
a:link {  
    background-color: #FF0000;  
}  
  
/* seleciona todos os links cujo valor de "href" é um endereço já visitado */  
a:visited {  
    background-color: #00FF00;  
}  
  
/*seleciona o link no exato momento em que clicamos nele */  
a:active {  
    background-color: #0000FF;  
}
```


Pseudo-classe de estados

```
<a href="http://www.ufsc.br">  
  Visite o site institucinal da UFSC  
</a>
```



Visite o site institucinal da UFSC



Visite o site institucinal da UFSC

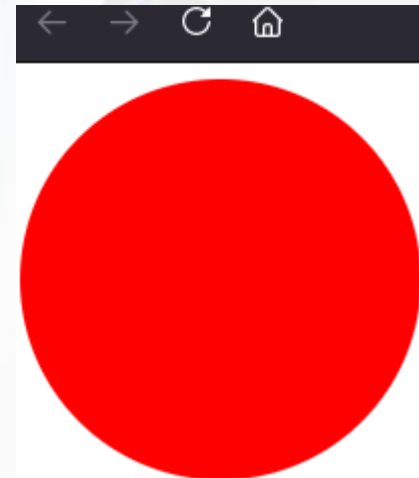


Visite o site institucinal da UFSC

Border-Radius

- ✓ Uma das novidades mais celebradas do CSS foi a adição de bordas arredondadas.

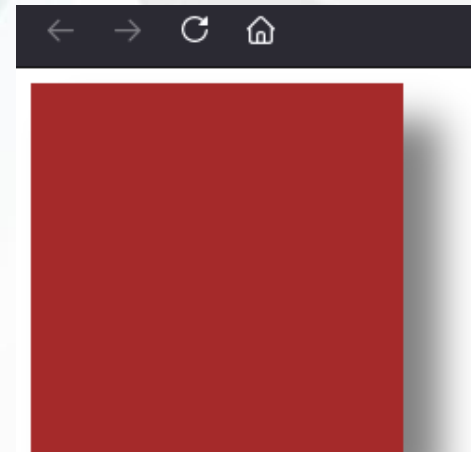
```
div {  
  border-radius: 100px;  
  background-color: red;  
  width: 200px;  
  height: 200px;  
}
```



Box-Shadow

- ✓ Além de sombras em texto, podemos colocar sombras em qualquer elemento com box-shadow. A sintaxe é parecida com a do text-shadow:

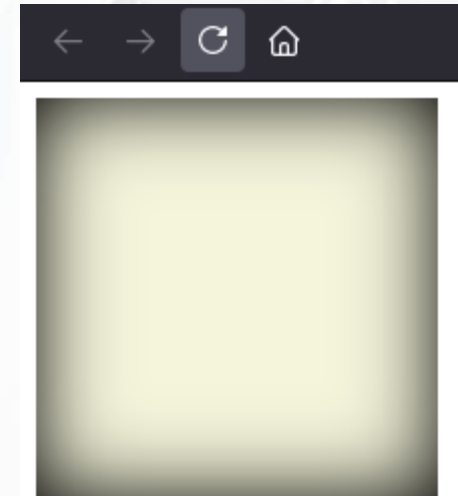
```
div {  
  box-shadow: 20px 20px 20px gray;  
  width: 200px;  
  height: 200px;  
  background-color: brown;  
}
```



Box-Shadow

- ✓ é possível usar a keyword inset para uma borda interna ao elemento:

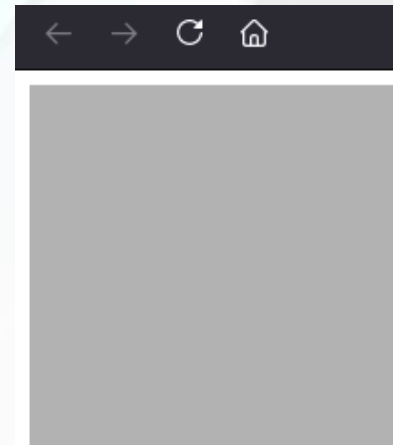
```
div {  
  box-shadow: inset 0 0 40px black;  
  width: 200px;  
  height: 200px;  
  background-color: beige;  
}
```



Opacidade

- ✓ É possível mudar a opacidade de um elemento para que ele seja mais transparente com o uso da propriedade `opacity`:

```
div {  
  opacity: 0.3;  
  width: 200px;  
  height: 200px;  
  background-color: black;  
}
```



Text-Shadow

- ✓ Outro efeito do CSS é o suporte a sombras em textos com text-shadow. Sua sintaxe recebe o deslocamento da sombra e sua cor:

```
h1 {  
  text-shadow: 5px 5px 8px gray;  
}
```

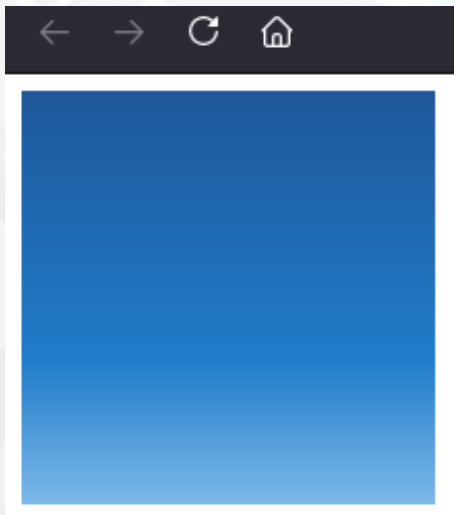


Gradiente

- ✓ O CSS traz também suporte à declaração de gradientes sem que precisemos trabalhar com imagens.
- ✓ Existe suporte a gradientes lineares e radiais, inclusive com múltiplas paradas.

Gradiente

```
.linear {  
  background: linear-gradient(to bottom, █ #1e5799 0%, █ #207cca 65%, █ #7db9e8 100%);  
  width: 200px;  
  height: 200px;  
}
```



Transitions

- ✓ Com as transitions, conseguimos animar o processo de mudança de algum valor do CSS.
- ✓ Por exemplo: temos um elemento na posição `top:10px` e, quando passarmos o mouse em cima (hover), queremos que o elemento vá para `top:30px`.

Transitions

```
div {  
  width: 483px;  
  height: 298px;  
  background-color: #676470;  
  transition: all 0.3s ease;  
}  
  
.fade {  
  opacity: 0.5;  
}  
  
.fade:hover {  
  opacity: 1;  
}
```


Transitions

- ✓ Por padrão, a animação é linear, mas temos outros tipos para animações mais suaves:
 - linear - velocidade constante na animação;
 - ease - redução gradual na velocidade da animação;
 - ease-in - aumento gradual na velocidade da animação;
 - ease-in-out - aumento gradual, depois redução gradual na velocidade da animação;
 - cubic-bezier(x1,y1,x2,y2) - curva de velocidade para animação customizada (avançado);

Transforms

- ✓ Podemos alterar o ângulo de um elemento, mostrá-lo em uma escala maior ou menor que seu tamanho padrão ou alterar a posição do elemento sem sofrer interferência de sua estrutura.

```
img {  
  transform: rotate(-10deg);  
}
```

Font-face

- ✓ A diretiva @font-face destina-se a especificar um tipo de fonte e ter a garantia que a fonte especificada será renderizada no navegador do usuário independentemente de ele ter ou não a fonte instalada em seu sistema operacional.

```
@font-face {  
  font-family: minhafonte;  
  src: url(../fonts/sansation_light.woff);  
}
```

Import

- ✓ Uma das formas mais simples e garantidas são as fontes do Google Web fonts, pois são fontes próprias para web e todas testadas e aprovadas pela plataforma google.

```
@import "http://fonts.googleapis.com/css?family=Raleway";  
body{  
    font-family: 'Raleway', sans-serif;  
}
```




Universidade Federal de Santa Catarina
Curso de Tecnologias da Informação e Comunicação
Desenvolvimento de Sistemas Web
Professor: Adriano de Oliveira

FONTE: W3schools

[CSS Tutorial \(w3schools.com\)](http://w3schools.com)

[CSS-Tricks](#)