



3. Construção de interfaces de usuário

3.1. Introdução ao HTML

3.2. Introdução ao CSS

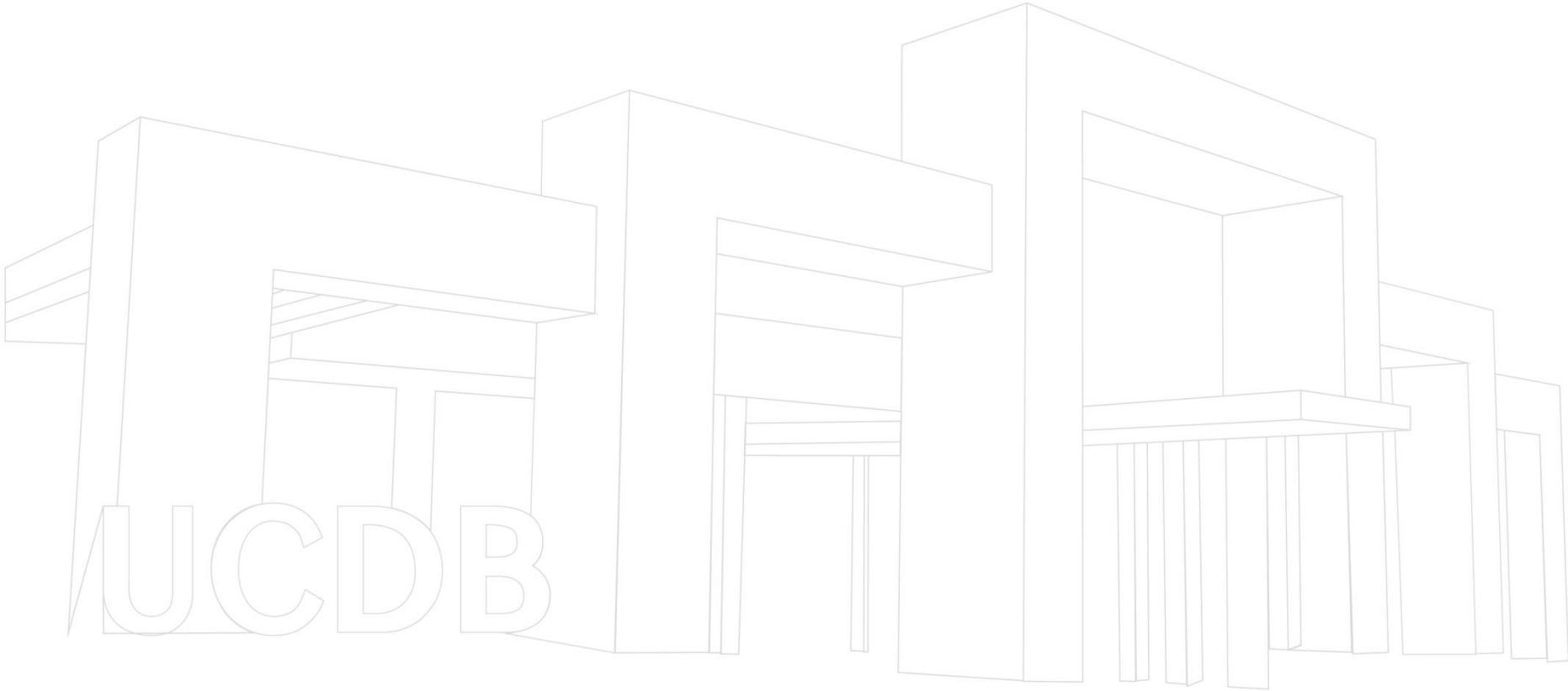
Prof. Me. Marcos Alves
marcos@ucdb.br



3.1. Introdução ao HTML

Prof. Me. Marcos Alves
marcos@ucdb.br

Linguagem HTML



Linguagem HTML

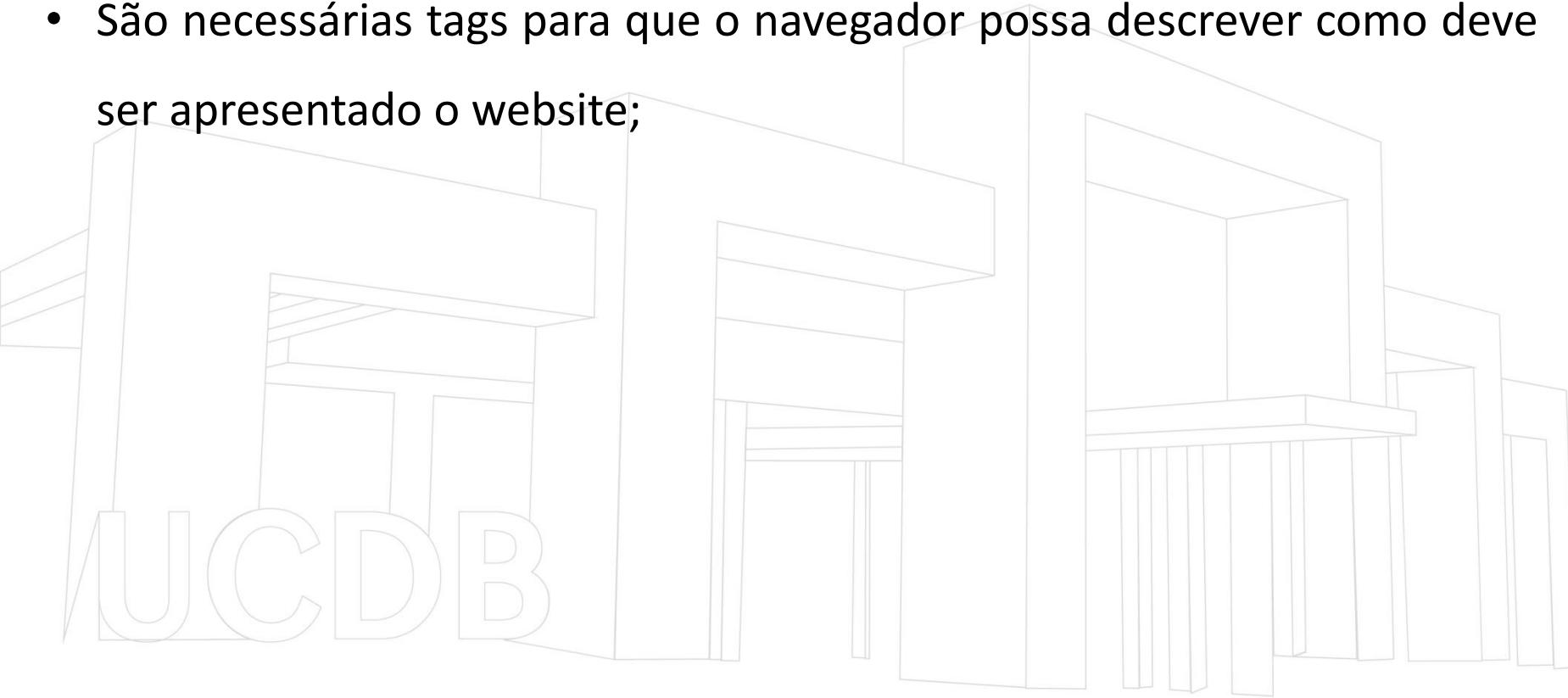
TAGS



Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;



Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;

Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;
- Exemplo:

Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;
- Exemplo:
abertura: `<html>`

Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;
- Exemplo:
abertura: `<html>`
fechamento: `</html>`

Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;
- Exemplo:
abertura: `<html>`
fechamento: `</html>`
- Há dois tipos de tags, as de abertura e de fechamento;

Linguagem HTML

TAGS

- São necessárias tags para que o navegador possa descrever como deve ser apresentado o website;
- As tags tem sempre o mesmo formato começam, com o sinal ‘<’ e terminam com ‘>’;
- Exemplo:
 - abertura: `<html>`
 - fechamento: `</html>`
- Há dois tipos de tags, as de abertura e de fechamento;
- Tudo que estiver contido entre uma tag de abertura e uma tag de fechamento será processado segundo o comando contido na tag.

Linguagem HTML

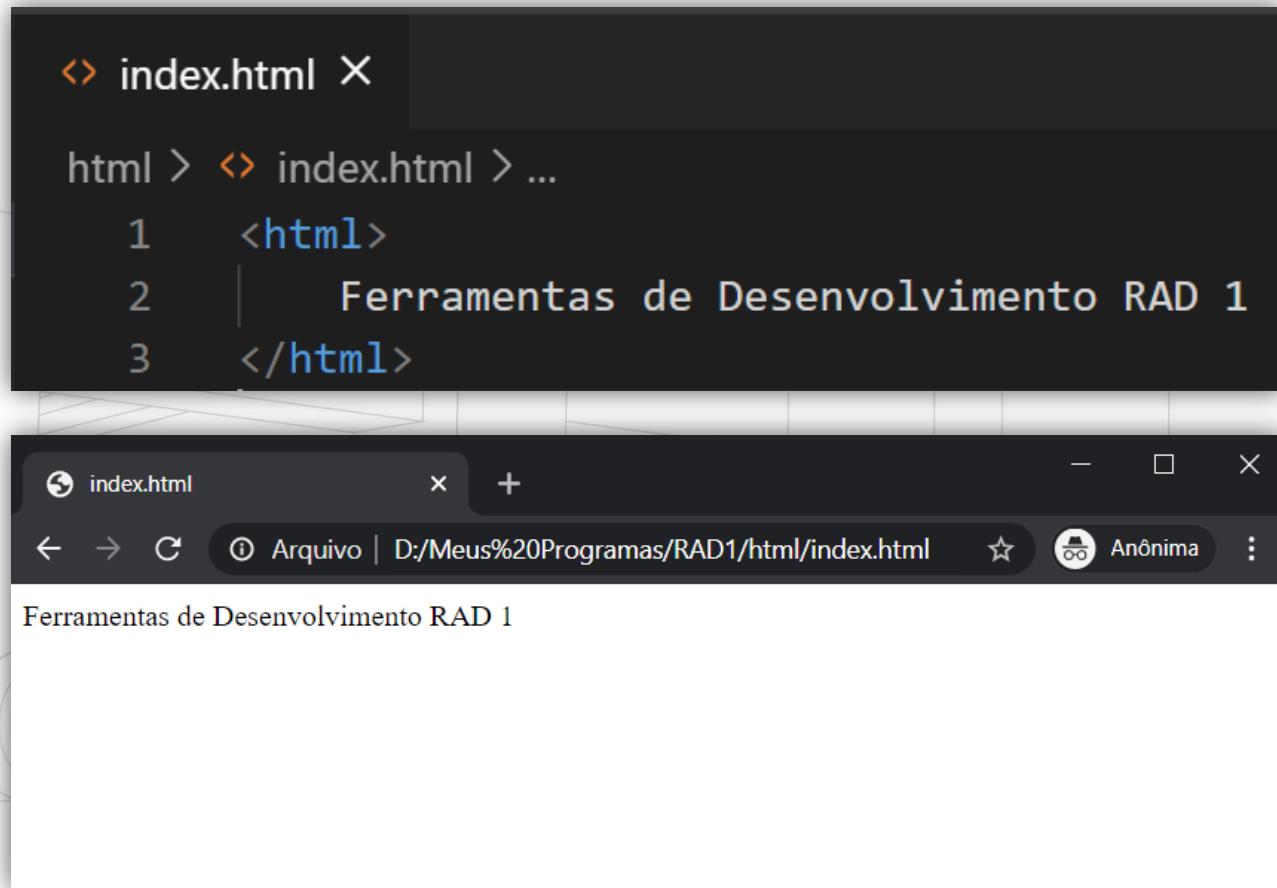
TAGS

```
<> index.html >

html > <> index.html > ...
1   <html>
2   |   Ferramentas de Desenvolvimento RAD 1
3   </html>
```

Linguagem HTML

TAGS



The image shows a screenshot of a web browser window and its corresponding code editor. The browser window title is "index.html" and the address bar shows "Arquivo | D:/Meus%20Programas/RAD1/html/index.html". The page content is "Ferramentas de Desenvolvimento RAD 1". Above the browser window, a code editor window displays the following HTML code:

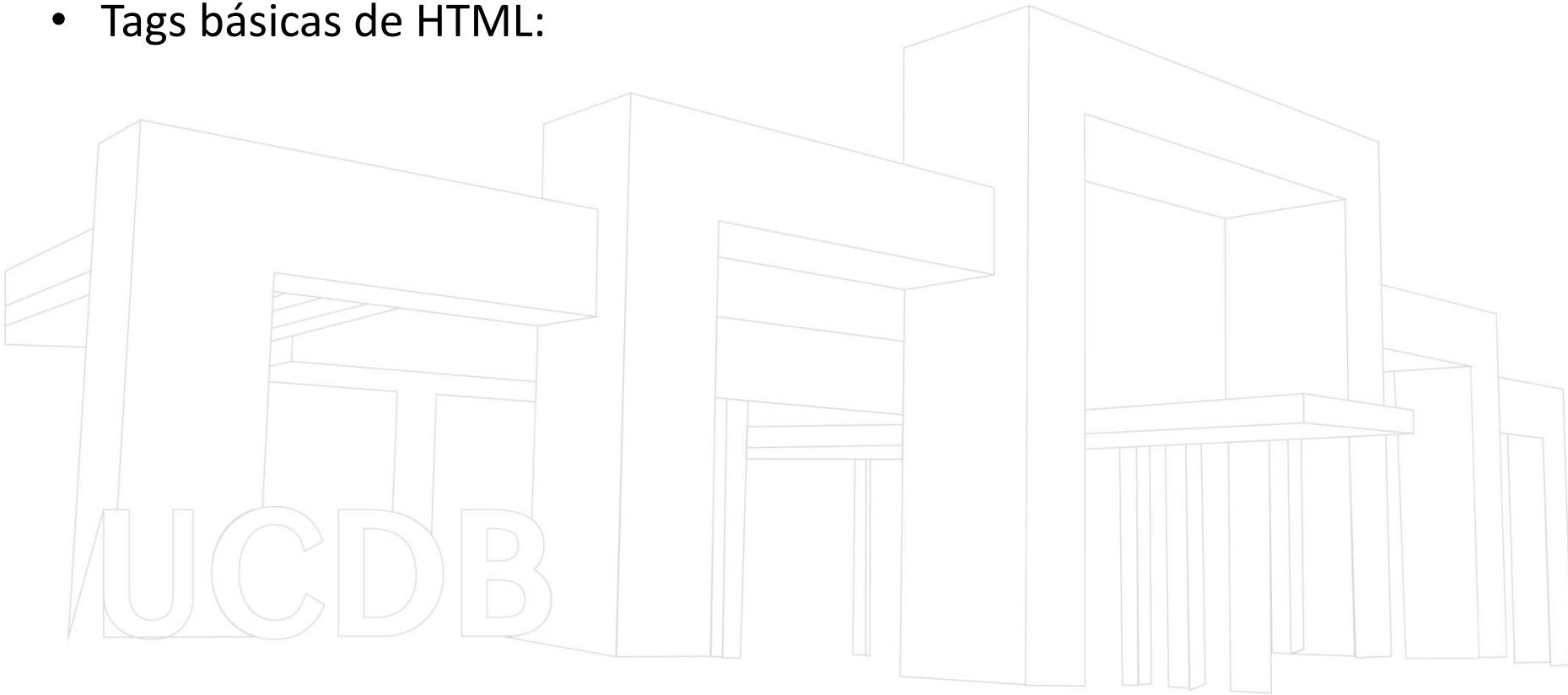
```
<> index.html < />

html > <> index.html > ...
1   <html>
2   |     Ferramentas de Desenvolvimento RAD 1
3   </html>
```

Linguagem HTML

TAGS

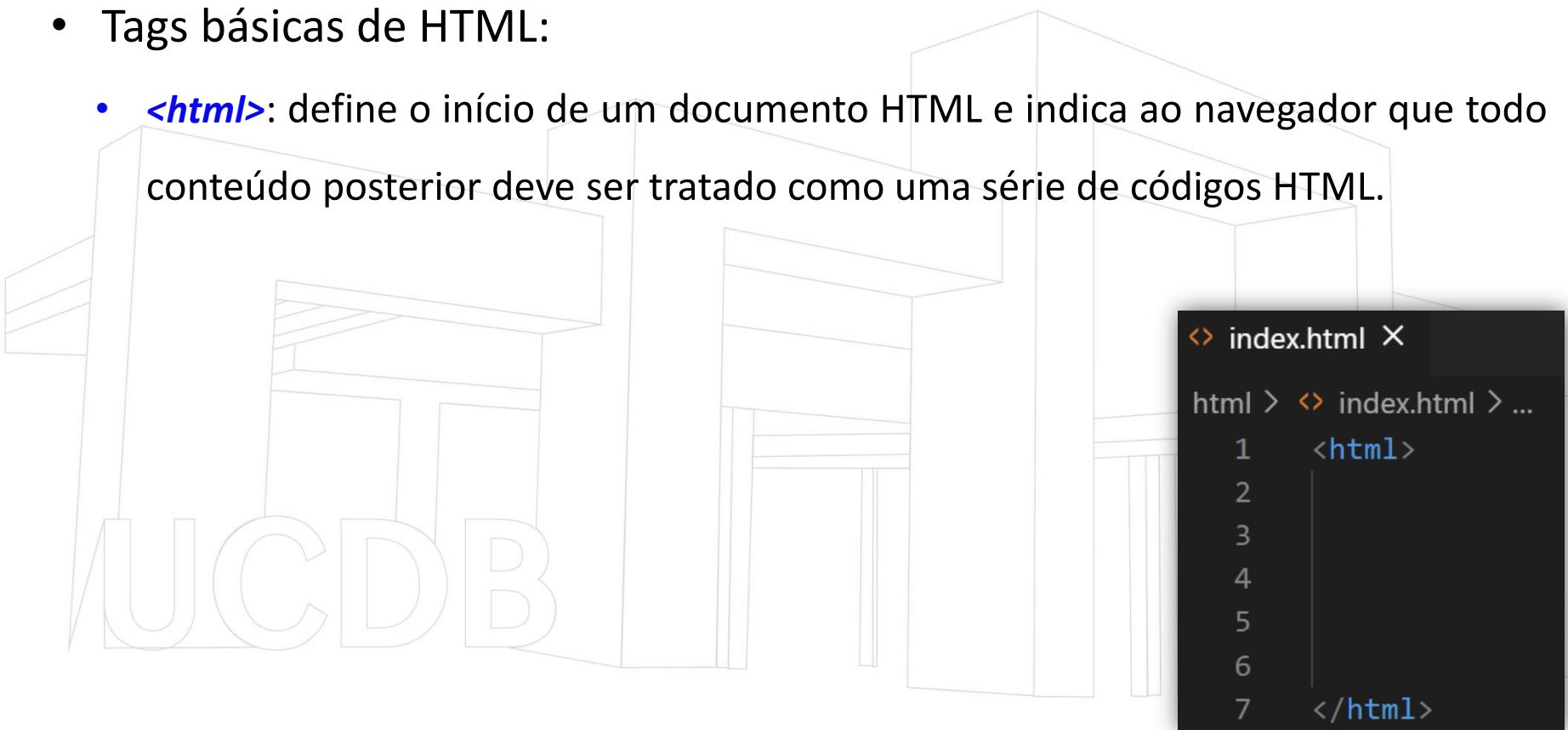
- Tags básicas de HTML:



Linguagem HTML

TAGS

- Tags básicas de HTML:
 - `<html>`: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.

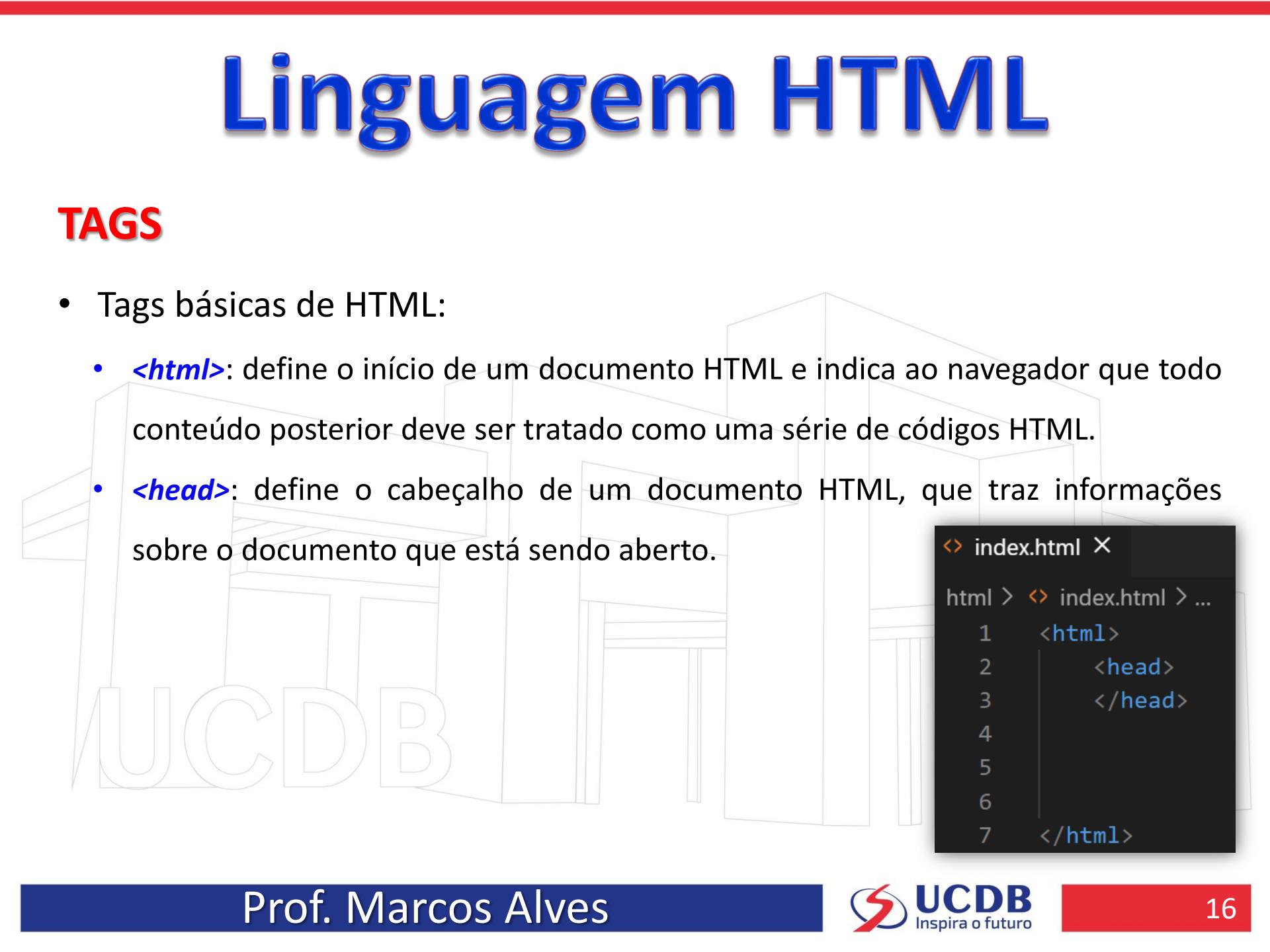


```
↳ index.html X  
  
html > ↳ index.html > ...  
1   <html>  
2  
3  
4  
5  
6  
7   </html>
```

Linguagem HTML

TAGS

- Tags básicas de HTML:
 - `<html>`: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.
 - `<head>`: define o cabeçalho de um documento HTML, que traz informações sobre o documento que está sendo aberto.

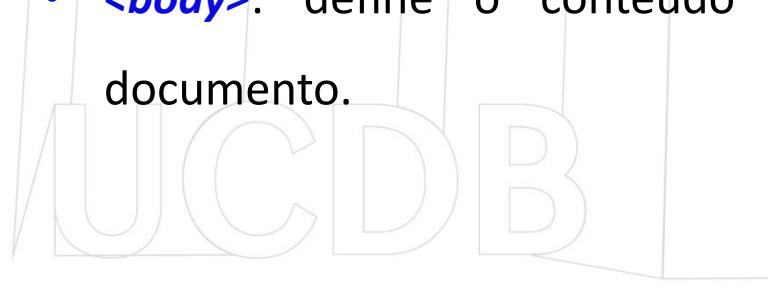


```
index.html X
html > index.html > ...
1   <html>
2     <head>
3     </head>
4
5
6
7   </html>
```

Linguagem HTML

TAGS

- Tags básicas de HTML:
 - **<html>**: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.
 - **<head>**: define o cabeçalho de um documento HTML, que traz informações sobre o documento que está sendo aberto.
 - **<body>**: define o conteúdo principal, o corpo do documento.



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3     </head>
4
5   <body>
6   </body>
7 </html>
```

Linguagem HTML

TAGS

- Tags básicas de HTML:
 - **<html>**: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.
 - **<head>**: define o cabeçalho de um documento HTML, que traz informações sobre o documento que está sendo aberto.
 - **<body>**: define o conteúdo principal, o corpo do documento.
 - Esta é a parte do documento HTML que é exibida no navegador.

```
index.html X
html > index.html > ...
1   <html>
2     <head>
3     </head>
4   <body>
5     </body>
6   </html>
7 
```

Linguagem HTML

TAGS

- Tags básicas de HTML:
 - **<html>**: define o início de um documento HTML e indica ao navegador que todo conteúdo posterior deve ser tratado como uma série de códigos HTML.
 - **<head>**: define o cabeçalho de um documento HTML, que traz informações sobre o documento que está sendo aberto.
 - **<body>**: define o conteúdo principal, o corpo do documento.
 - Esta é a parte do documento HTML que é exibida no navegador.
 - No corpo pode-se definir propriedades comuns a toda página, como cor de fundo, margens, e outras formatações.

```
index.html X
html > index.html > ...
1   <html>
2     <head>
3     </head>
4
5   <body>
6   </body>
7 </html>
```

Linguagem HTML

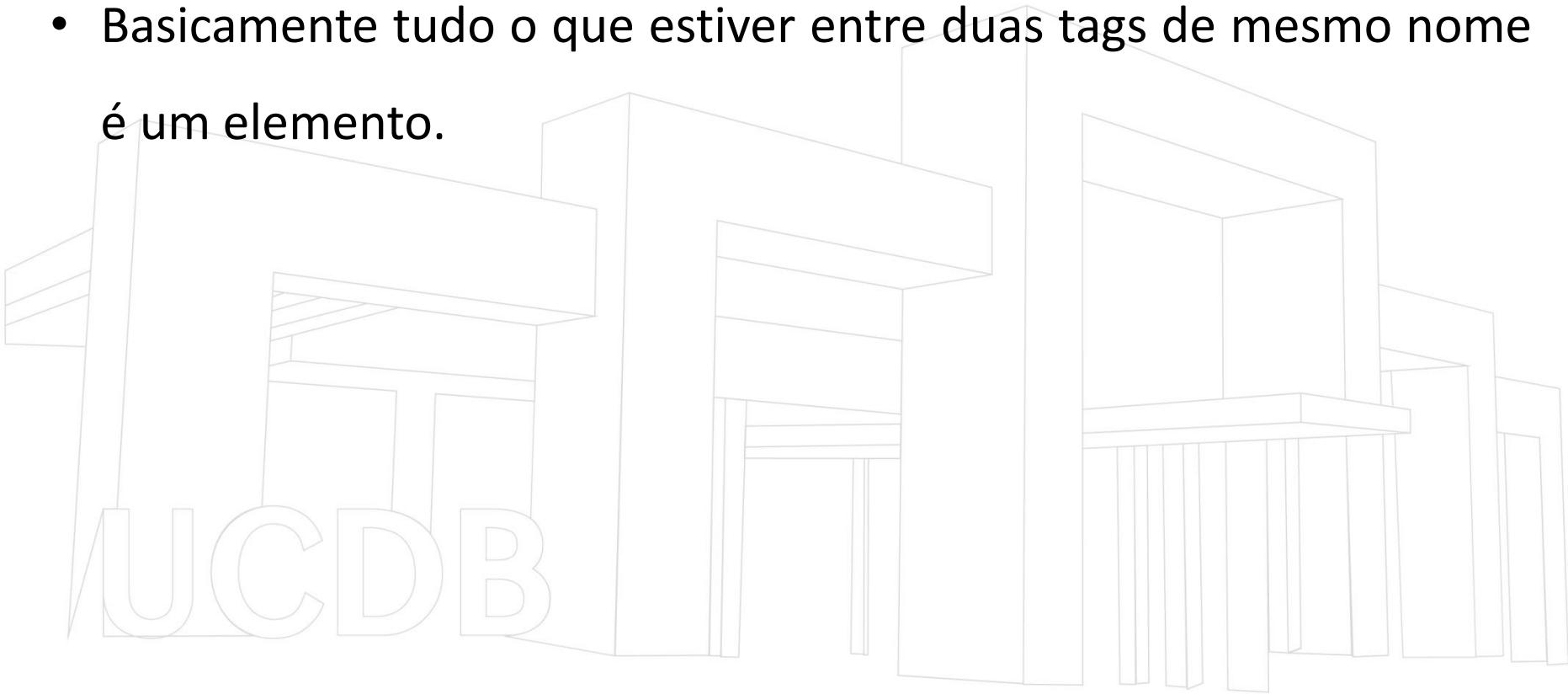
ELEMENTOS HTML



Linguagem HTML

ELEMENTOS HTML

- Basicamente tudo o que estiver entre duas tags de mesmo nome é um elemento.



Linguagem HTML

ELEMENTOS HTML

- Basicamente tudo o que estiver entre duas tags de mesmo nome é um elemento.
- Por exemplo, o elemento BODY é tudo o que se encontra entre `<body>` (tag de abertura) e `</body>` (tag de fechamento).

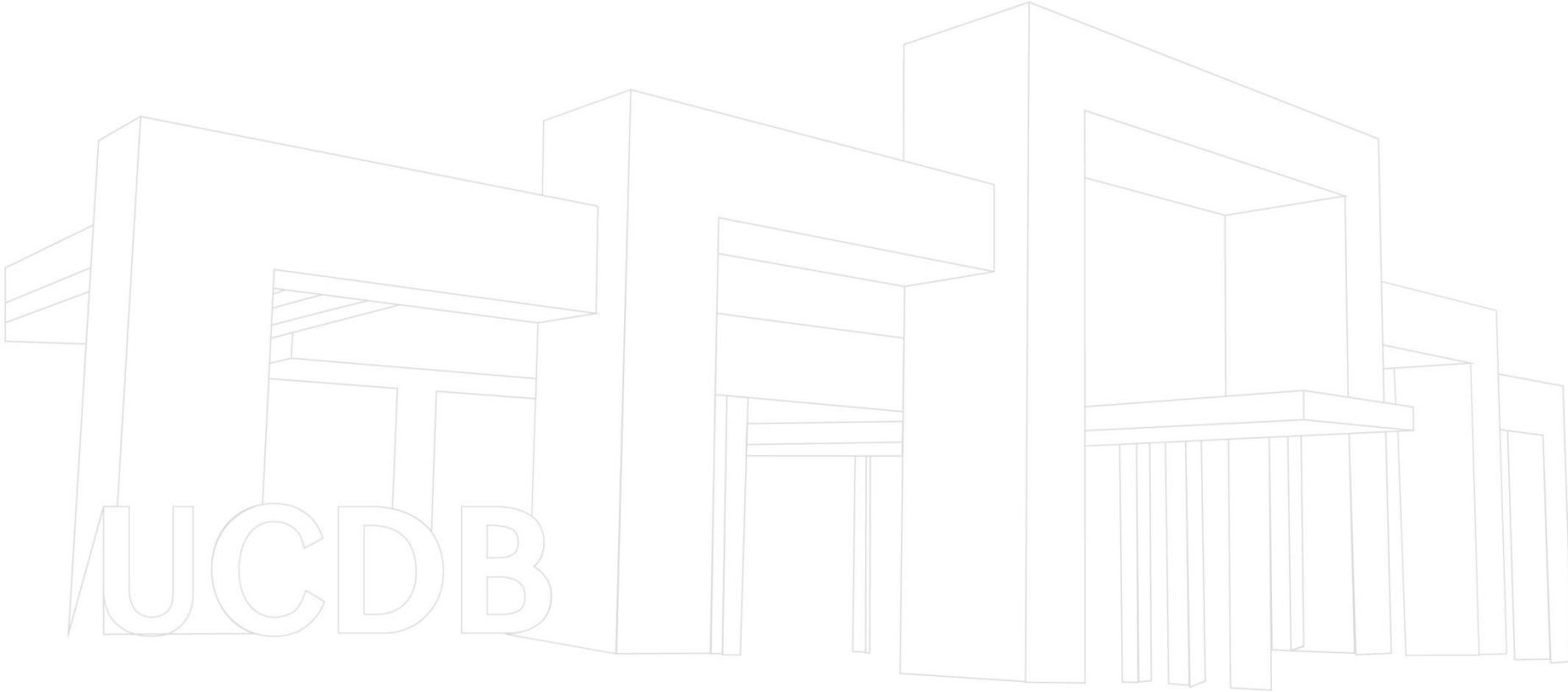
Linguagem HTML

ELEMENTOS HTML

- Basicamente tudo o que estiver entre duas tags de mesmo nome é um elemento.
- Por exemplo, o elemento BODY é tudo o que se encontra entre `<body>` (tag de abertura) e `</body>` (tag de fechamento).
- Outros exemplos de elementos são os cabeçalhos, os parágrafos e as tabelas.

Linguagem HTML

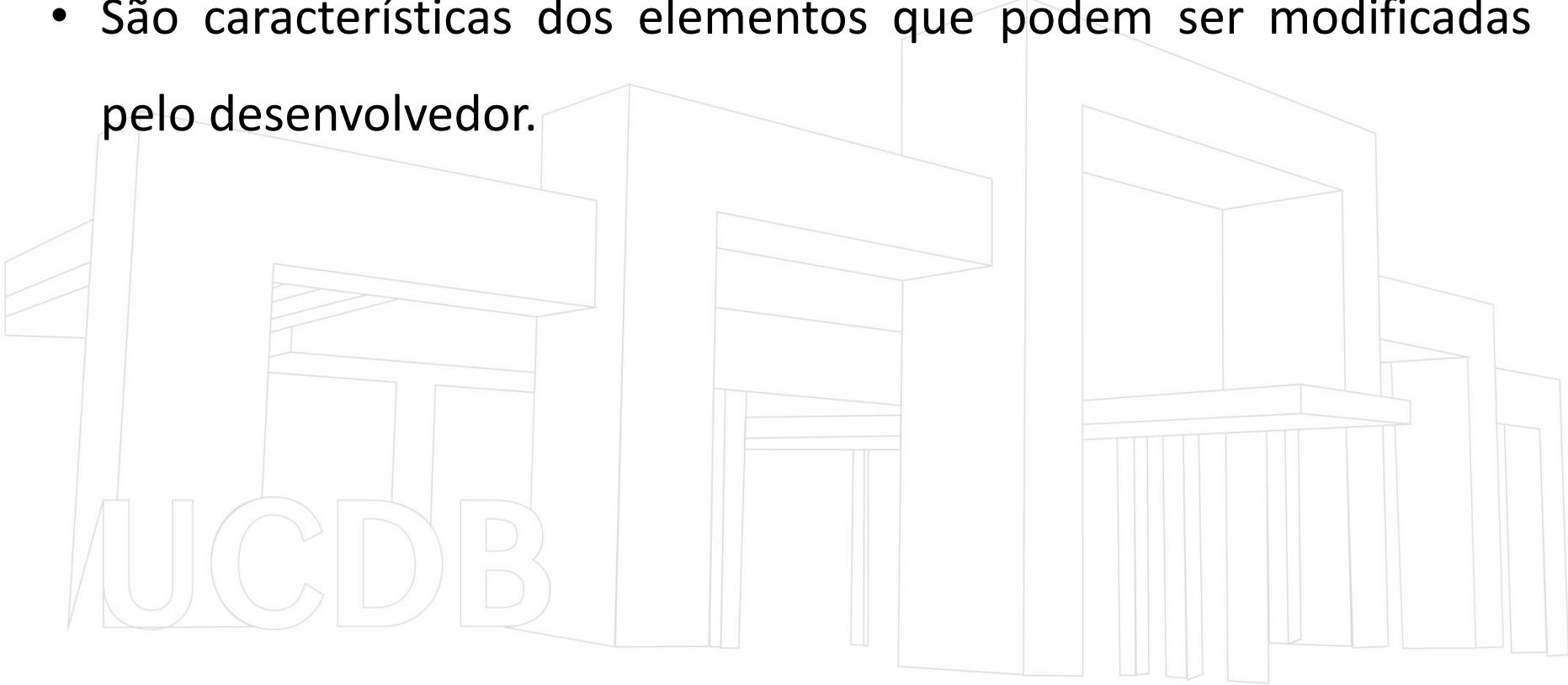
ATRIBUTOS HTML



Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.



Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.
- As propriedades da fonte, por exemplo, são atributos que podem ser modificados no cabeçalho, nos parágrafos, etc.

Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.
- As propriedades da fonte, por exemplo, são atributos que podem ser modificados no cabeçalho, nos parágrafos, etc.
- Os atributos devem ser sempre modificados na tag de abertura.

Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.
- As propriedades da fonte, por exemplo, são atributos que podem ser modificados no cabeçalho, nos parágrafos, etc.
- Os atributos devem ser sempre modificados na tag de abertura.
- Exemplo:

Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.
- As propriedades da fonte, por exemplo, são atributos que podem ser modificados no cabeçalho, nos parágrafos, etc.
- Os atributos devem ser sempre modificados na tag de abertura.
- Exemplo:
`<body>`
conteúdo
`</body>`

Linguagem HTML

ATRIBUTOS HTML

- São características dos elementos que podem ser modificadas pelo desenvolvedor.
- As propriedades da fonte, por exemplo, são atributos que podem ser modificados no cabeçalho, nos parágrafos, etc.
- Os atributos devem ser sempre modificados na tag de abertura.
- Exemplo:
`<body atributo=valor>`
conteúdo
`</body>`

Linguagem HTML

DEFININDO A ESTRUTURA DA PÁGINA EM HTML

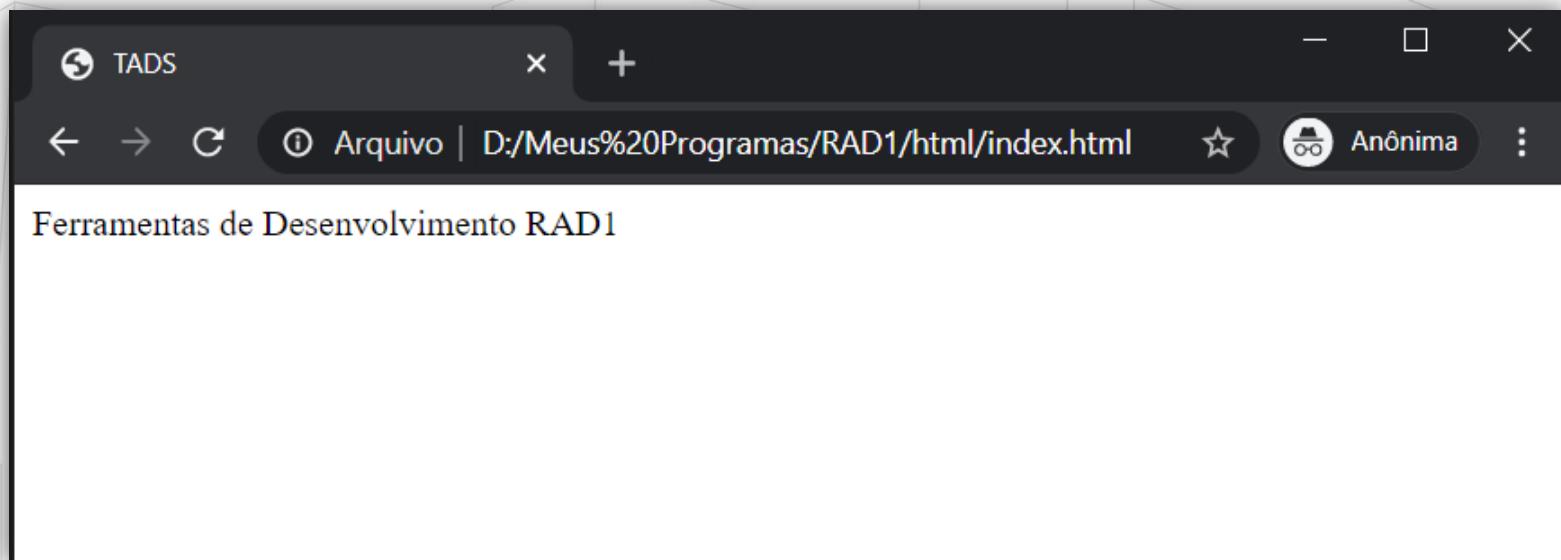
- **Cabeçalho (`<head>`):** dentro dele utilizamos a tag `<title>` que define o título da página



Linguagem HTML

DEFININDO A ESTRUTURA DA PÁGINA EM HTML

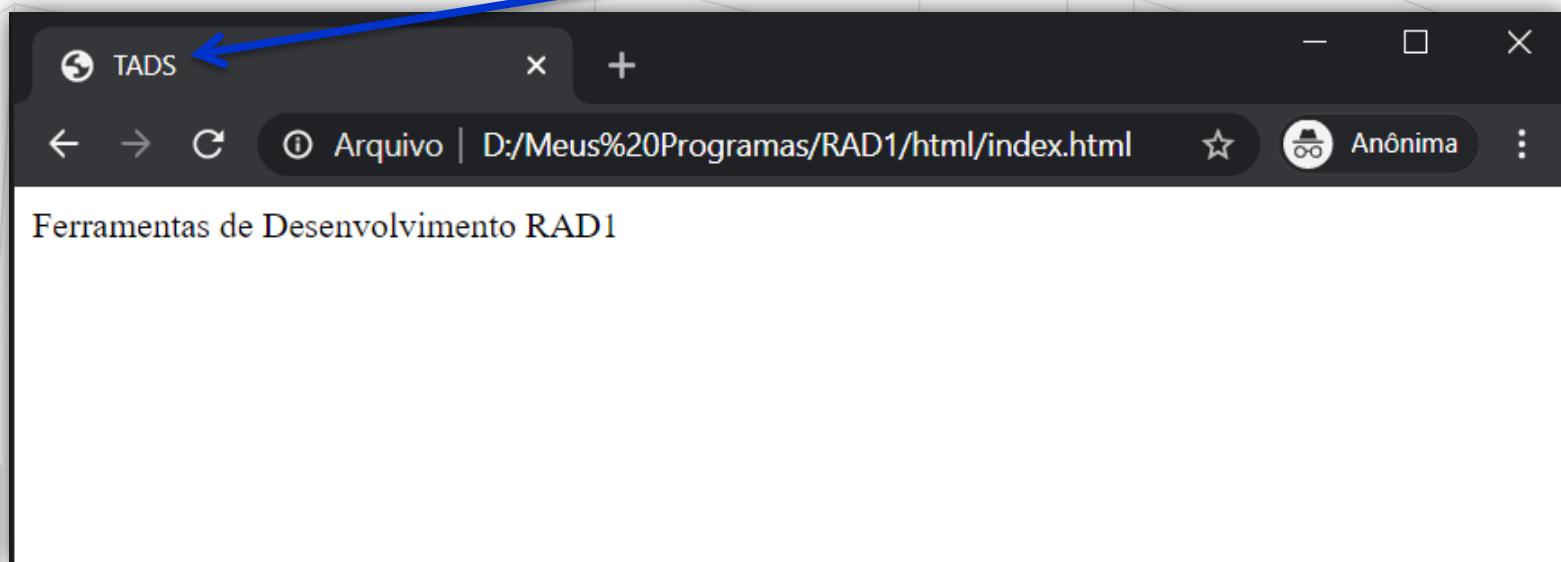
- **Cabeçalho (`<head>`):** dentro dele utilizamos a tag `<title>` que define o título da página, que é exibido na barra de título dos navegadores:



Linguagem HTML

DEFININDO A ESTRUTURA DA PÁGINA EM HTML

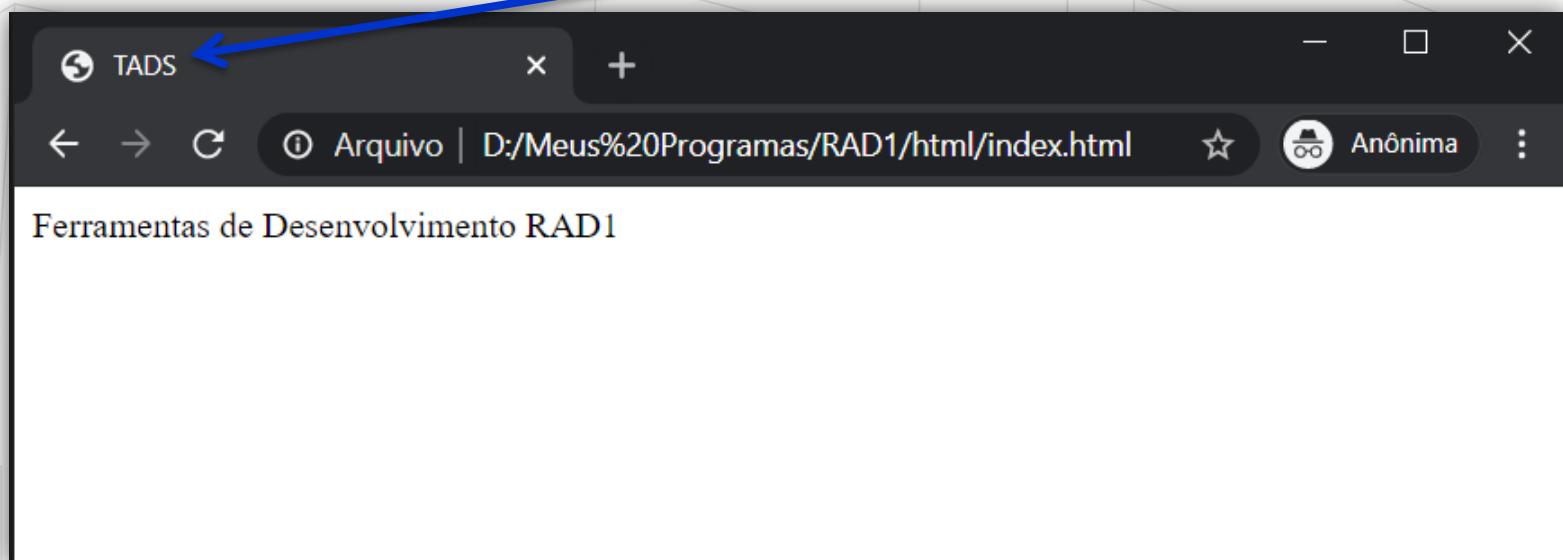
- **Cabeçalho (`<head>`)**: dentro dele utilizamos a tag `<title>` que define o título da página, que é exibido na **barra de título** dos navegadores:



Linguagem HTML

DEFININDO A ESTRUTURA DA PÁGINA EM HTML

- **Cabeçalho (`<head>`)**: dentro dele utilizamos a tag `<title>` que define o título da página, que é exibido na **barra de título** dos navegadores:



- **Corpo (`<body>`)**: no corpo encontramos as tags de formatação, de tabelas e imagens, etc.

Linguagem HTML

FORMATAÇÃO DE TEXTO



Linguagem HTML

FORMATAÇÃO DE TEXTO

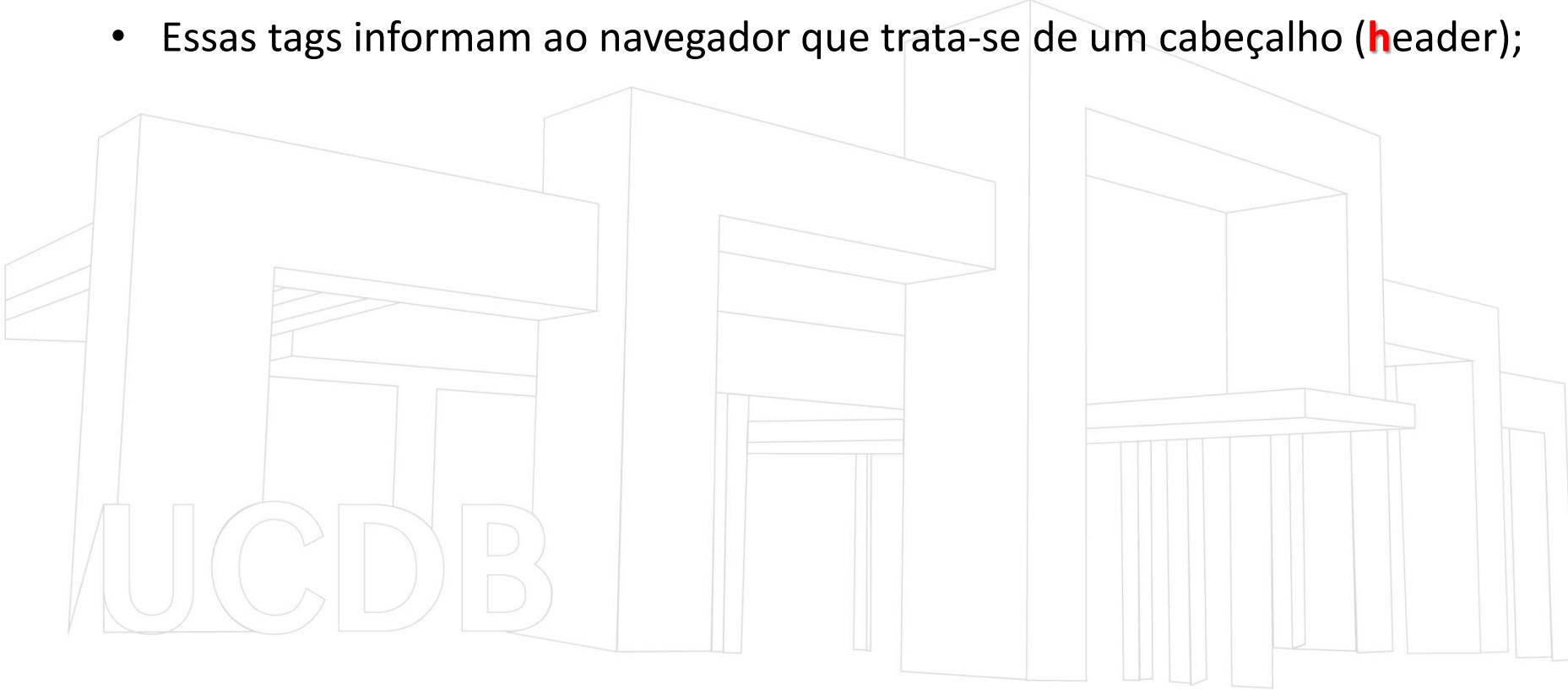
- `<h1...h6>`:



Linguagem HTML

FORMATAÇÃO DE TEXTO

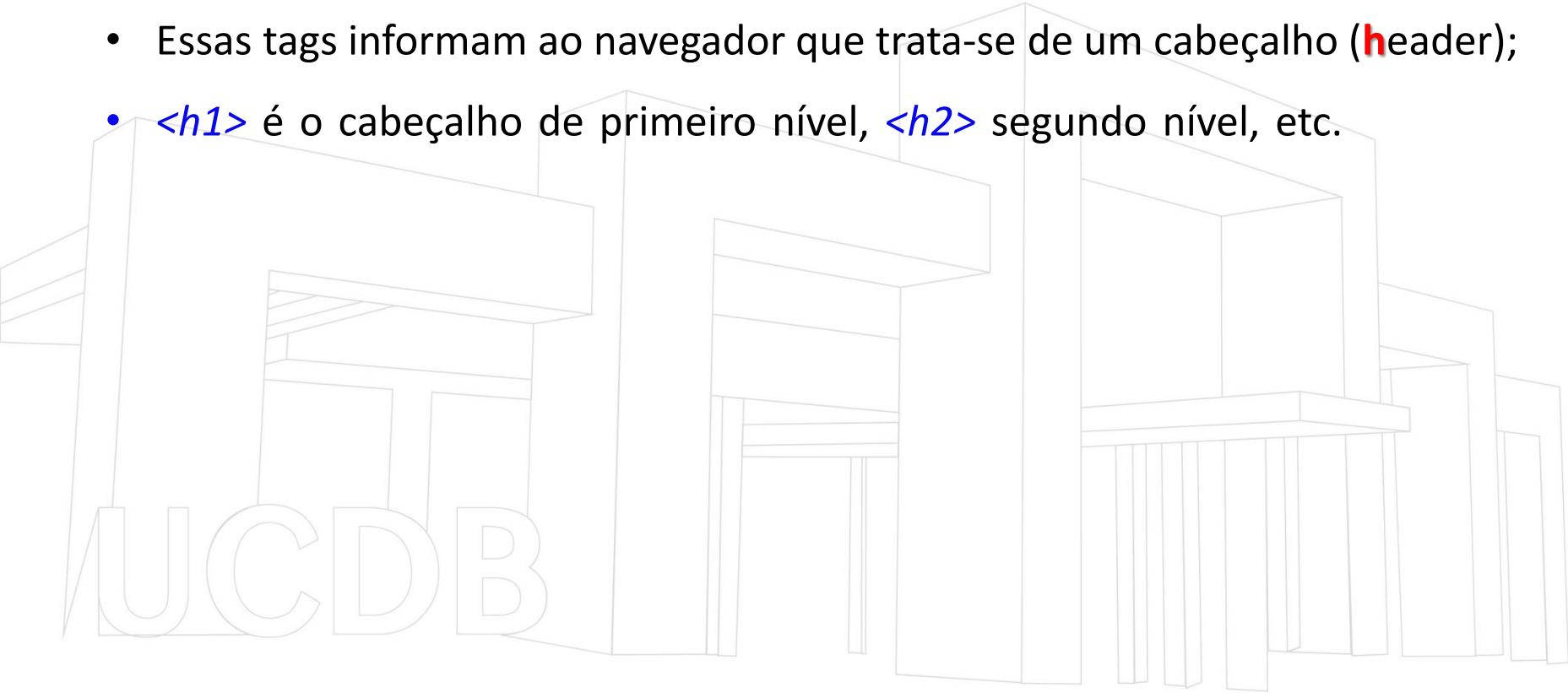
- `<h1...h6>`:
 - Essas tags informam ao navegador que trata-se de um cabeçalho (**header**);



Linguagem HTML

FORMATAÇÃO DE TEXTO

- `<h1...h6>`:
 - Essas tags informam ao navegador que trata-se de um cabeçalho (**header**);
 - `<h1>` é o cabeçalho de primeiro nível, `<h2>` segundo nível, etc.



Linguagem HTML

FORMATAÇÃO DE TEXTO

- `<h1...h6>`:
 - Essas tags informam ao navegador que trata-se de um cabeçalho (**header**);
 - `<h1>` é o cabeçalho de primeiro nível, `<h2>` segundo nível, etc. Do maior para o menor tamanho de texto, respectivamente.

Linguagem HTML

FORMATAÇÃO DE TEXTO

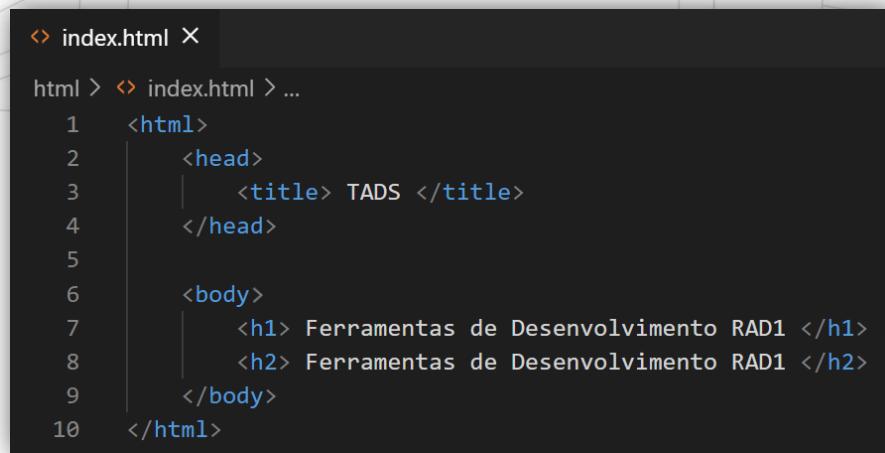
- `<h1...h6>`:
 - Essas tags informam ao navegador que trata-se de um cabeçalho (**header**);
 - `<h1>` é o cabeçalho de primeiro nível, `<h2>` segundo nível, etc. Do maior para o menor tamanho de texto, respectivamente.

```
index.html X
html > index.html > ...
1  <html>
2   |  <head>
3   |   |  <title> TADS </title>
4   |   </head>
5
6   <body>
7     |  <h1> Ferramentas de Desenvolvimento RAD1 </h1>
8     |  <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9   </body>
10 </html>
```

Linguagem HTML

FORMATAÇÃO DE TEXTO

- `<h1...h6>`:
 - Essas tags informam ao navegador que trata-se de um cabeçalho (**header**);
 - `<h1>` é o cabeçalho de primeiro nível, `<h2>` segundo nível, etc. Do maior para o menor tamanho de texto, respectivamente.



```
index.html X
html > index.html > ...
1  <html>
2   |  <head>
3   |   |  <title> TADS </title>
4   |  </head>
5
6   <body>
7     |  <h1> Ferramentas de Desenvolvimento RAD1 </h1>
8     |  <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9   </body>
10 </html>
```



Linguagem HTML

ALINHAMENTOS



Linguagem HTML

ALINHAMENTOS

- `<h1 align=center>`
- `<h1 align=right >`
- `<h1 align=left>`



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <h1 align=center> Ferramentas de Desenvolvimento RAD1 </h1>
8       <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9     </body>
10    </html>
```

TADS Arquivo | D:/Meus%20Programas/RAD1/html/index.html Anônima ...

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

Linguagem HTML

ALINHAMENTOS

- `<h1 align=center>`
- `<h1 align=right >`
- `<h1 align=left>`



The slide features a large, faint watermark of the letters "UCDB" in the background.

index.html X

```
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <h1 align=right> Ferramentas de Desenvolvimento RAD1 </h1>
8       <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9     </body>
10    </html>
```

TADS Arquivo | D:/Meus%20Programas/RAD1/html/index.html Anônima :

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

Linguagem HTML

ALINHAMENTOS

- `<h1 align=center>`
- `<h1 align=right >`
- `<h1 align=left>`



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <h1 align=left> Ferramentas de Desenvolvimento RAD1 </h1>
8       <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9     </body>
10    </html>
```

TADS Arquivo | D:/Meus%20Programas/RAD1/html/index.html Anônima :

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

Linguagem HTML

ALINHAMENTOS

- `<h1 align=center>`
- `<h1 align=right >`
- `<h1 align=left>`

Observação

O alinhamento à esquerda
é o padrão para texto.



The screenshot shows a code editor and a browser window. The code editor displays the following HTML file:

```
index.html X
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <h1 align=left> Ferramentas de Desenvolvimento RAD1 </h1>
8     <h2> Ferramentas de Desenvolvimento RAD1 </h2>
9   </body>
10 </html>
```

The browser window shows the rendered output:

TADS

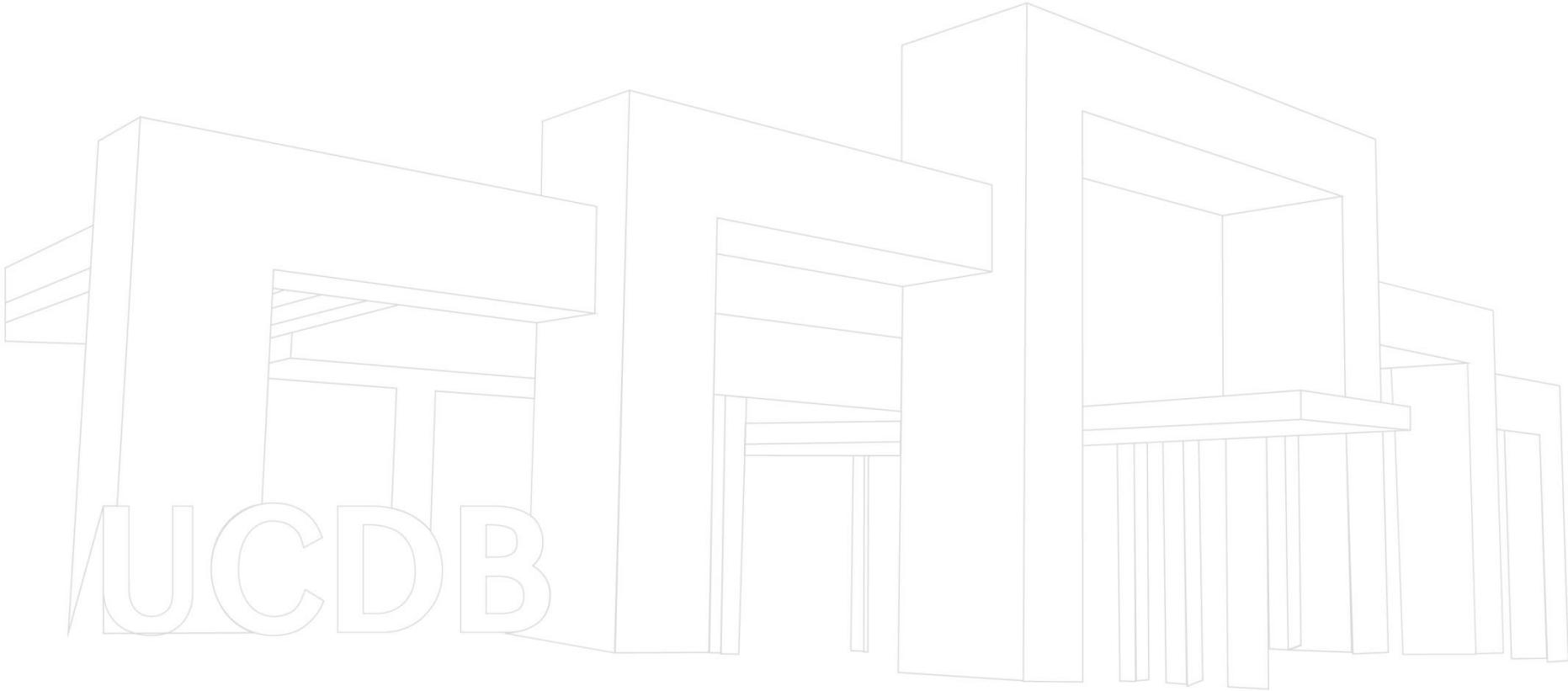
Arquivo | D:/Meus%20Programas/RAD1/html/index.html

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

Linguagem HTML

EFEITOS NO TEXTO



Linguagem HTML

EFEITOS NO TEXTO

- Negrito (**< b >**):

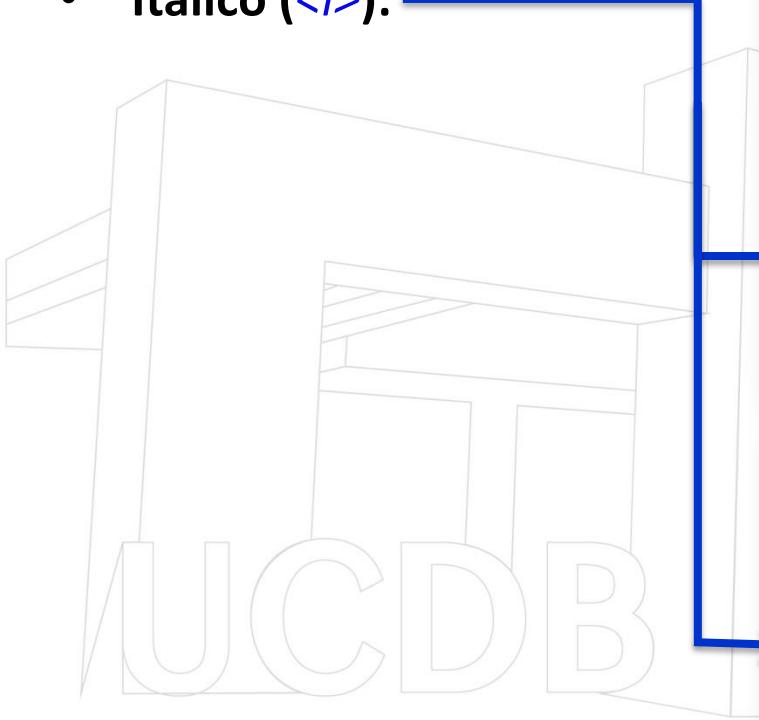
```
index.html X
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8     <h1> <i>Ferramentas de Desenvolvimento RAD1</i> </h1>
9     <h1> <u>Ferramentas de Desenvolvimento RAD1</u> </h1>
10    <h1> <s>Ferramentas de Desenvolvimento RAD1</s> </h1>
11  </body>
12 </html>
```

Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
~~Ferramentas de Desenvolvimento RAD1~~

Linguagem HTML

EFEITOS NO TEXTO

- **Negrito (**):**
- **Itálico (*<i>*):** —————



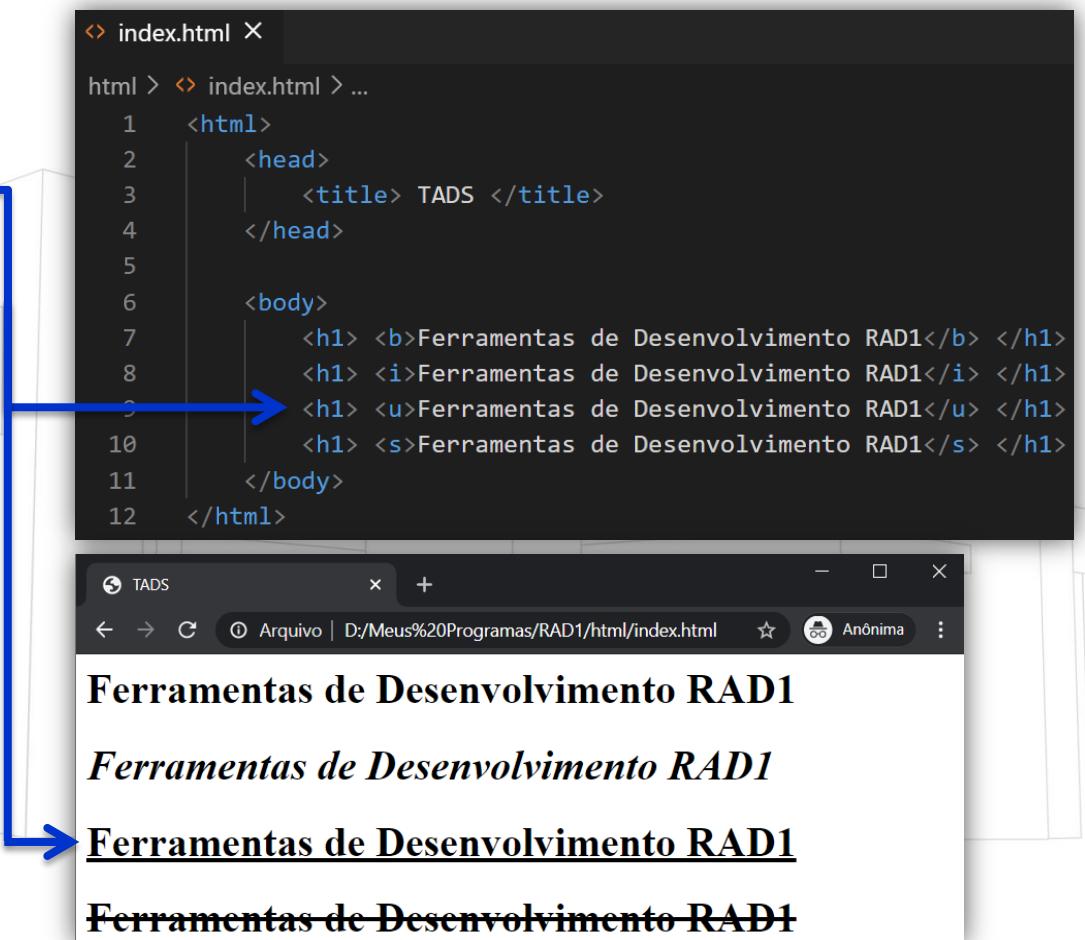
```
index.html
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8     <h1> <i>Ferramentas de Desenvolvimento RAD1</i> </h1>
9     <h1> <u>Ferramentas de Desenvolvimento RAD1</u> </h1>
10    <h1> <s>Ferramentas de Desenvolvimento RAD1</s> </h1>
11  </body>
12 </html>
```

Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
~~Ferramentas de Desenvolvimento RAD1~~

Linguagem HTML

EFEITOS NO TEXTO

- **Negrito (**):**
- **Itálico (*<i>*):**
- **Sublinhado (*<u>*):**



```
index.html
html > index.html > ...
1  <html>
2   <head>
3    |   <title> TADS </title>
4   </head>
5
6   <body>
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8     <h1> <i>Ferramentas de Desenvolvimento RAD1</i> </h1>
9     <h1> <u>Ferramentas de Desenvolvimento RAD1</u> </h1>
10    <h1> <s>Ferramentas de Desenvolvimento RAD1</s> </h1>
11   </body>
12 </html>
```

The browser window shows the rendered output:

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

Ferramentas de Desenvolvimento RAD1

~~Ferramentas de Desenvolvimento RAD1~~

Linguagem HTML

EFEITOS NO TEXTO

- **Negrito (**):**
- **Itálico (*<i>*):**
- **Sublinhado (*<u>*):**
- **Riscado (*<s>*):** —————

```
index.html
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8     <h1> <i>Ferramentas de Desenvolvimento RAD1</i> </h1>
9     <h1> <u>Ferramentas de Desenvolvimento RAD1</u> </h1>
10    <h1> <s>Ferramentas de Desenvolvimento RAD1</s> </h1>
11  </body>
12 </html>
```

Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
Ferramentas de Desenvolvimento RAD1
~~Ferramentas de Desenvolvimento RAD1~~

Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (`<p>`):



Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (`<p>`):
 - EXEMPLO:



Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (`<p>`):

- EXEMPLO:

```
<p>
```

```
</p>
```



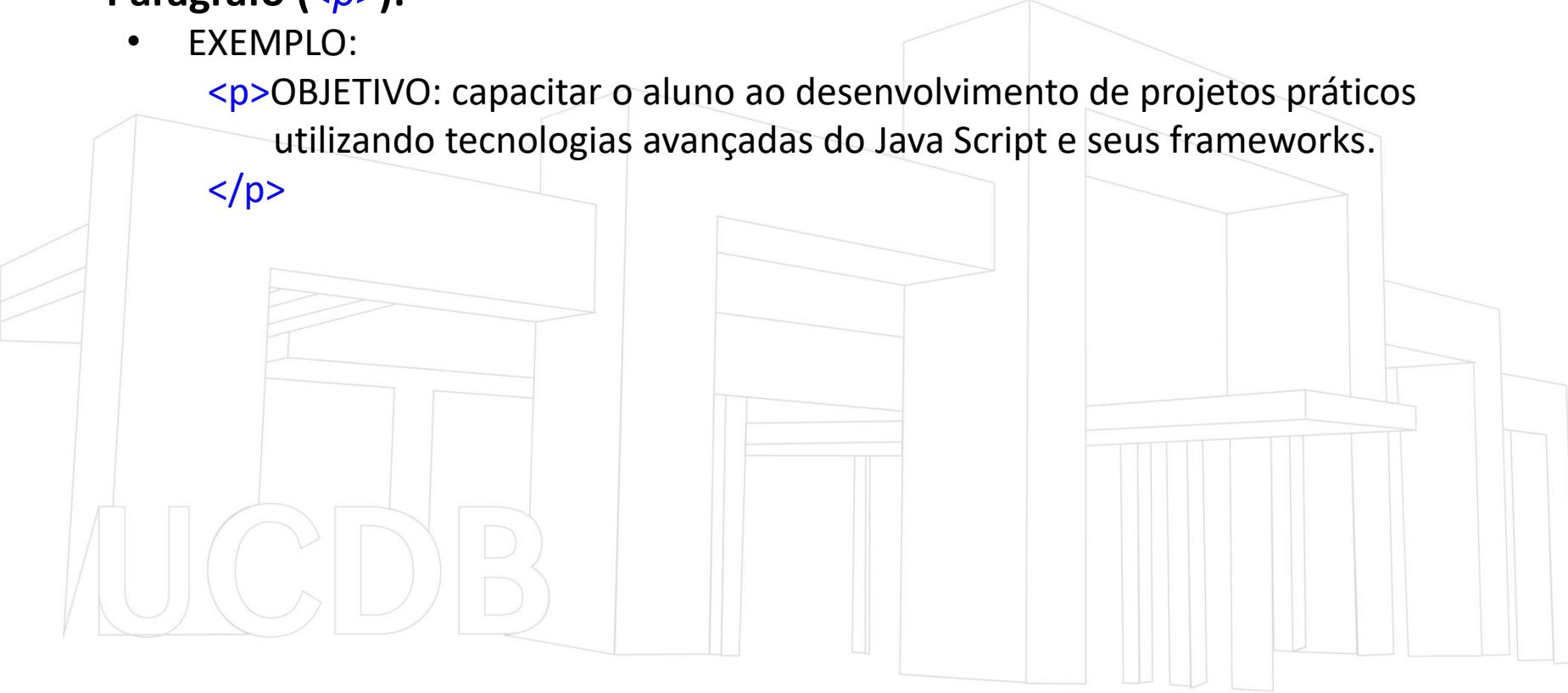
Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (*<p>*):

- EXEMPLO:

*<p>*OBJETIVO: capacitar o aluno ao desenvolvimento de projetos práticos utilizando tecnologias avançadas do Java Script e seus frameworks.
</p>



Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (*<p>*):

- EXEMPLO:

*< p >*OBJETIVO: capacitar o aluno ao desenvolvimento de projetos práticos utilizando tecnologias avançadas do Java Script e seus frameworks.
</ p >

```
index.html X

html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8     <p>OBJETIVO: capacitar o aluno ao
9       desenvolvimento de projetos práticos
10      utilizando tecnologias avançadas do Java
11        Script e seus frameworks.
12
13   </p>
14 </body>
15 </html>
```



Linguagem HTML

EFEITOS NO TEXTO

- Parágrafo (*<p>*):

- EXEMPLO:

*<p>*OBJETIVO: capacitar o aluno ao desenvolvimento de projetos práticos utilizando tecnologias avançadas do Java Script e seus frameworks.
</p>

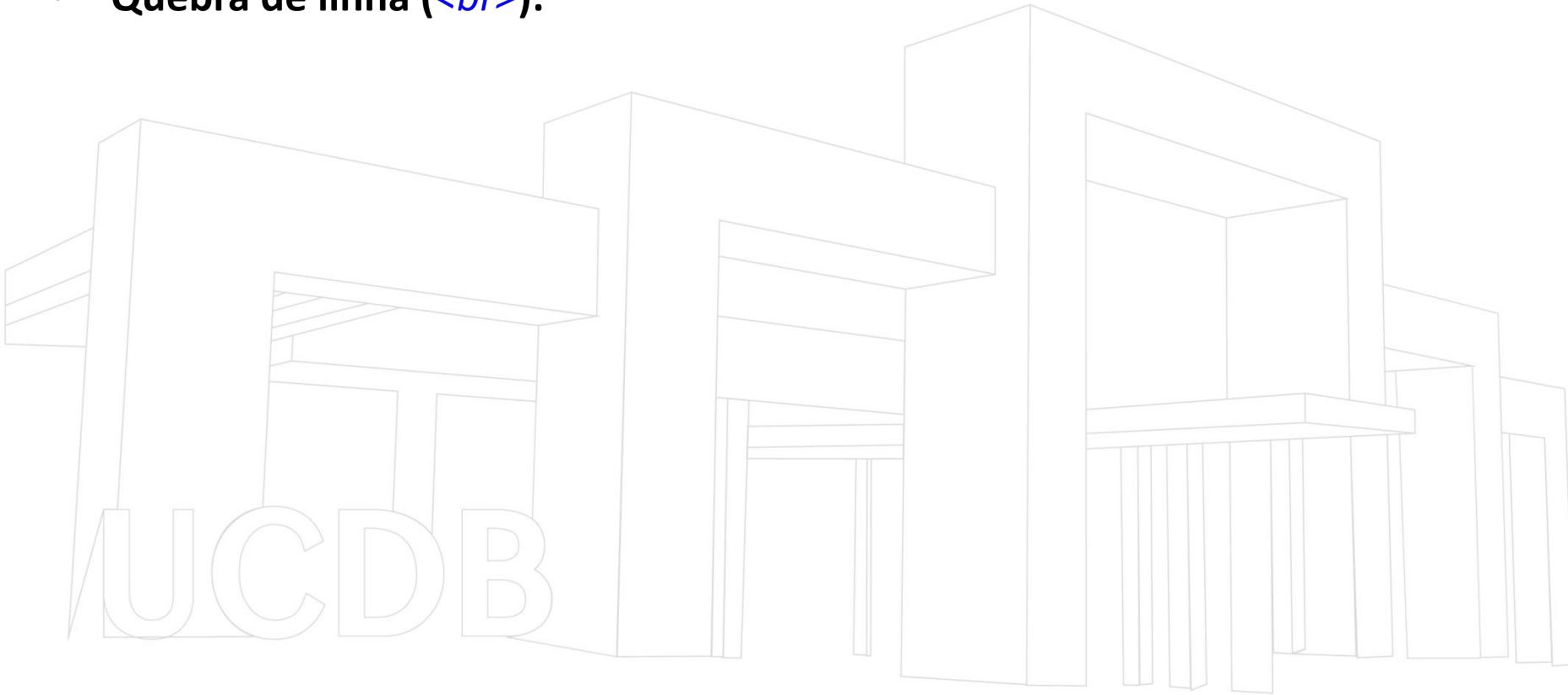
```
index.html X
html > index.html > ...
1  <html>
2    <head>
3      <title> TADS </title>
4    </head>
5
6    <body>
7      <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>
8      <p>OBJETIVO: capacitar o aluno ao
9        desenvolvimento de projetos práticos
10       utilizando tecnologias avanças do Java
11         Script e seus frameworks.
12
13    </p>
14  </body>
15
16</html>
```



Linguagem HTML

EFEITOS NO TEXTO

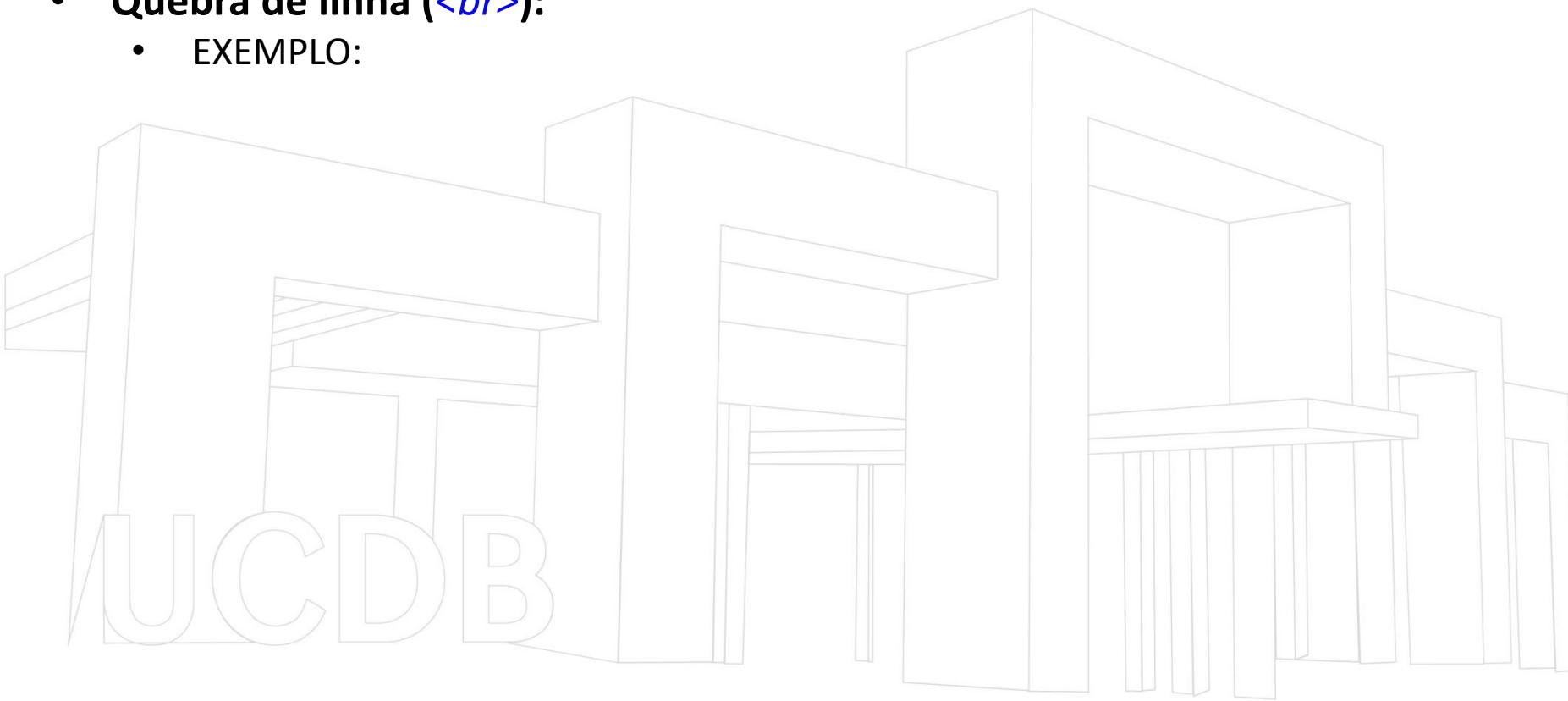
- Quebra de linha (`
`):



Linguagem HTML

EFEITOS NO TEXTO

- Quebra de linha (`
`):
 - EXEMPLO:



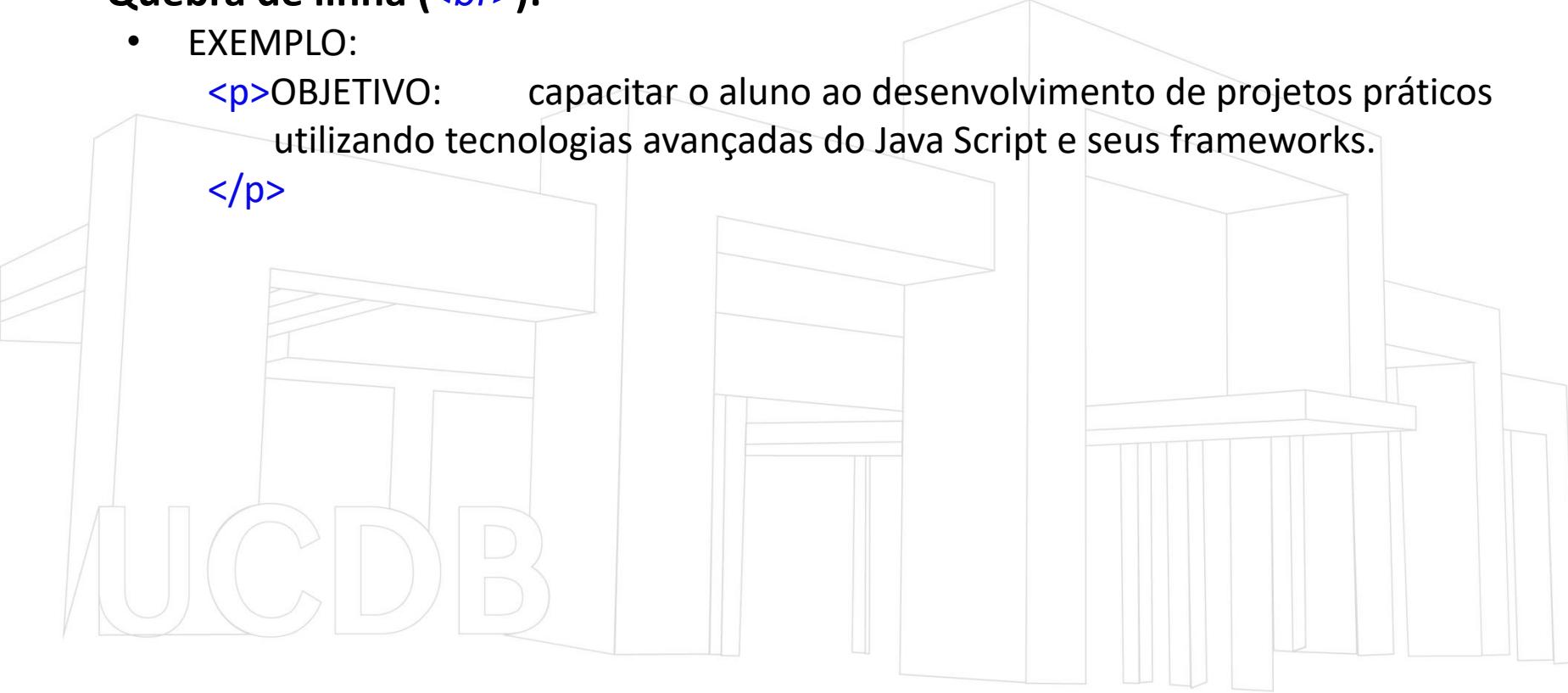
Linguagem HTML

EFEITOS NO TEXTO

- Quebra de linha (`
`):

- EXEMPLO:

`<p>OBJETIVO: capacitar o aluno ao desenvolvimento de projetos práticos
utilizando tecnologias avançadas do Java Script e seus frameworks.
</p>`



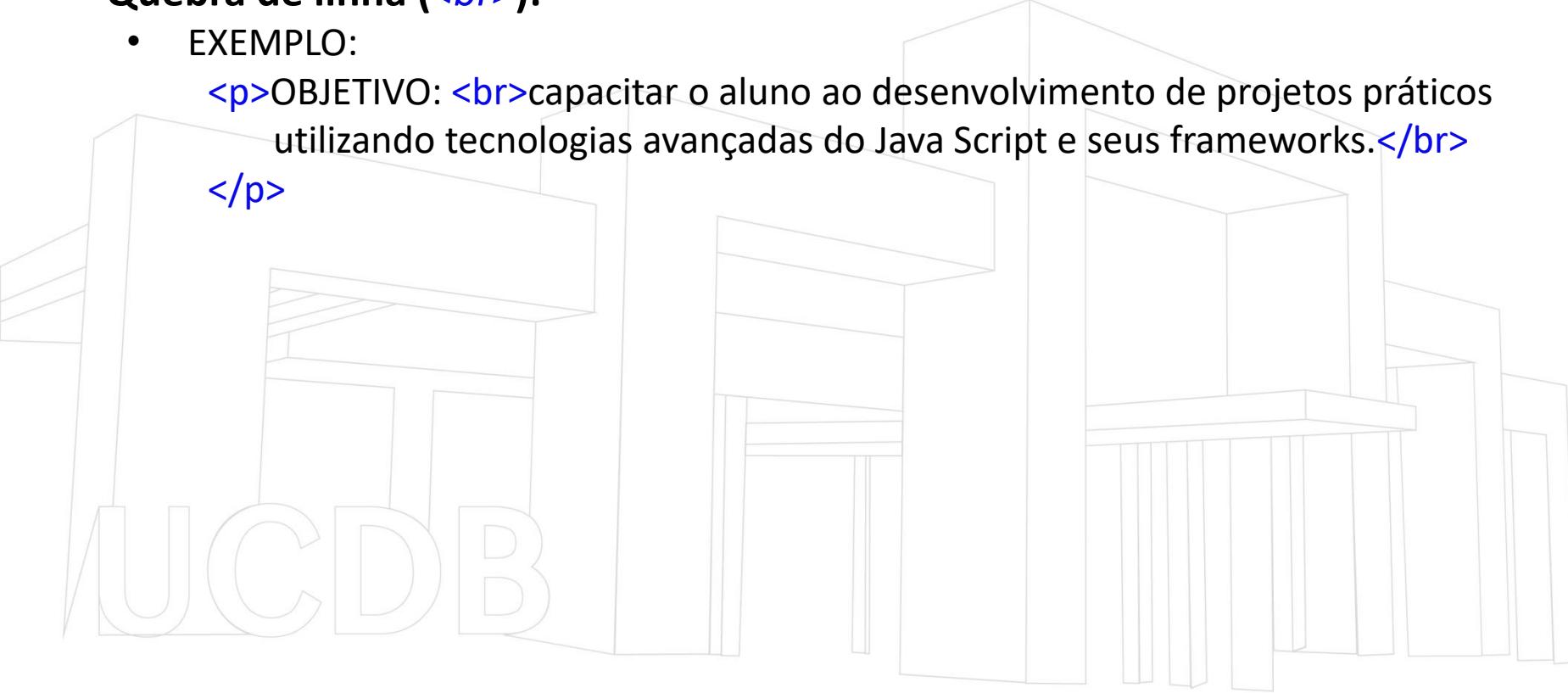
Linguagem HTML

EFEITOS NO TEXTO

- Quebra de linha (`
`):

- EXEMPLO:

```
<p>OBJETIVO: <br>capacitar o aluno ao desenvolvimento de projetos práticos  
utilizando tecnologias avançadas do Java Script e seus frameworks.</br>  
</p>
```



Linguagem HTML

EFEITOS NO TEXTO

- Quebra de linha (`
`):

- EXEMPLO:

```
<p>OBJETIVO: <br>capacitar o aluno ao desenvolvimento de projetos práticos  
utilizando tecnologias avançadas do Java Script e seus frameworks.</br>  
</p>
```

```
index.html X  
  
html > index.html > ...  
1   <html>  
2     <head>  
3       <title> TADS </title>  
4     </head>  
5  
6     <body>  
7       <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>  
8       <p>OBJETIVO: <br>capacitar o aluno ao  
9         desenvolvimento de projetos práticos  
10        utilizando tecnologias avançadas do Java  
11          Script e seus frameworks.</br>  
12        </p>  
13      </body>  
14    </html>
```

Linguagem HTML

EFEITOS NO TEXTO

- Quebra de linha (`
`):

- EXEMPLO:

```
<p>OBJETIVO: <br>capacitar o aluno ao desenvolvimento de projetos práticos  
utilizando tecnologias avançadas do Java Script e seus frameworks.</br>  
</p>
```



```
index.html X  
html > index.html > ...  
1  <html>  
2   <head>  
3     <title> TADS </title>  
4   </head>  
5  
6   <body>  
7     <h1> <b>Ferramentas de Desenvolvimento RAD1</b> </h1>  
8     <p>OBJETIVO: <br>capacitar o aluno ao  
9       desenvolvimento de projetos práticos  
10      utilizando tecnologias avançadas do Java  
11        Script e seus frameworks.</br>  
12     </p>  
13   </body>  
14 </html>
```



Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):



Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):
 - EXEMPLO:



Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):
 - EXEMPLO:

Ferramentas de Desenvolvimento RAD1



Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):

- EXEMPLO:

```
<small><b>Ferramentas de Desenvolvimento RAD1</b></small>
```



Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):

- EXEMPLO:

<small>Ferramentas de Desenvolvimento RAD1</small>

```
index.html ×

html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <p>OBJETIVO: <br>capacitar o aluno ao
9       desenvolvimento de projetos práticos
10      utilizando tecnologias avanças do Java
11      Script e seus frameworks.<br>
12    </p>
13  </body>
14 </html>
```

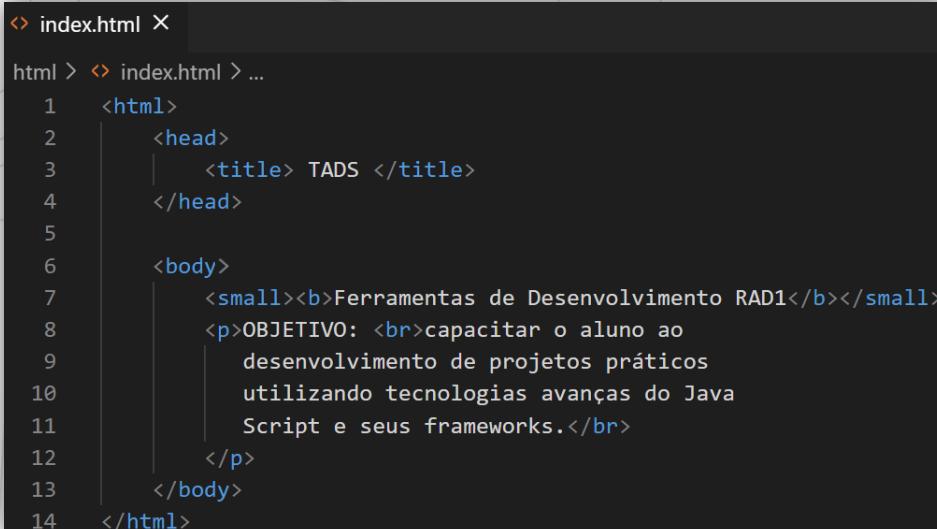
Linguagem HTML

EFEITOS NO TEXTO

- Letra pequena (*<small>*):

- EXEMPLO:

<small>Ferramentas de Desenvolvimento RAD1</small>



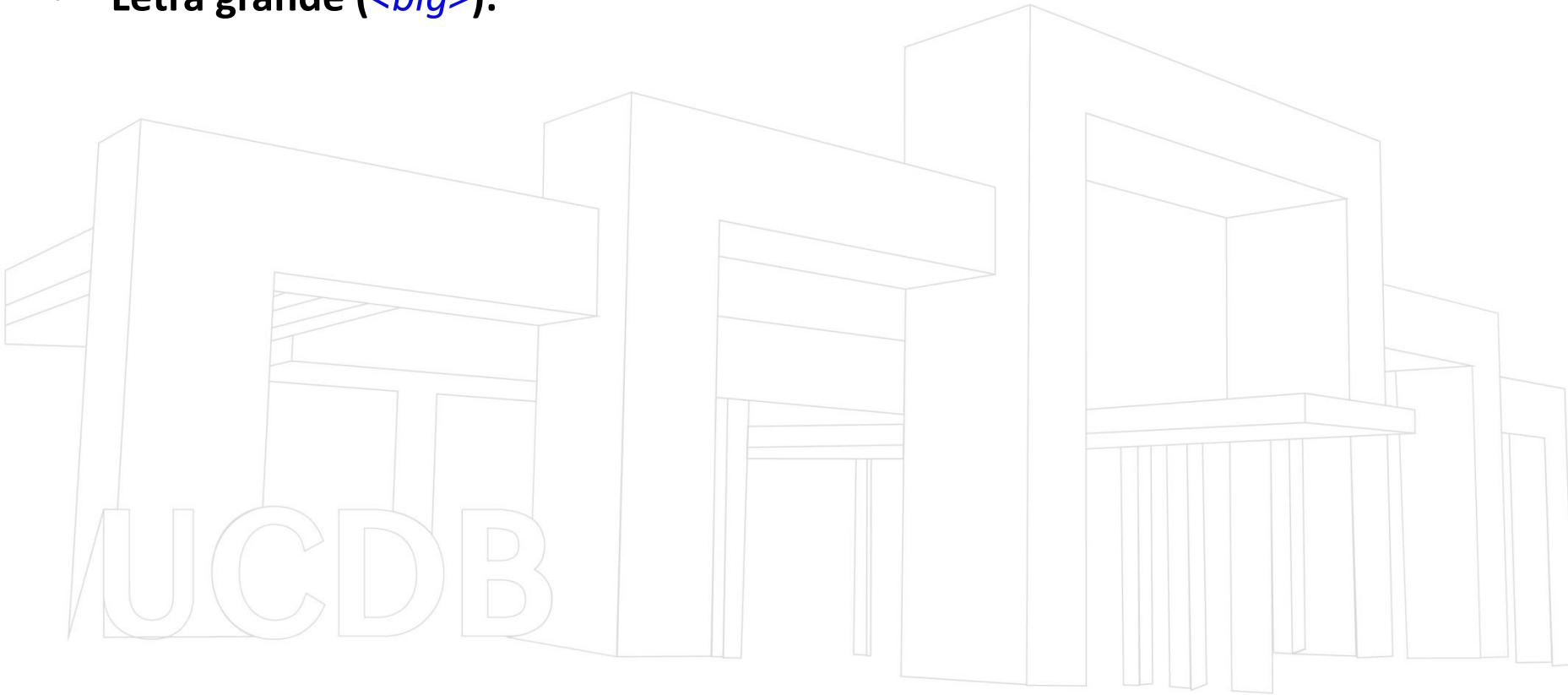
```
index.html ×  
html > index.html > ...  
1  <html>  
2    <head>  
3      <title> TADS </title>  
4    </head>  
5  
6    <body>  
7      <small><b>Ferramentas de Desenvolvimento RAD1</b></small>  
8      <p>OBJETIVO: <br>capacitar o aluno ao  
9        desenvolvimento de projetos práticos  
10       utilizando tecnologias avanças do Java  
11         Script e seus frameworks.<br>  
12      </p>  
13    </body>  
14  </html>
```



Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):



Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):
 - EXEMPLO:



Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):

- EXEMPLO:

Ferramentas de Desenvolvimento RAD1



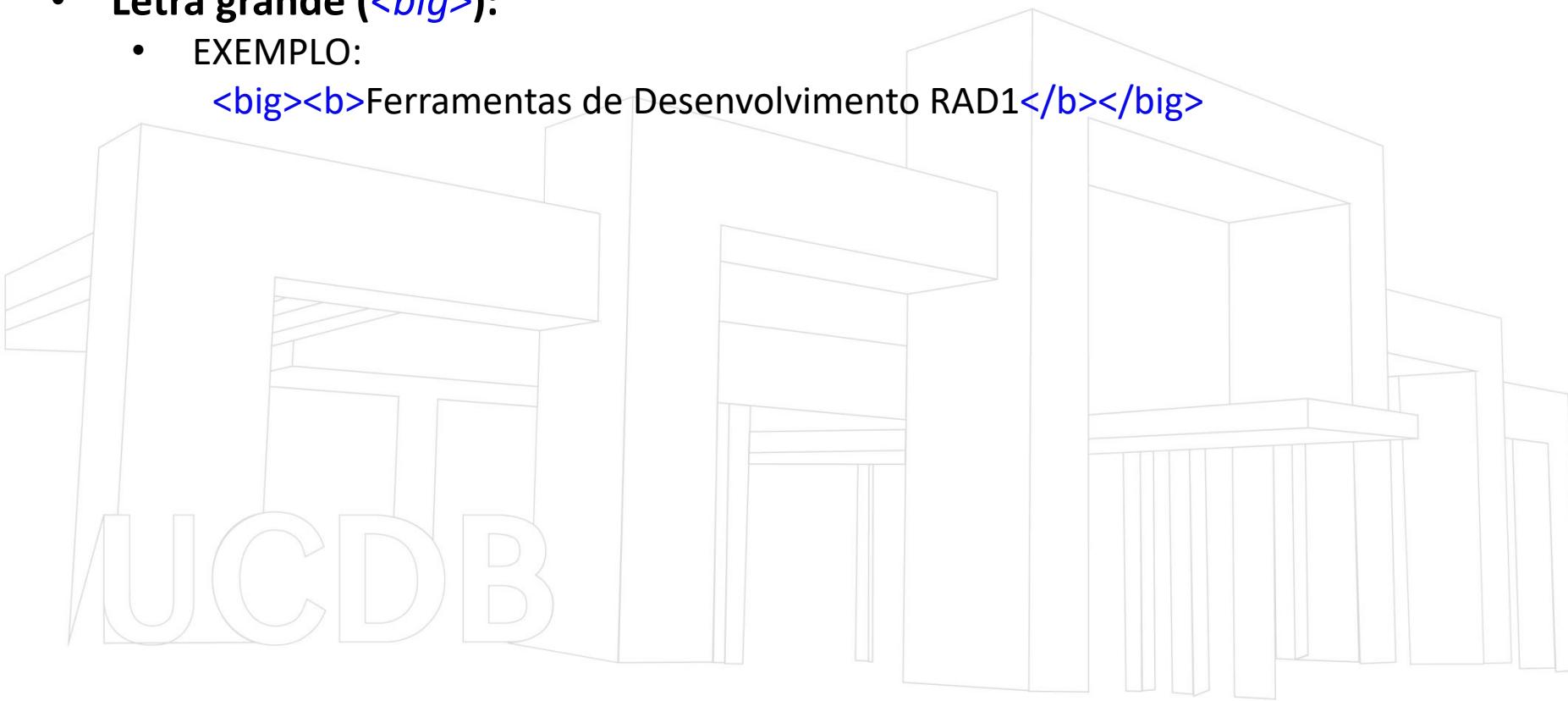
Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):

- EXEMPLO:

```
<big><b>Ferramentas de Desenvolvimento RAD1</b></big>
```



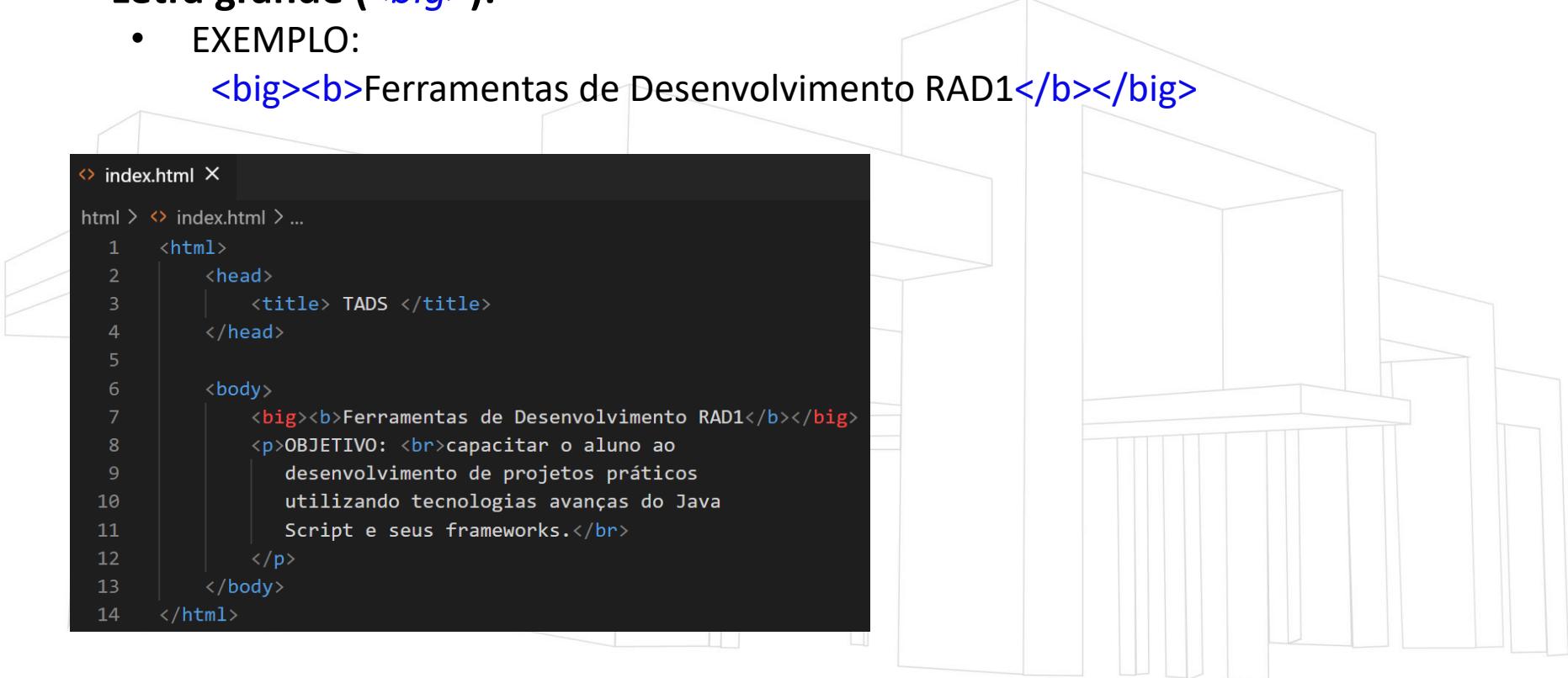
Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):

- EXEMPLO:

```
<big><b>Ferramentas de Desenvolvimento RAD1</b></big>
```



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <big><b>Ferramentas de Desenvolvimento RAD1</b></big>
8       <p>OBJETIVO: <br>capacitar o aluno ao
9         desenvolvimento de projetos práticos
10        utilizando tecnologias avançadas do Java
11          Script e seus frameworks.<br>
12        </p>
13     </body>
14   </html>
```

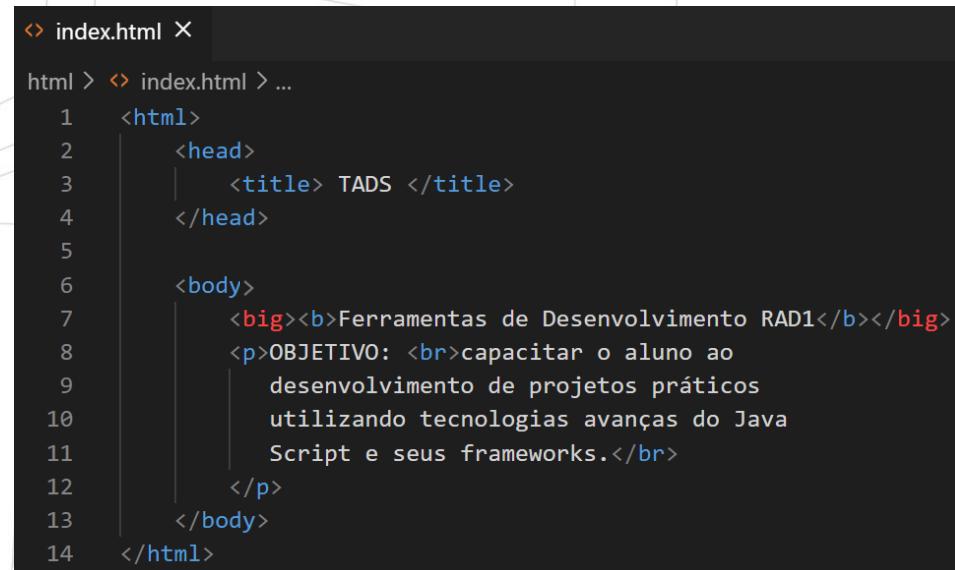
Linguagem HTML

EFEITOS NO TEXTO

- Letra grande (*<big>*):

- EXEMPLO:

```
<big><b>Ferramentas de Desenvolvimento RAD1</b></big>
```



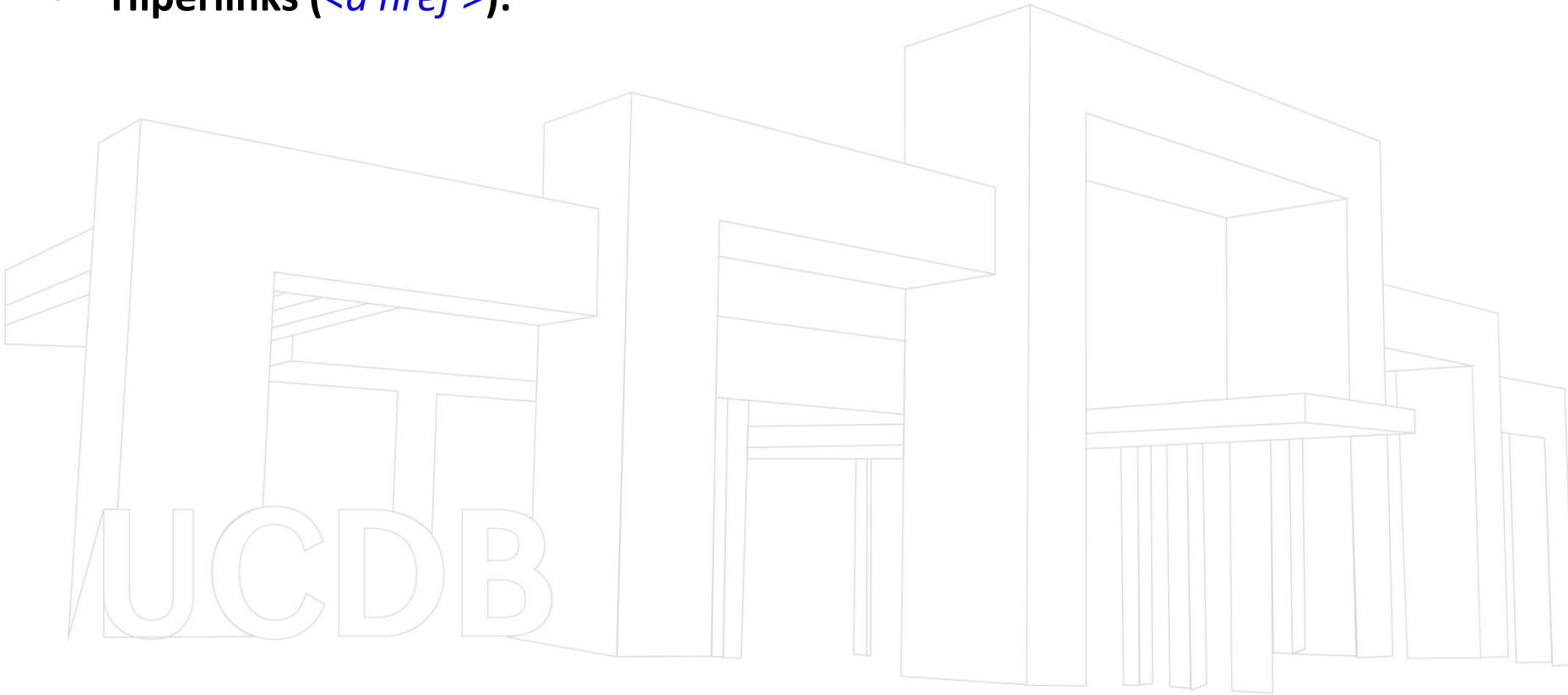
```
index.html x
html > index.html > ...
1  <html>
2   |  <head>
3   |   |  <title> TADS </title>
4   |  </head>
5
6   <body>
7     |  <big><b>Ferramentas de Desenvolvimento RAD1</b></big>
8     |  <p>OBJETIVO: <br>capacitar o aluno ao
9     |    desenvolvimento de projetos práticos
10    |    utilizando tecnologias avanças do Java
11    |    Script e seus frameworks.<br>
12    |  </p>
13   </body>
14 </html>
```



Linguagem HTML

EFEITOS NO TEXTO

- Hiperlinks (*<a href>*):



Linguagem HTML

EFEITOS NO TEXTO

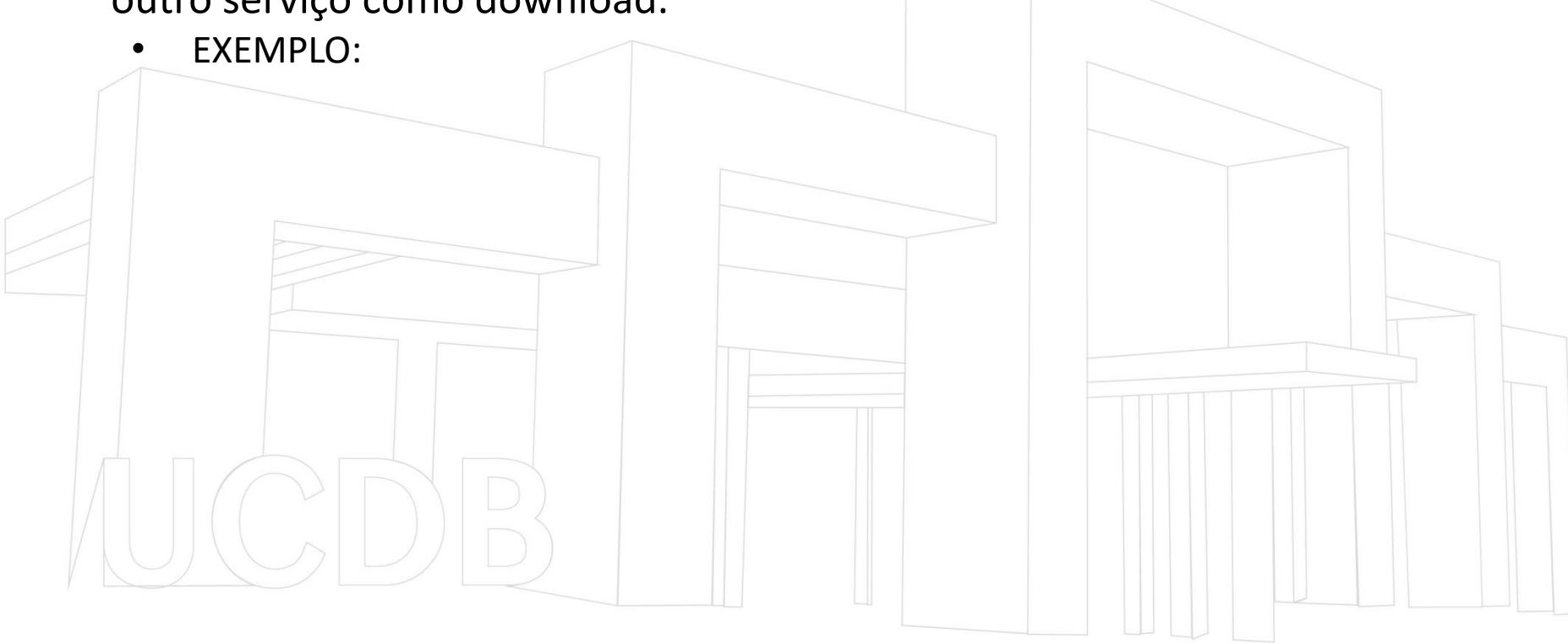
- **Hiperlinks (*<a href>*):** para um outro local, seja uma página, um e-mail ou outro serviço como download.



Linguagem HTML

EFEITOS NO TEXTO

- **Hiperlinks (*<a href>*):** para um outro local, seja uma página, um e-mail ou outro serviço como download.
 - EXEMPLO:



Linguagem HTML

EFEITOS NO TEXTO

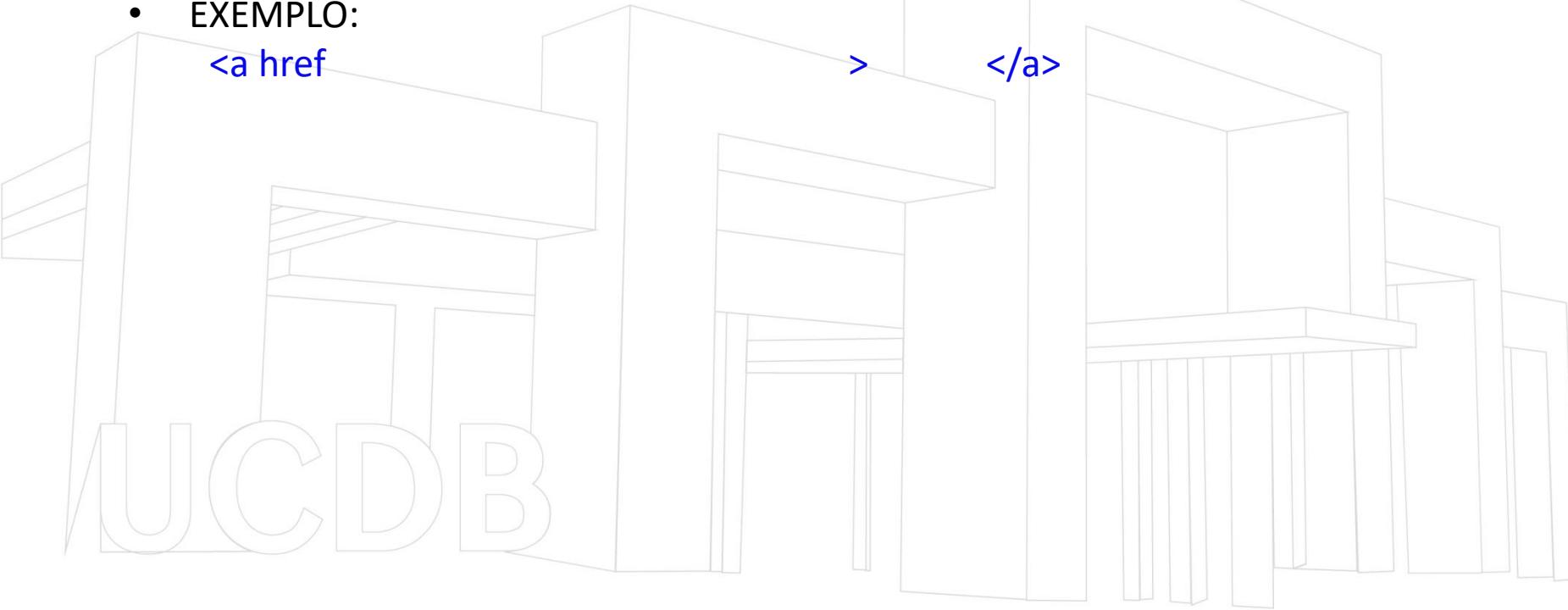
- **Hiperlinks (*< a href >*):** para um outro local, seja uma página, um e-mail ou outro serviço como download.

- EXEMPLO:

< a href

>

**



Linguagem HTML

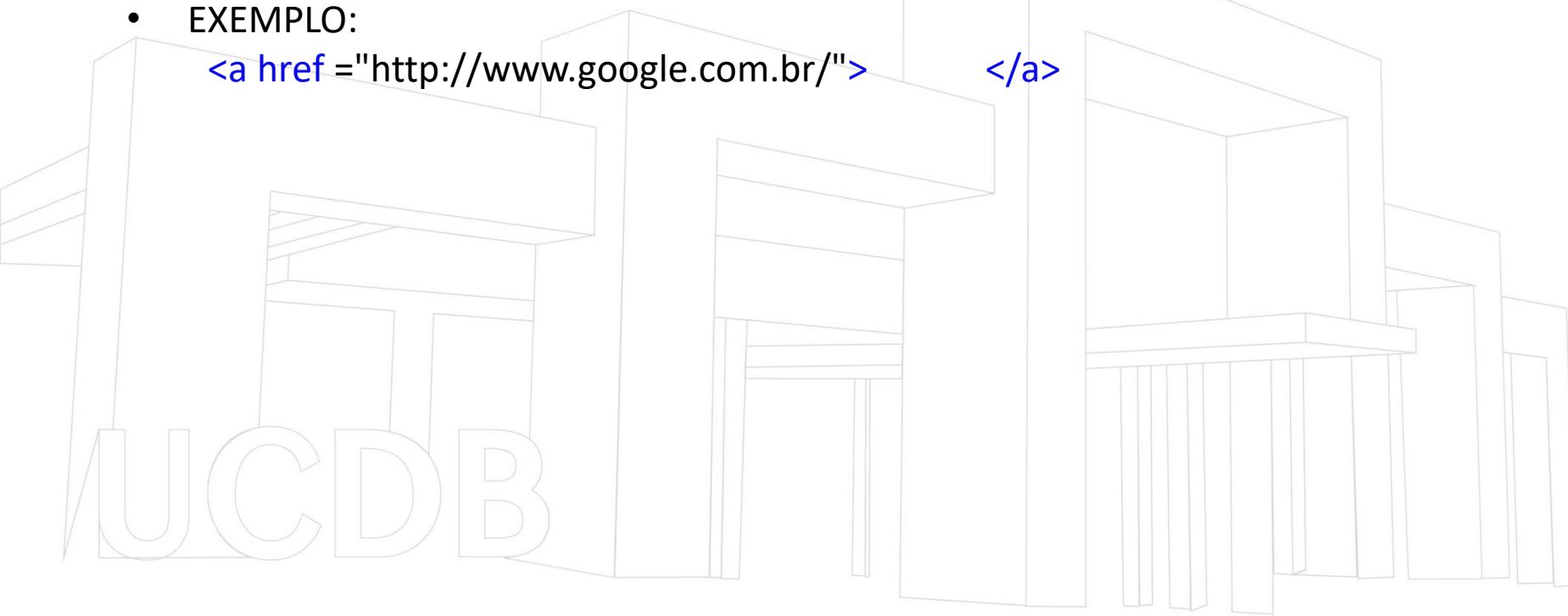
EFEITOS NO TEXTO

- **Hiperlinks (*<a href>*):** para um outro local, seja uma página, um e-mail ou outro serviço como download.

- EXEMPLO:

```
<a href = "http://www.google.com.br/">
```

```
</a>
```



Linguagem HTML

EFEITOS NO TEXTO

- **Hiperlinks (*<a href>*):** para um outro local, seja uma página, um e-mail ou outro serviço como download.

- EXEMPLO:

```
<a href = "http://www.google.com.br/">Google</a>
```



Linguagem HTML

EFEITOS NO TEXTO

- **Hiperlinks ()**: para um outro local, seja uma página, um e-mail ou outro serviço como download.

- EXEMPLO:

```
<a href ="http://www.google.com.br/">Google</a>
```

```
index.html X

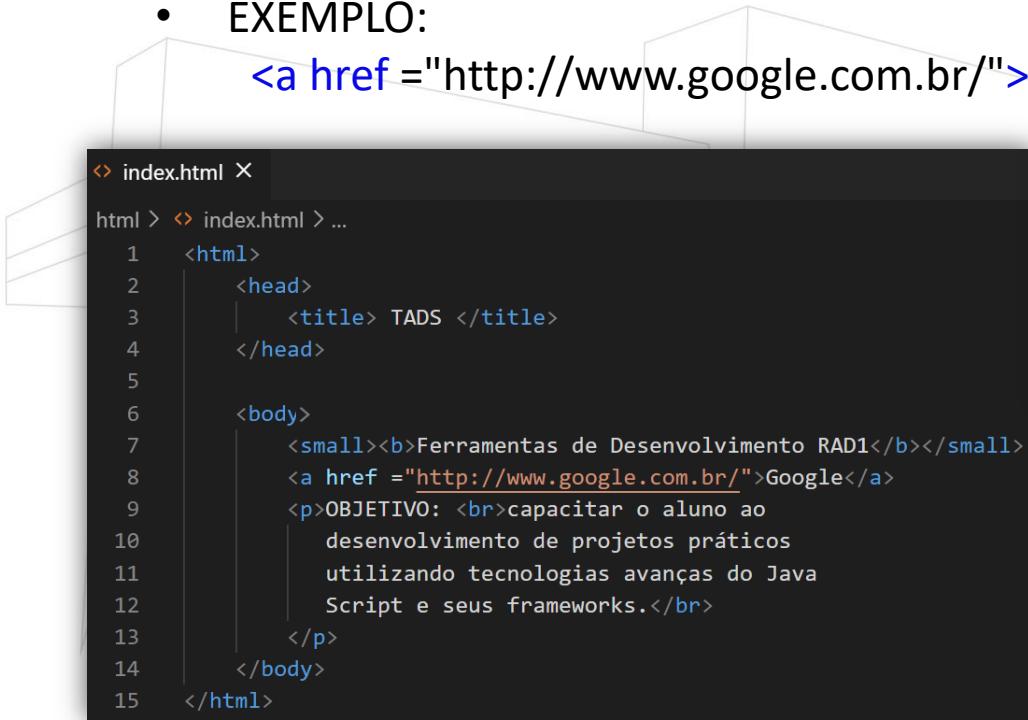
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href ="http://www.google.com.br/">Google</a>
9       <p>OBJETIVO: <br>capacitar o aluno ao
10      | desenvolvimento de projetos práticos
11      | utilizando tecnologias avanças do Java
12      | Script e seus frameworks.<br>
13     </p>
14   </body>
15 </html>
```

Linguagem HTML

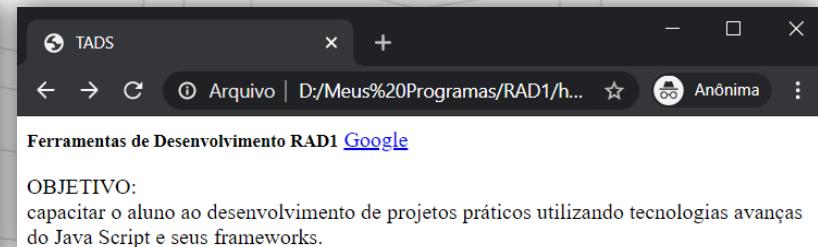
EFEITOS NO TEXTO

- **Hiperlinks ()**: para um outro local, seja uma página, um e-mail ou outro serviço como download.
 - EXEMPLO:

```
<a href ="http://www.google.com.br/">Google</a>
```



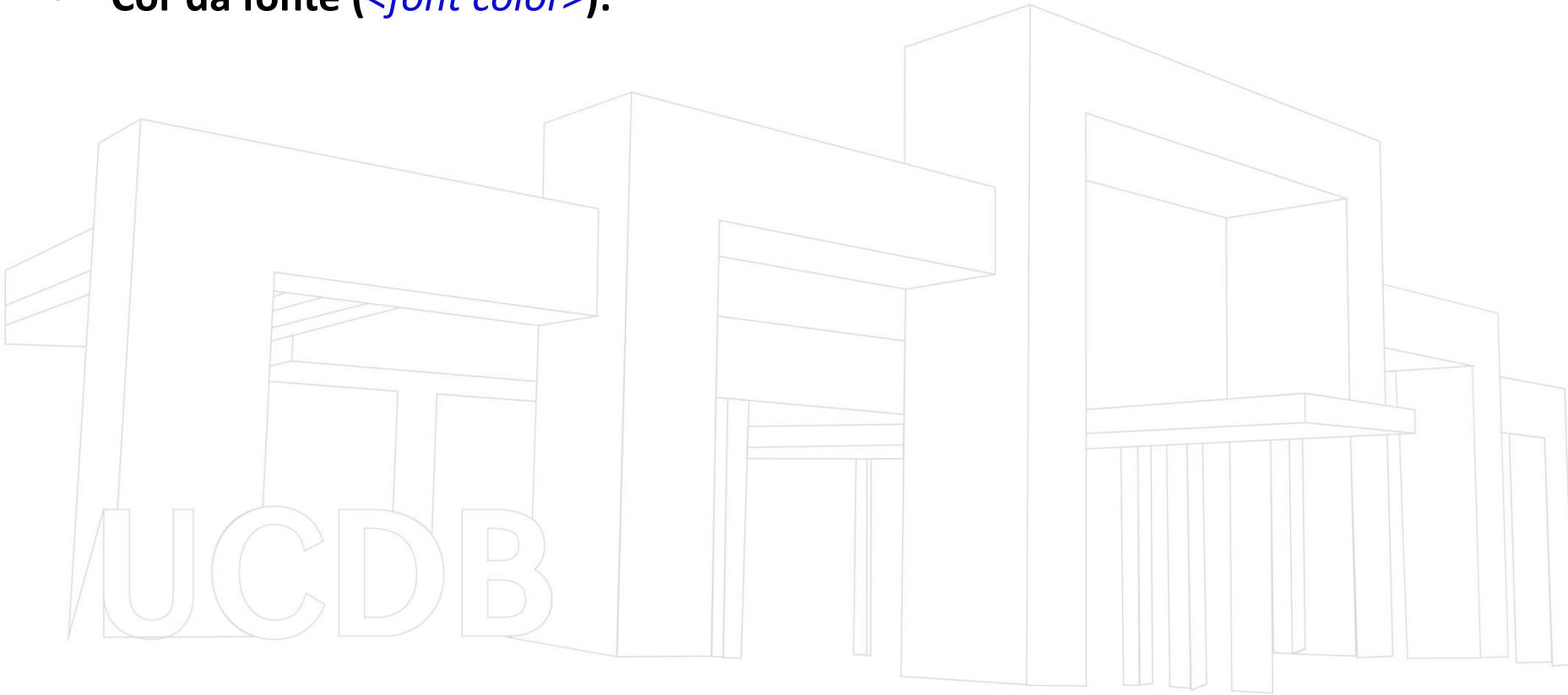
```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href ="http://www.google.com.br/">Google</a>
9       <p>OBJETIVO: <br>capacitar o aluno ao
10      | desenvolvimento de projetos práticos
11      | utilizando tecnologias avanças do Java
12      | Script e seus frameworks.<br>
13     </p>
14   </body>
15 </html>
```



Linguagem HTML

EFEITOS NO TEXTO

- Cor da fonte (**):



Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.



Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.
 - EXEMPLO:

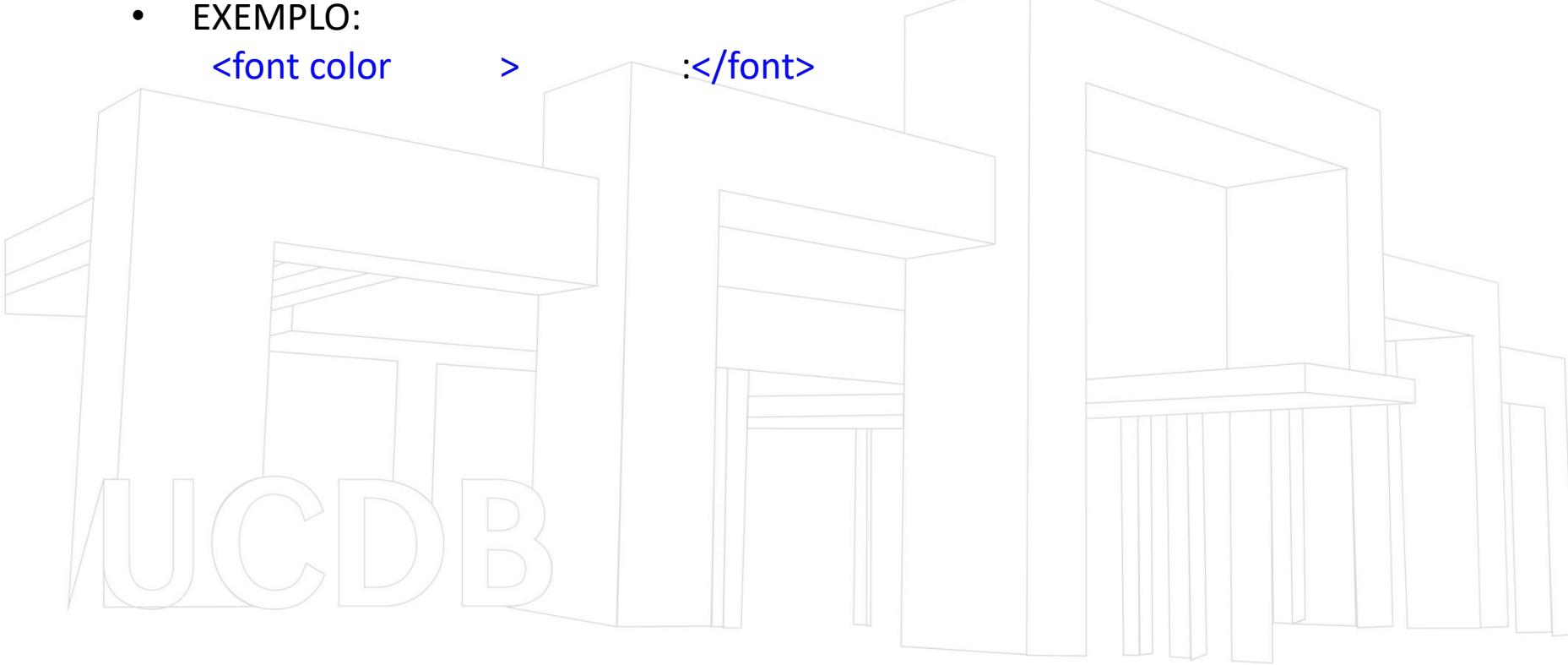


Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**)**: a cor deve ser escrita em inglês.
 - EXEMPLO:

* :



Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.
- EXEMPLO:

```
<font color="red">
```

```
:</font>
```



Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.
- EXEMPLO:

```
<font color="red">OBJETIVO:</font>
```



Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.
- EXEMPLO:

```
<font color="red">OBJETIVO:</font>
```

```
index.html X
html > index.html > ...
1  <html>
2    <head>
3      <title> TADS </title>
4    </head>
5
6    <body>
7      <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8      <a href ="http://www.google.com.br/">Google</a>
9      <p><font color="red">OBJETIVO:</font><br>capacitar o aluno ao
10     desenvolvimento de projetos práticos
11     utilizando tecnologias avanças do Java
12     Script e seus frameworks.<br>
13   </p>
14 </body>
15 </html>>
```

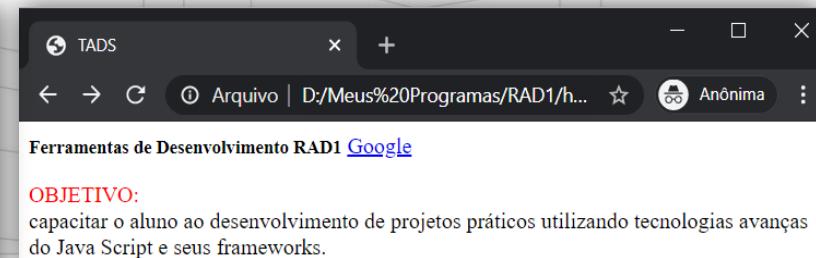
Linguagem HTML

EFEITOS NO TEXTO

- **Cor da fonte (**):** a cor deve ser escrita em inglês.
- EXEMPLO:

OBJETIVO:

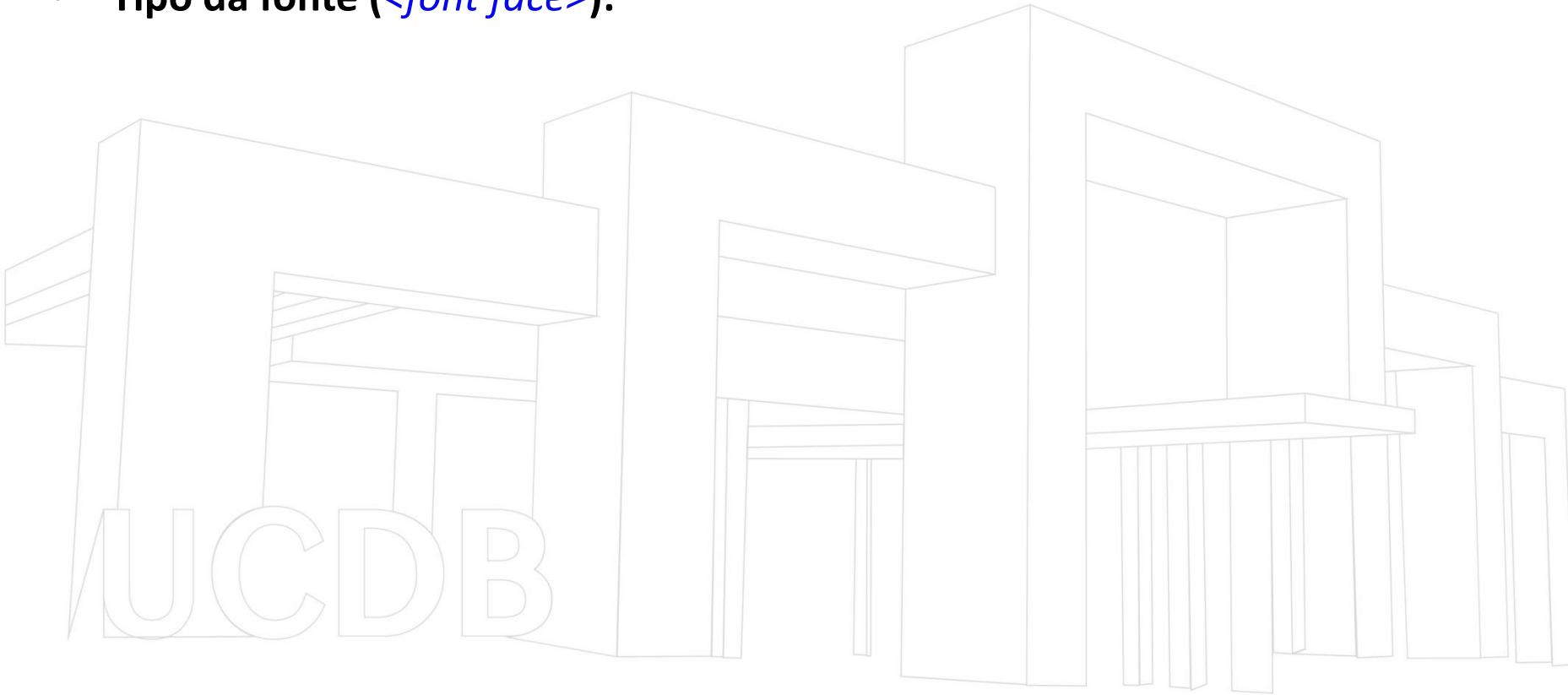
```
index.html X
html > index.html > ...
1  <html>
2    <head>
3      <title> TADS </title>
4    </head>
5
6    <body>
7      <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8      <a href ="http://www.google.com.br/">Google</a>
9      <p><font color="red">OBJETIVO:</font><br>capacitar o aluno ao
10     desenvolvimento de projetos práticos
11     utilizando tecnologias avanças do Java
12     Script e seus frameworks.<br>
13   </p>
14 </body>
15 </html>>
```



Linguagem HTML

EFEITOS NO TEXTO

- Tipo da fonte (**):



Linguagem HTML

EFEITOS NO TEXTO

- **Tipo da fonte (**):**
 - EXEMPLO:



Linguagem HTML

EFEITOS NO TEXTO

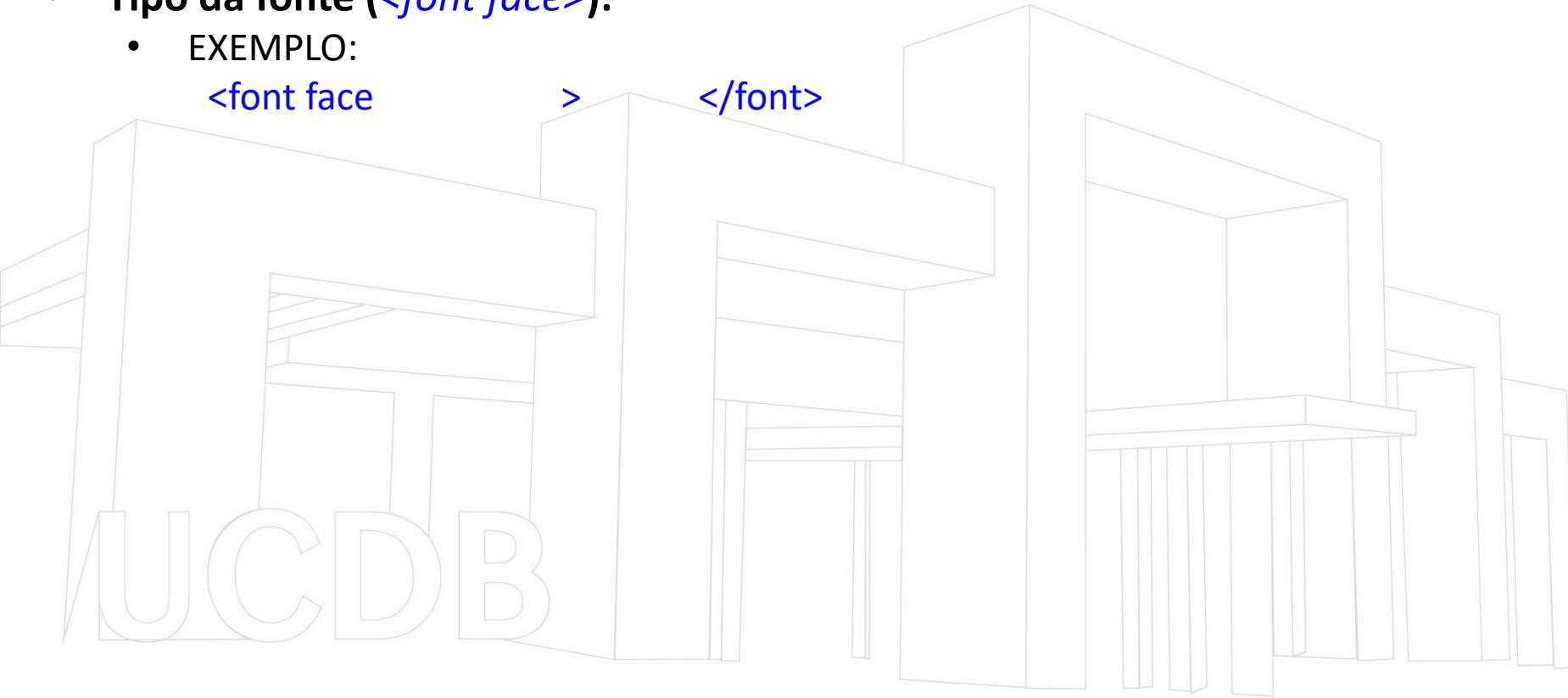
- **Tipo da fonte (**):**

- EXEMPLO:

- `<font face`

- `>`

- ``



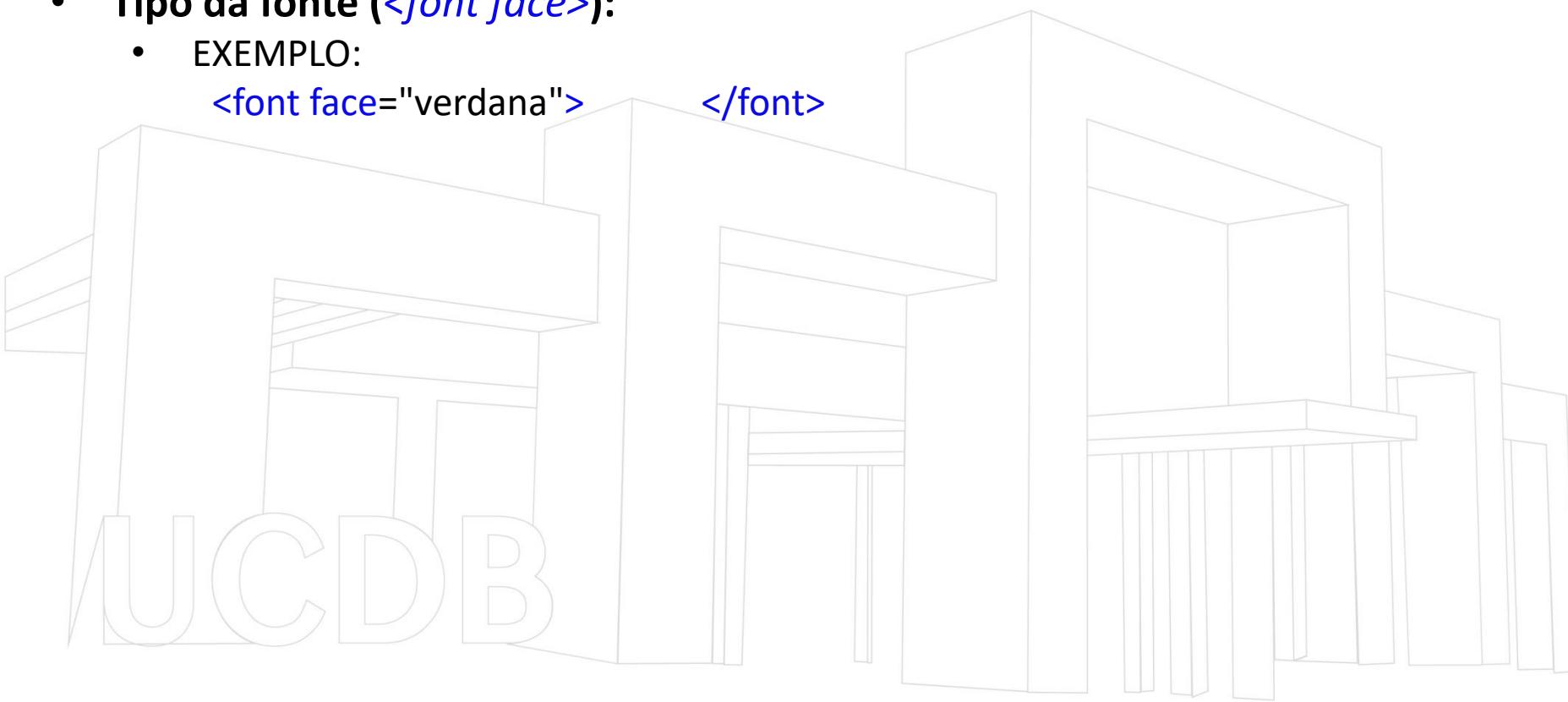
Linguagem HTML

EFEITOS NO TEXTO

- **Tipo da fonte (**):**

- EXEMPLO:

```
<font face="verdana"> </font>
```



Linguagem HTML

EFEITOS NO TEXTO

- **Tipo da fonte (**):**

- EXEMPLO:

```
<font face="verdana">TEXTO:</font>
```



Linguagem HTML

EFEITOS NO TEXTO

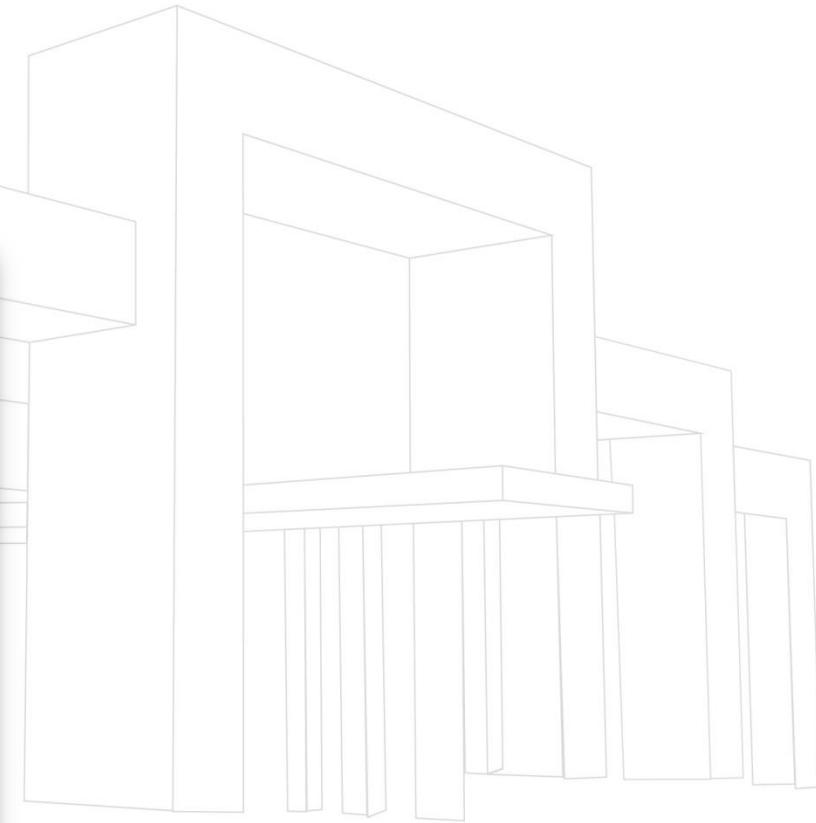
- **Tipo da fonte (**):**

- EXEMPLO:

```
<font face="verdana">TEXTO:</font>
```

```
index.html X

html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href ="http://www.google.com.br/">Google</a>
9       <p><font color="red">OBJETIVO:</font>
10      <font face="verdana"><br>capacitar o aluno ao
11          desenvolvimento de projetos práticos
12          utilizando tecnologias avanças do Java
13          Script e seus frameworks.<br></font>
14      </p>
15    </body>
16  </html>>
```



Linguagem HTML

EFEITOS NO TEXTO

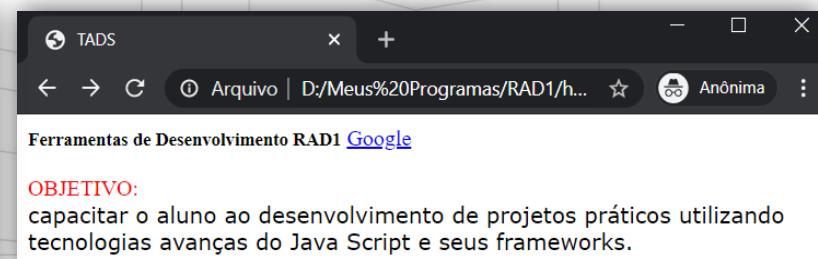
- **Tipo da fonte (**):**

- EXEMPLO:

```
<font face="verdana">TEXTO:</font>
```



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href ="http://www.google.com.br/">Google</a>
9       <p><font color="red">OBJETIVO:</font>
10      <font face="verdana"><br>capacitar o aluno ao
11      desenvolvimento de projetos práticos
12      utilizando tecnologias avanças do Java
13      Script e seus frameworks.<br></font>
14    </p>
15  </body>
16 </html>>
```



Linguagem HTML

EFEITOS NO TEXTO

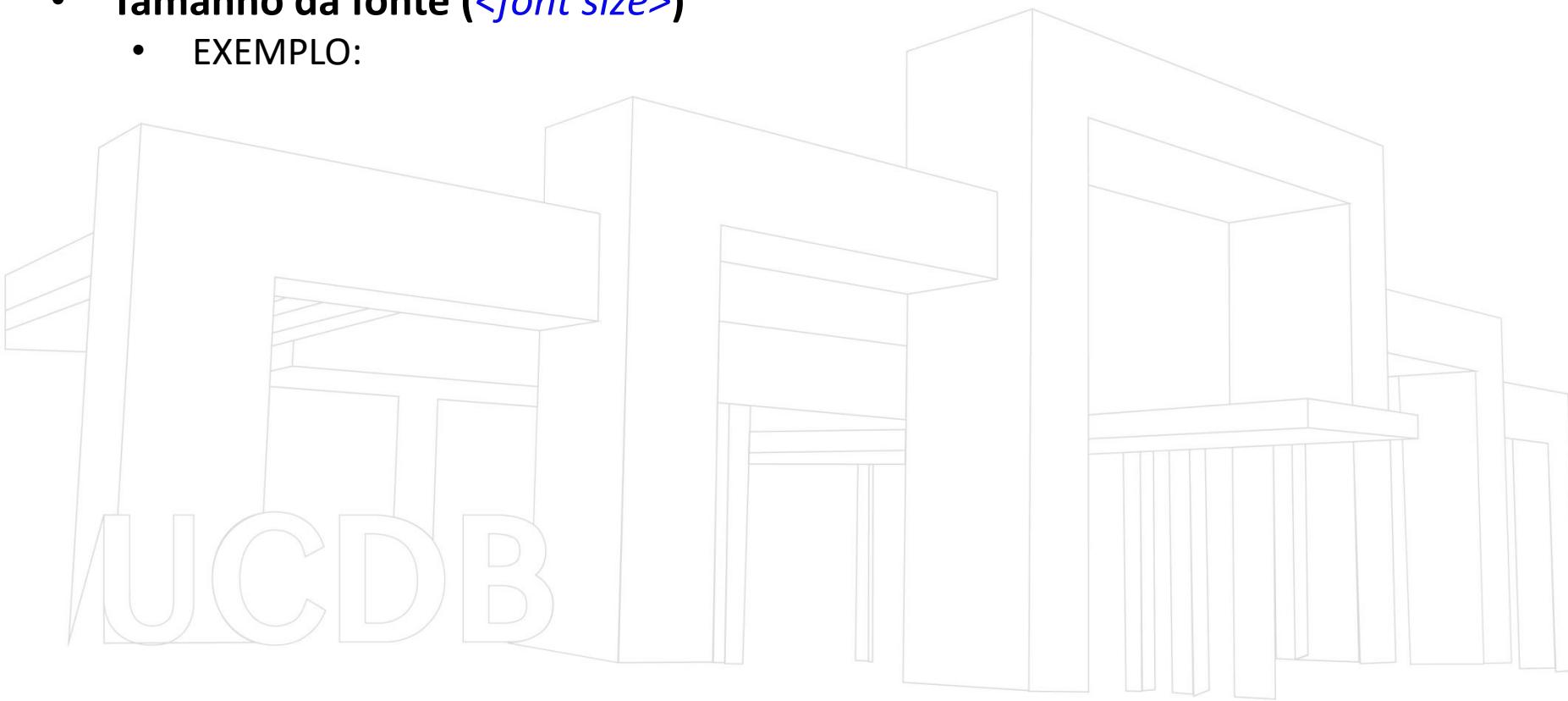
- Tamanho da fonte (**)



Linguagem HTML

EFEITOS NO TEXTO

- Tamanho da fonte (**)
 - EXEMPLO:



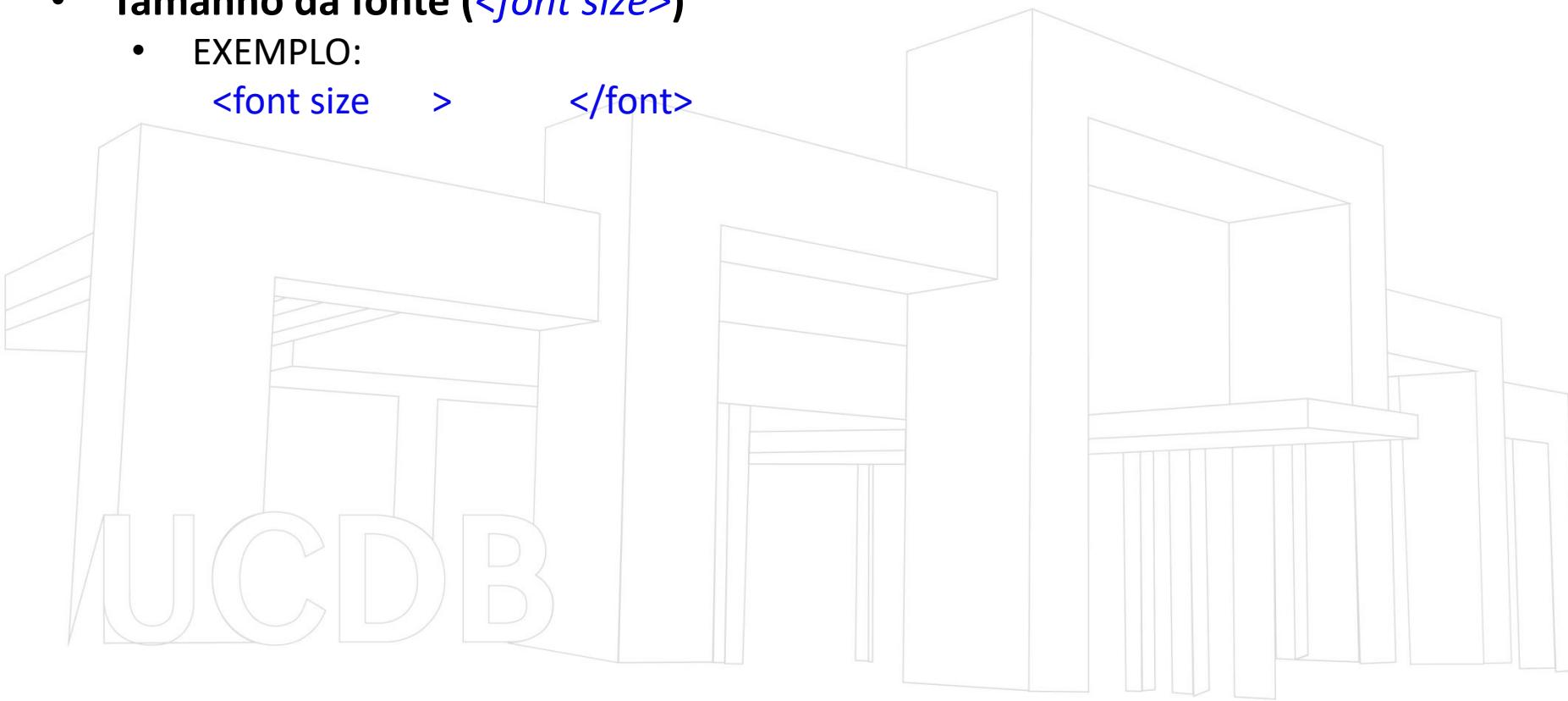
Linguagem HTML

EFEITOS NO TEXTO

- Tamanho da fonte (**)

- EXEMPLO:

```
<font size      >      </font>
```



Linguagem HTML

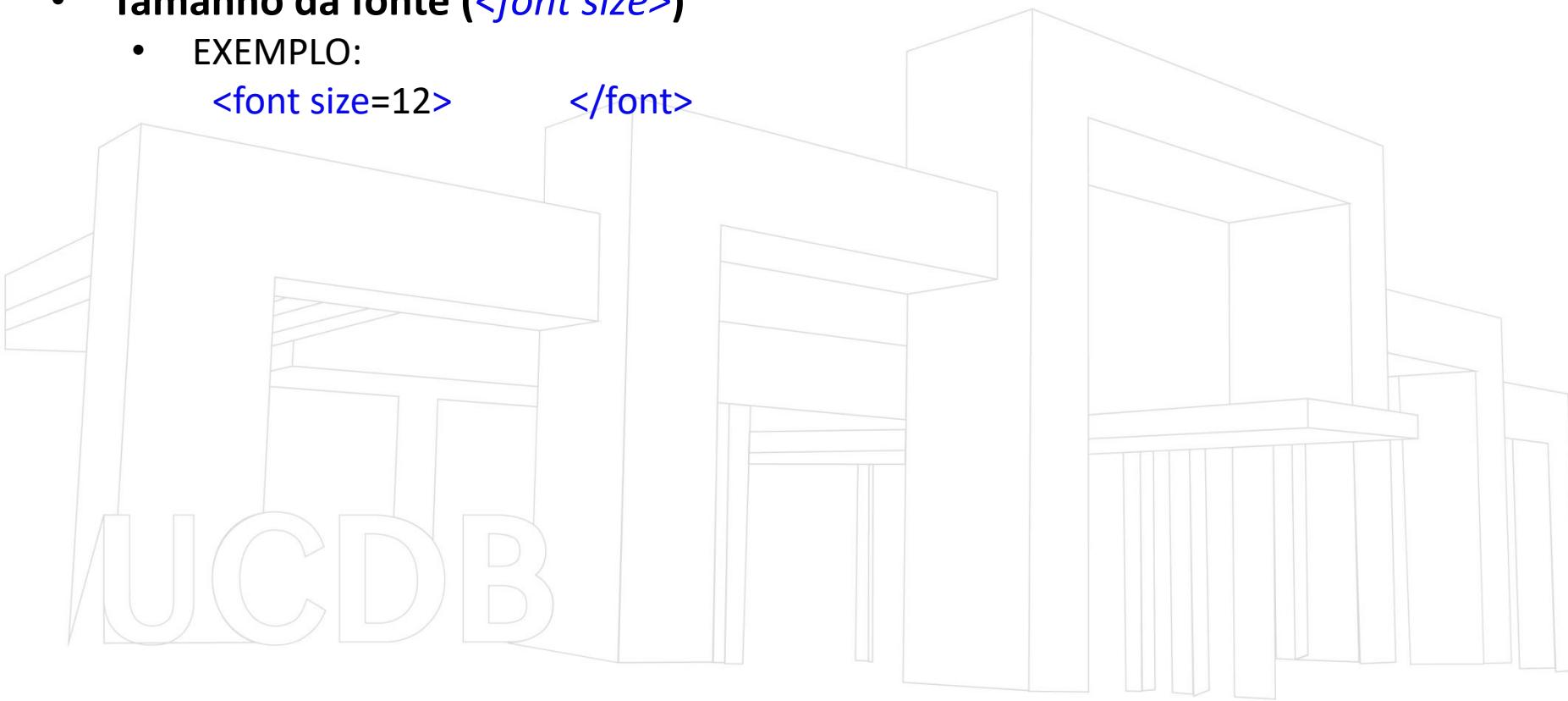
EFEITOS NO TEXTO

- Tamanho da fonte (**)

- EXEMPLO:

**

**

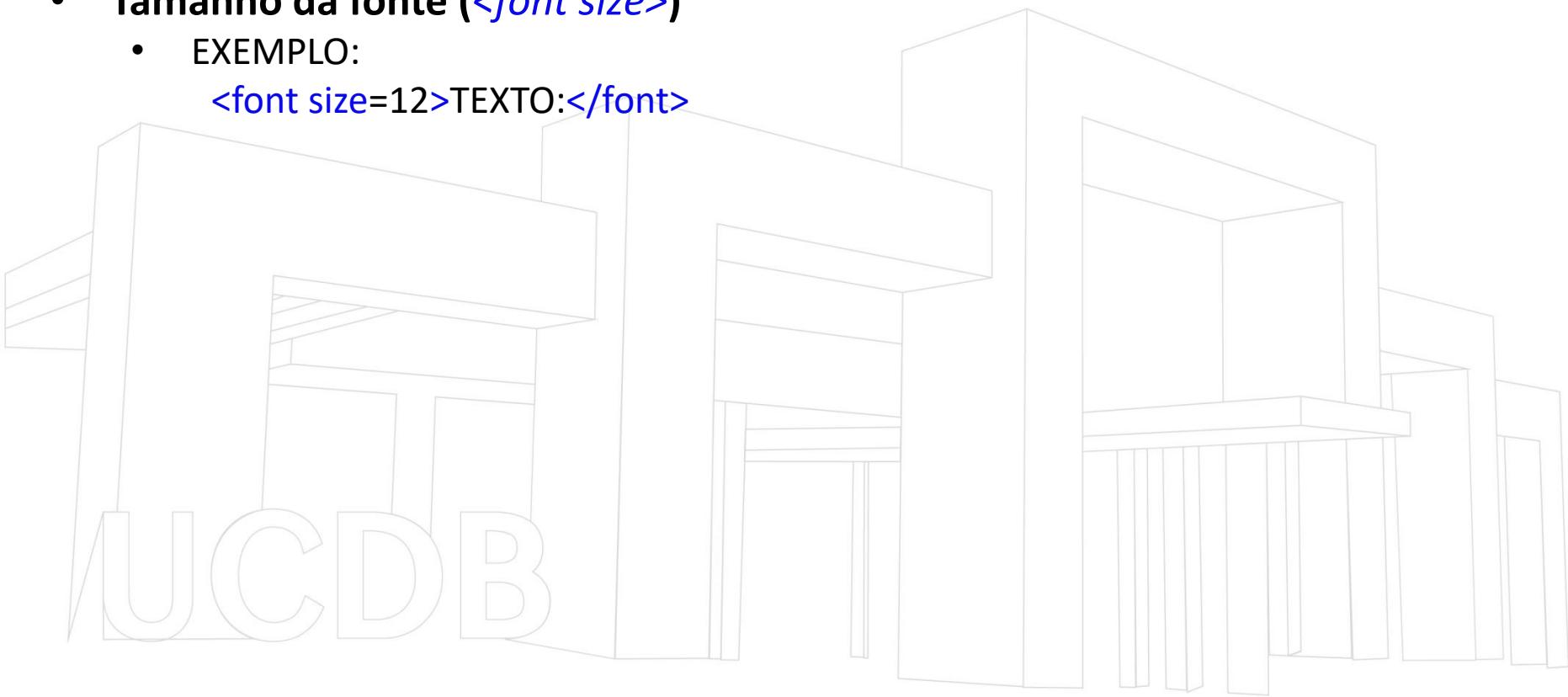


Linguagem HTML

EFEITOS NO TEXTO

- Tamanho da fonte (**)
 - EXEMPLO:

TEXTO:



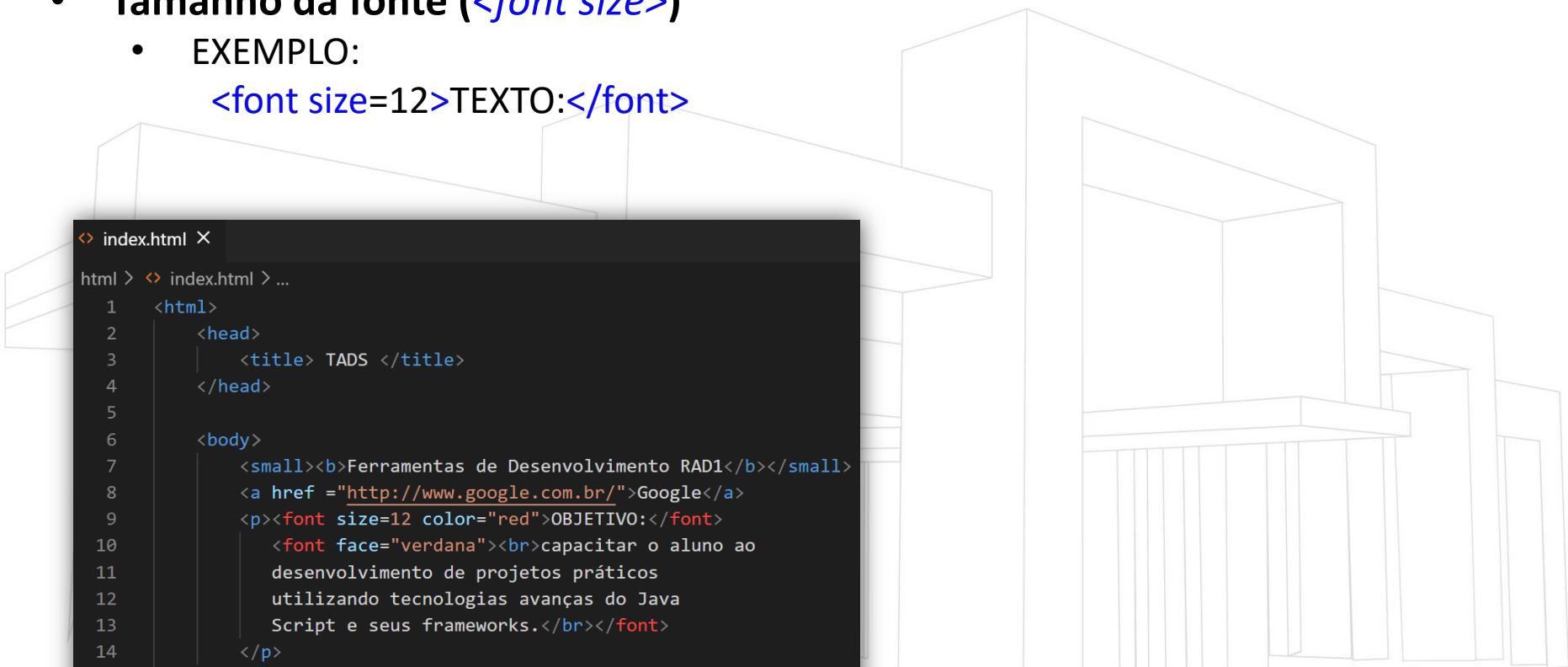
Linguagem HTML

EFEITOS NO TEXTO

- Tamanho da fonte (**)

- EXEMPLO:

```
<font size=12>TEXTO:</font>
```



```
index.html ×
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href = "http://www.google.com.br/">Google</a>
9       <p><font size=12 color="red">OBJETIVO:</font>
10      <font face="verdana"><br>capacitar o aluno ao
11        desenvolvimento de projetos práticos
12        utilizando tecnologias avanças do Java
13        Script e seus frameworks.<br></font>
14      </p>
15    </body>
16  </html>
```

Linguagem HTML

EFEITOS NO TEXTO

- Tamanho da fonte (**)

- EXEMPLO:

```
<font size=12>TEXTO:</font>
```



```
index.html X
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6     <body>
7       <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8       <a href = "http://www.google.com.br/">Google</a>
9       <p><font size=12 color="red">OBJETIVO:</font>
10      <font face="verdana"><br>capacitar o aluno ao
11        desenvolvimento de projetos práticos
12        utilizando tecnologias avanças do Java
13        Script e seus frameworks.<br></font>
14      </p>
15    </body>
16  </html>
```



Linguagem HTML

IMAGENS NA PÁGINA

- tag ``



Linguagem HTML

IMAGENS NA PÁGINA

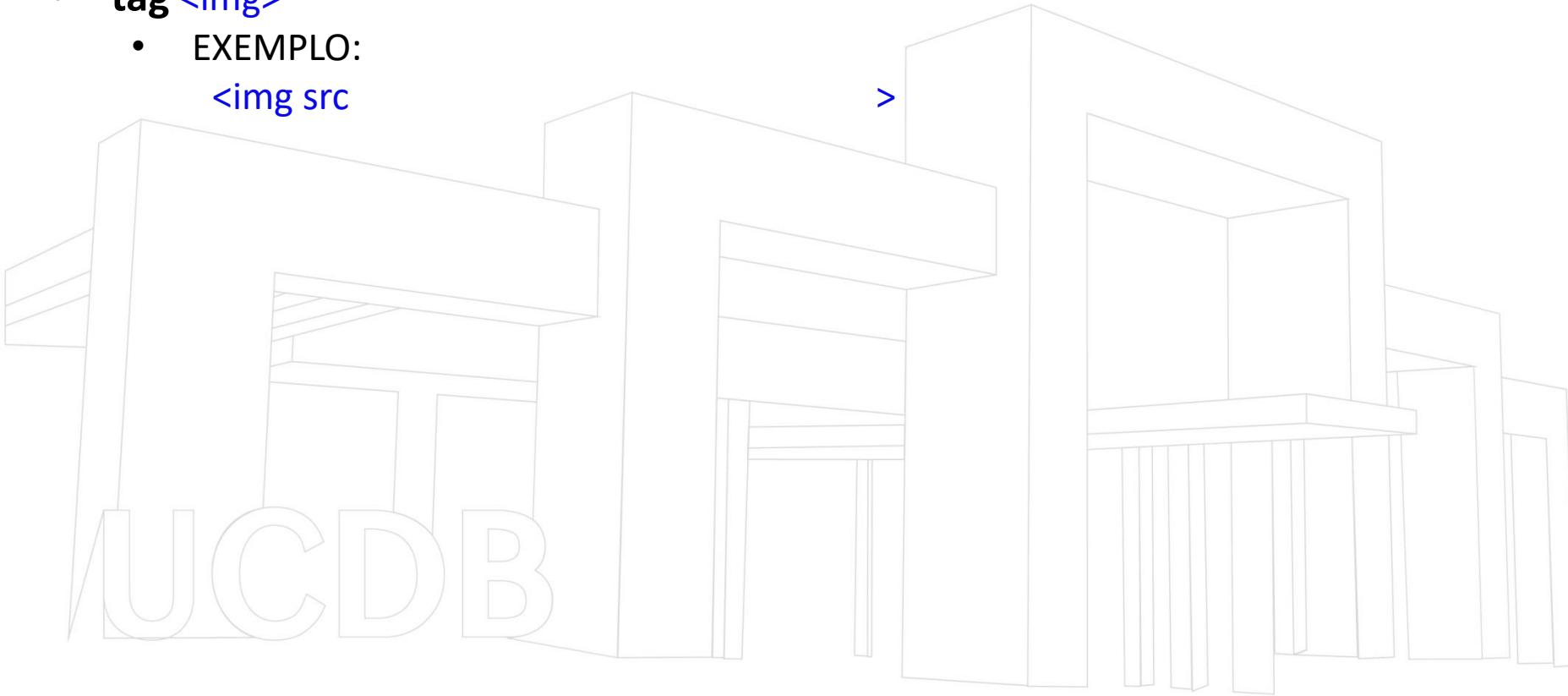
- tag ``
 - EXEMPLO:



Linguagem HTML

IMAGENS NA PÁGINA

- **tag ``**
 - EXEMPLO:
`<img src`



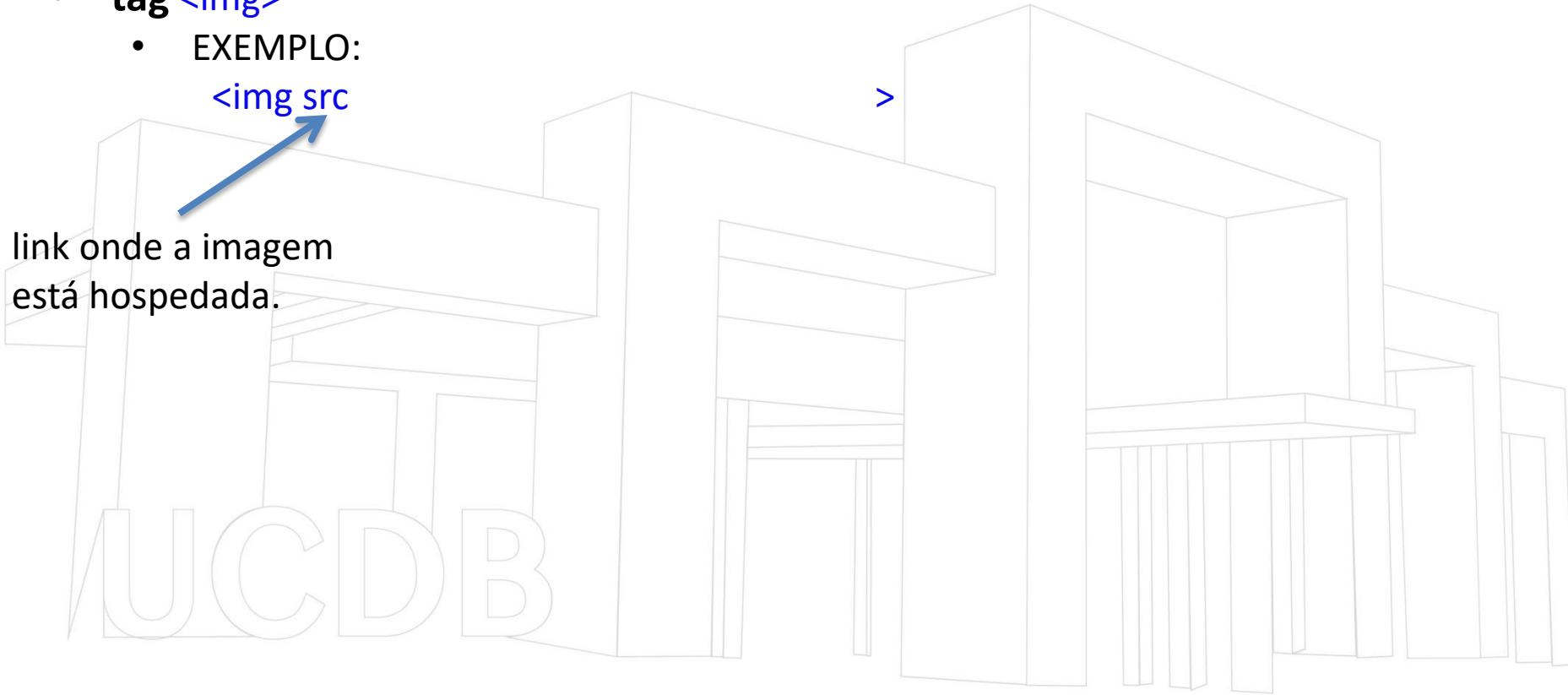
Linguagem HTML

IMAGENS NA PÁGINA

- **tag ``**
 - EXEMPLO:

`<img src`

link onde a imagem
está hospedada.

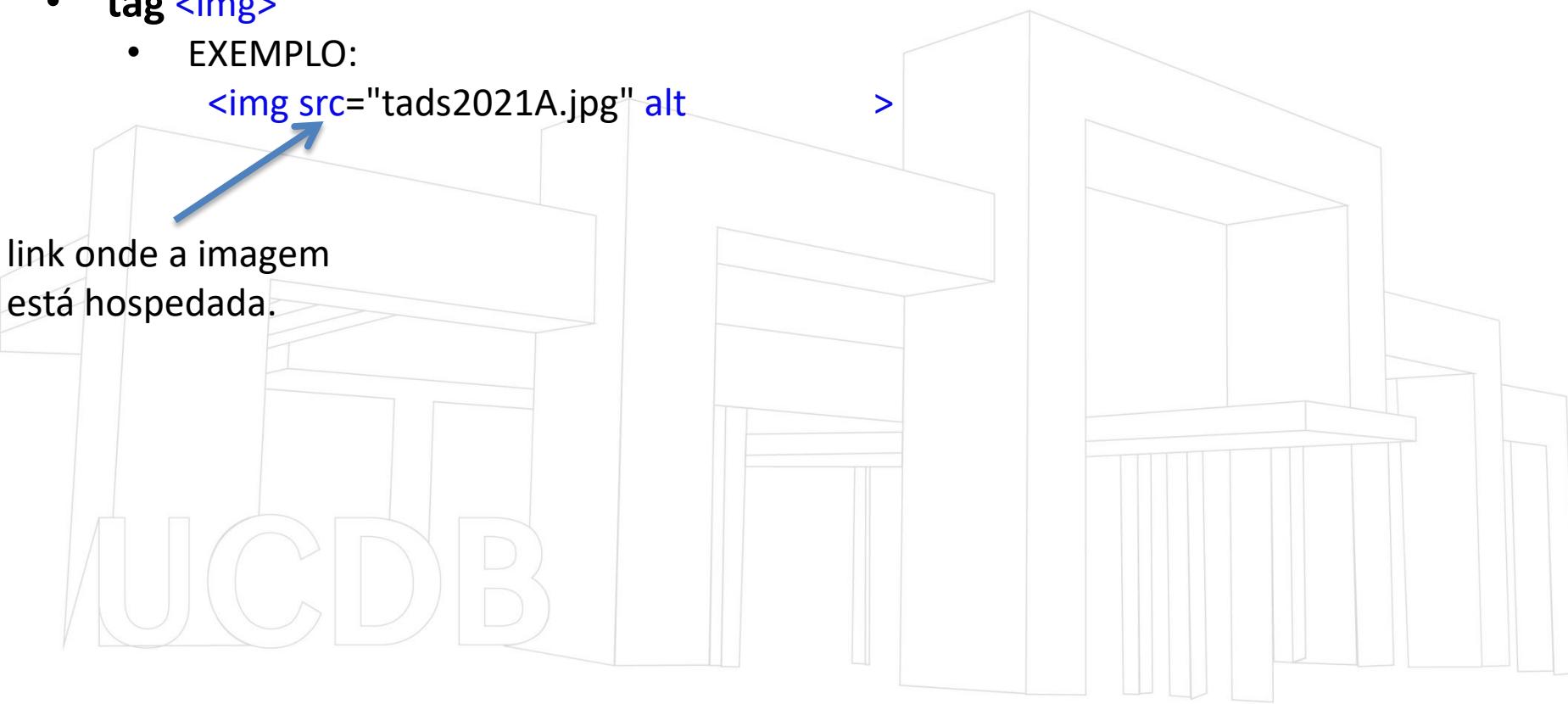


Linguagem HTML

IMAGENS NA PÁGINA

- **tag ``**
 - EXEMPLO:

``**
 - EXEMPLO:
``



Linguagem HTML

IMAGENS NA PÁGINA

- **tag ``**
 - EXEMPLO:

``**
 - EXEMPLO:

```

```

link onde a imagem
está hospedada.

texto alternativo:

espaço reservado para a descrição textual daquela palavra. Este atributo é muito importante para que os mecanismos de busca, como o Google, possam compreender do que uma imagem se trata, ou acessibilidade.

Linguagem HTML

IMAGENS NA PÁGINA

- tag ``
 - EXEMPLO:

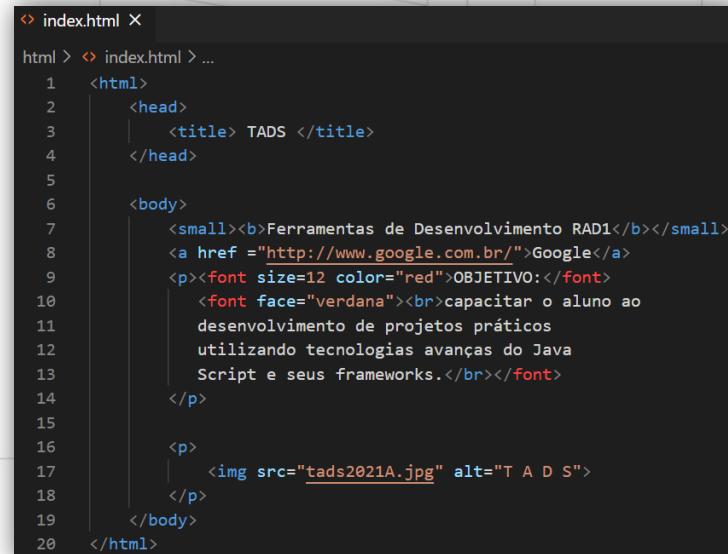
```

```

link onde a imagem
está hospedada.

texto alternativo:

espaço reservado para a descrição textual daquela palavra. Este atributo é muito importante para que os mecanismos de busca, como o Google, possam compreender do que uma imagem se trata, ou acessibilidade.



```
index.html X
html > index.html ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD</b></small>
8     <a href = "http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11      desenvolvimento de projetos práticos
12      utilizando tecnologias avanças do Java
13      Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      
18    </p>
19  </body>
20</html>
```

Linguagem HTML

IMAGENS NA PÁGINA

- tag ``

- EXEMPLO:

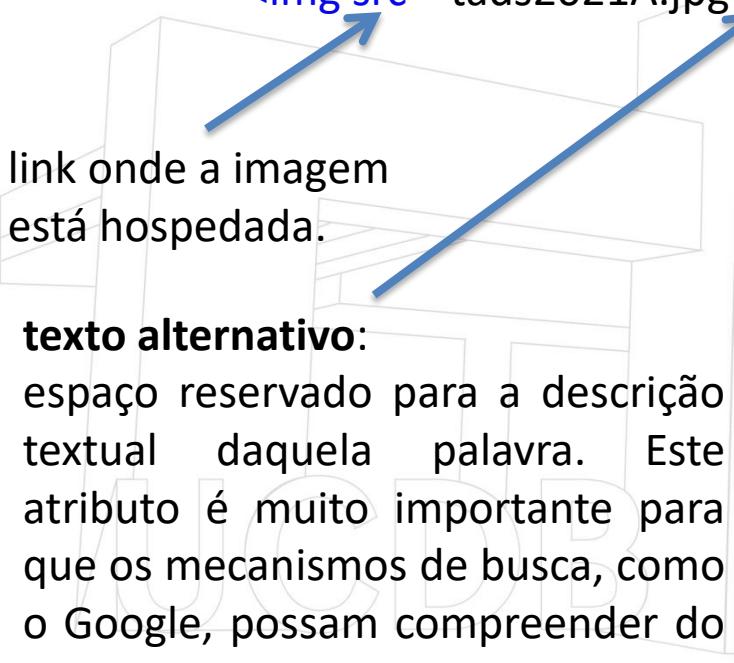
```

```

link onde a imagem
está hospedada.

texto alternativo:

espaço reservado para a descrição textual daquela palavra. Este atributo é muito importante para que os mecanismos de busca, como o Google, possam compreender do que uma imagem se trata, ou acessibilidade.



The screenshot shows a browser window with the title 'IADS'. The main content area displays a page with the text 'OBJETIVO:' and '2021A CURSO DE GRADUAÇÃO ANÁLISE E DESENVOLVIMENTO DE SISTEMAS'. Below this, there is some descriptive text about the course's objectives and a small gear icon. To the left of the browser, the developer tools are open, showing the file 'index.html' with the following code:

```
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <a href = "http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11        desenvolvimento de projetos práticos
12        utilizando tecnologias avançadas do Java
13        Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      
18    </p>
19  </body>
20 </html>
```

Linguagem HTML

LISTA ORDENADA



Linguagem HTML

LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (`` e ``).



Linguagem HTML

LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (`` e ``).
- **tag ``:** lista os elementos;



Linguagem HTML

LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (`` e ``).

- tag ``:** lista os elementos;
- tag ``:** ordena os elementos.

Linguagem HTML

LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (`` e ``).

- tag ``:** lista os elementos;
- tag ``:** ordena os elementos.

EXEMPLO:

UCDB

Linguagem HTML

LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (`` e ``).

- tag ``:** lista os elementos;
- tag ``:** ordena os elementos.

EXEMPLO:

```
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
  <li>React</li>
</ol>
```

Linguagem HTML

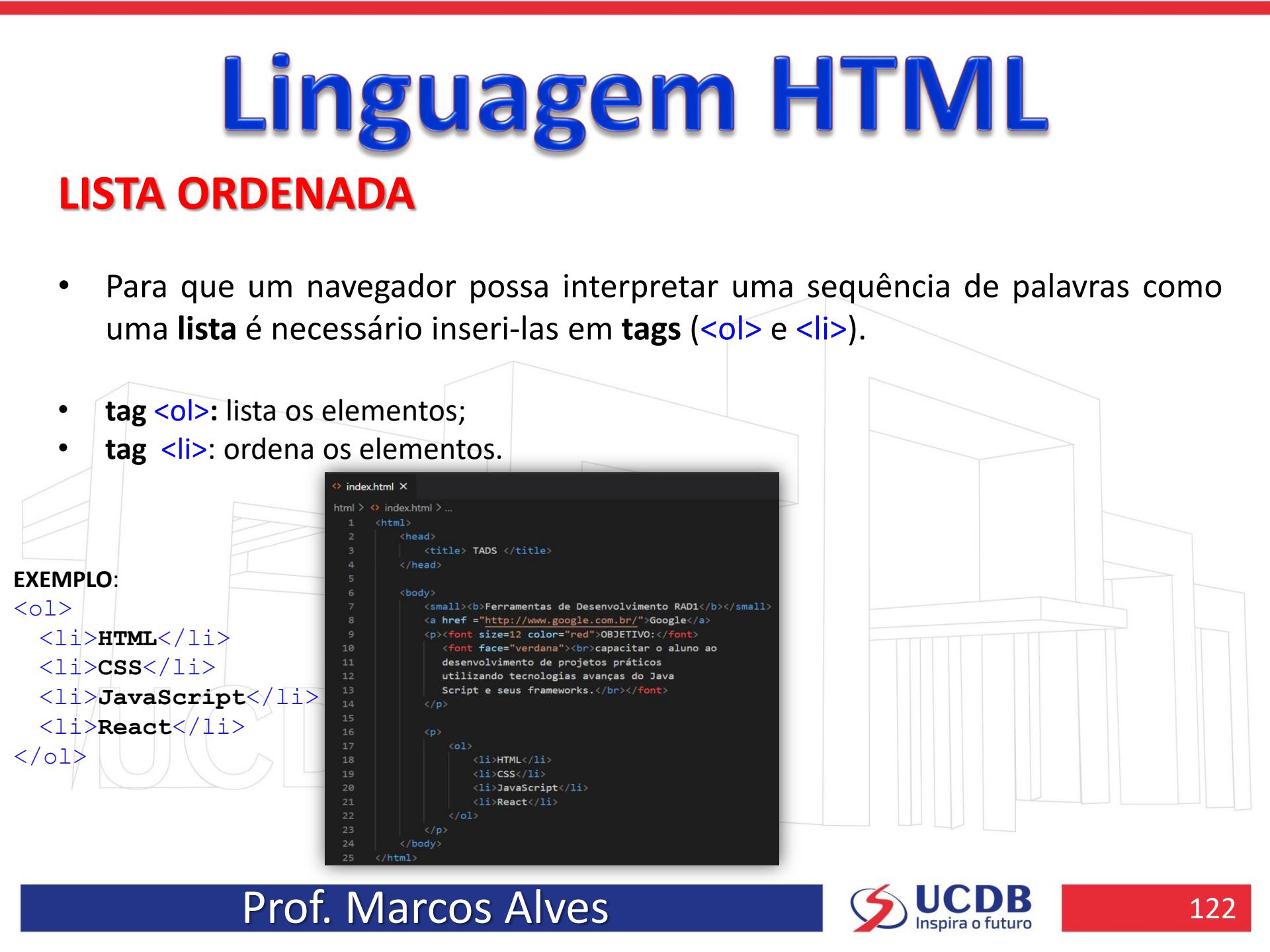
LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (**** e ****).

- tag **: lista os elementos;
- tag **: ordena os elementos.

EXEMPLO:

```
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
  <li>React</li>
</ol>
```



```
index.html x
html > index.html ...
1  <html>
2   <head>
3     |   <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <a href ="http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11      desenvolvimento de projetos práticos
12      utilizando tecnologias avanças do Java
13      Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      <ol>
18        <li>HTML</li>
19        <li>CSS</li>
20        <li>JavaScript</li>
21        <li>React</li>
22      </ol>
23    </p>
24  </body>
25 </html>
```

Linguagem HTML

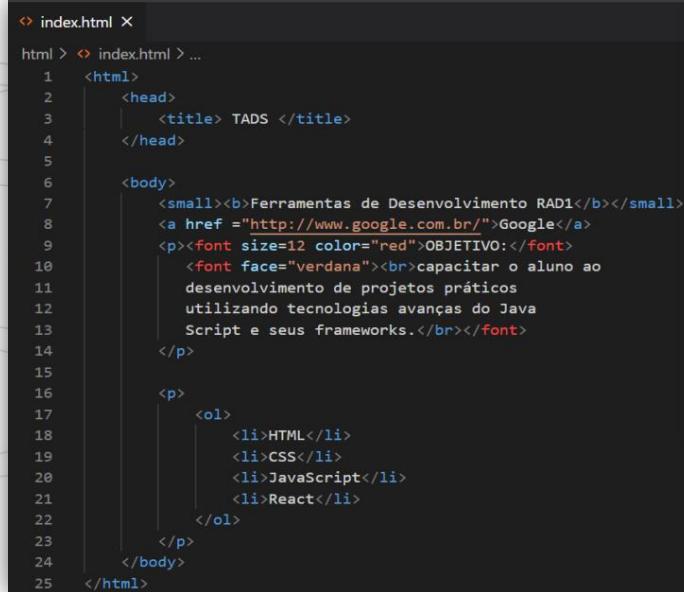
LISTA ORDENADA

- Para que um navegador possa interpretar uma sequência de palavras como uma **lista** é necessário inseri-las em **tags** (**** e ****).

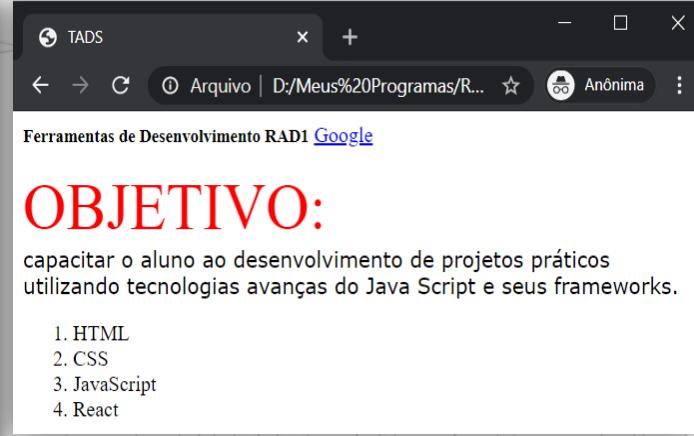
- tag **: lista os elementos;
- tag **: ordena os elementos.

EXEMPLO:

```
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
  <li>React</li>
</ol>
```



```
index.html x
html > index.html > ...
1  <html>
2   |  <head>
3   |   |  <title> TADS </title>
4   |  </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <a href ="http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11      desenvolvimento de projetos práticos
12      utilizando tecnologias avanças do Java
13      Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      <ol>
18        <li>HTML</li>
19        <li>CSS</li>
20        <li>JavaScript</li>
21        <li>React</li>
22      </ol>
23    </p>
24  </body>
25 </html>
```



Linguagem HTML

LISTA NÃO ORDENADA



Linguagem HTML

LISTA NÃO ORDENADA

- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.



Linguagem HTML

LISTA NÃO ORDENADA

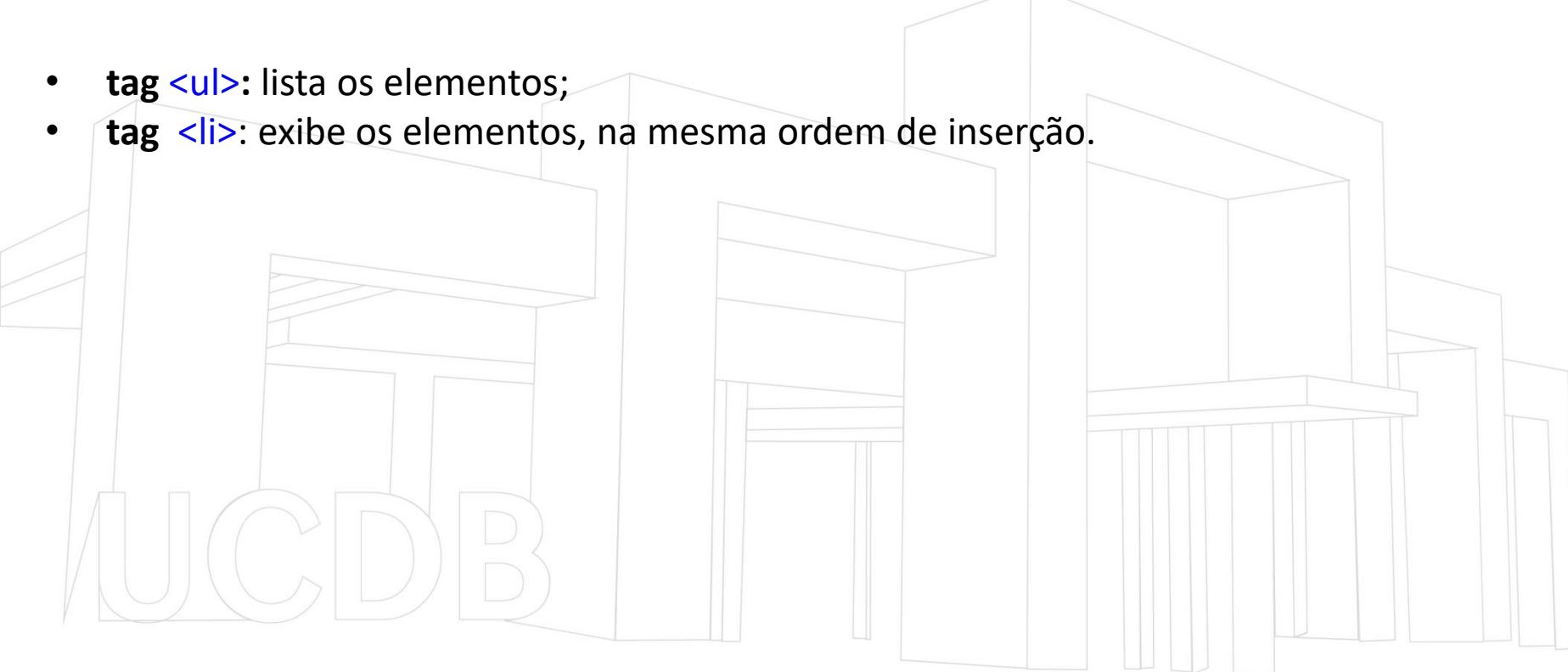
- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;



Linguagem HTML

LISTA NÃO ORDENADA

- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;
- **tag **: exibe os elementos, na mesma ordem de inserção.



Linguagem HTML

LISTA NÃO ORDENADA

- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;
- **tag **: exibe os elementos, na mesma ordem de inserção.

EXEMPLO:

UCDB

Linguagem HTML

LISTA NÃO ORDENADA

- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;
- **tag **: exibe os elementos, na mesma ordem de inserção.

EXEMPLO:

```
<ul>
  <li>PHP</li>
  <li>Java</li>
  <li>Python</li>
  <li>Ruby</li>
</ul>
```

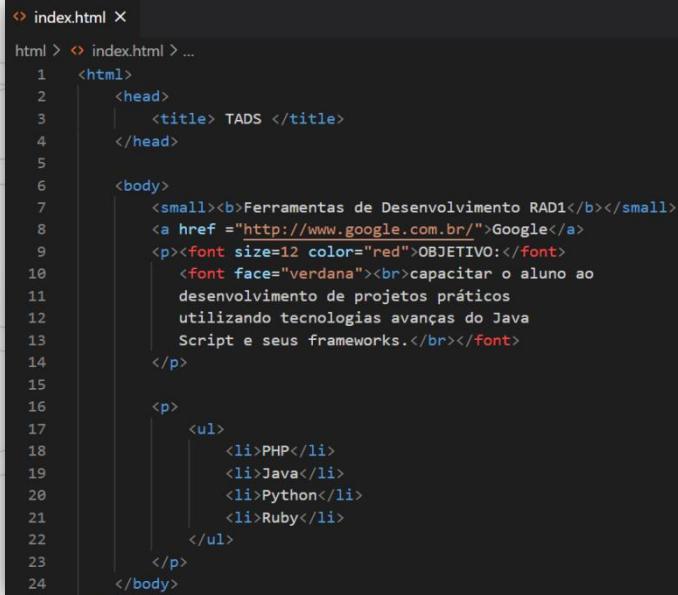
Linguagem HTML

LISTA NÃO ORDENADA

- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;
- **tag **: exibe os elementos, na mesma ordem de inserção.

EXEMPLO:

```
<ul>
  <li>PHP</li>
  <li>Java</li>
  <li>Python</li>
  <li>Ruby</li>
</ul>
```



```
index.html ×
html > index.html > ...
1  <html>
2    <head>
3      <title> TADS </title>
4    </head>
5
6    <body>
7      <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8      <a href ="http://www.google.com.br/">Google</a>
9      <p><font size=12 color="red">OBJETIVO:</font>
10        <font face="verdana"><br>capacitar o aluno ao
11          desenvolvimento de projetos práticos
12          utilizando tecnologias avançadas do Java
13          Script e seus frameworks.<br></font>
14      </p>
15
16      <p>
17        <ul>
18          <li>PHP</li>
19          <li>Java</li>
20          <li>Python</li>
21          <li>Ruby</li>
22        </ul>
23      </p>
24    </body>
```

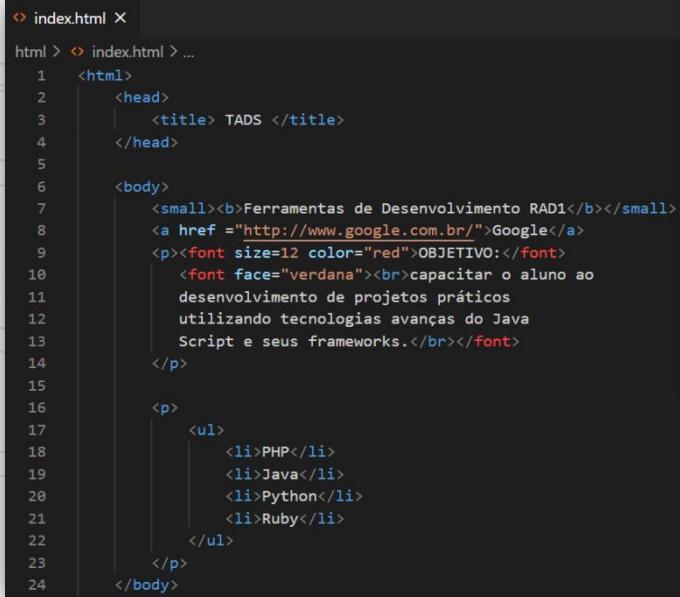
Linguagem HTML

LISTA NÃO ORDENADA

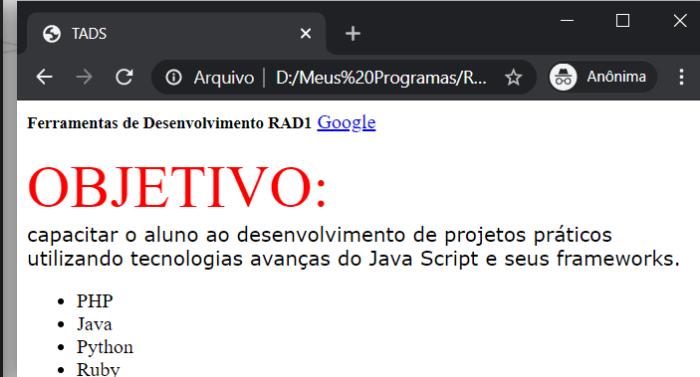
- Similar à anterior, só mesmo diferentes pela **tag** que abre e fecha a lista.
- **tag **: lista os elementos;
- **tag **: exibe os elementos, na mesma ordem de inserção.

EXEMPLO:

```
<ul>
  <li>PHP</li>
  <li>Java</li>
  <li>Python</li>
  <li>Ruby</li>
</ul>
```



```
index.html x
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <a href ="http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11      desenvolvimento de projetos práticos
12      utilizando tecnologias avançadas do Java
13      Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      <ul>
18        <li>PHP</li>
19        <li>Java</li>
20        <li>Python</li>
21        <li>Ruby</li>
22      </ul>
23    </p>
24  </body>
```



OBJETIVO:

capacitar o aluno ao desenvolvimento de projetos práticos
utilizando tecnologias avançadas do Java Script e seus frameworks.

- PHP
- Java
- Python
- Ruby

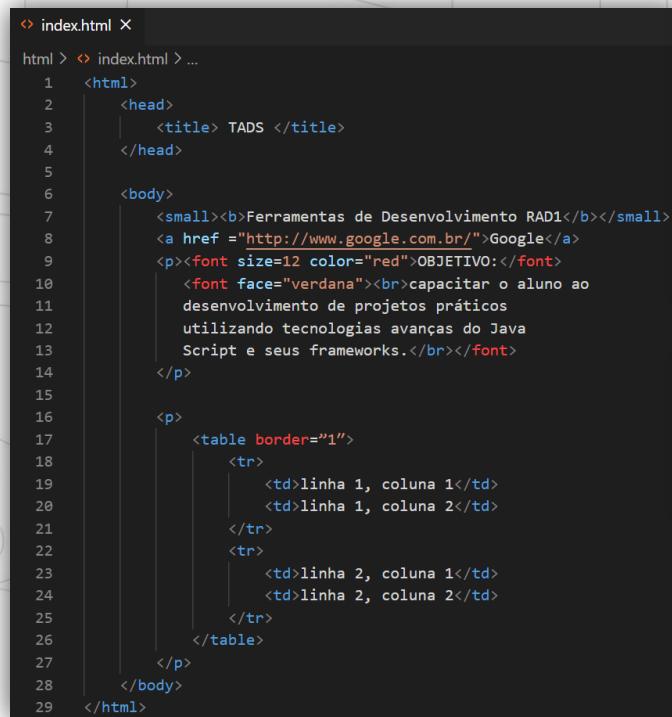
Linguagem HTML

TABELA

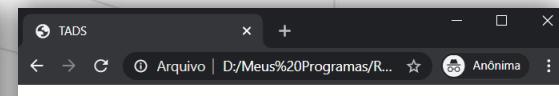
- Para inserir uma tabela em um arquivo HTML, utilizamos a tag `<table>`;
- As tabelas são divididas em **linhas**, indicada com a tag `<tr>`;
- As linhas dividem-se em **células**, indicadas pela tag `<td>`.

EXEMPLO:

```
<table border="1">
<tr>
    <td>linha 1, coluna 1</td>
    <td>linha 1, coluna 2</td>
</tr>
<tr>
    <td>linha 2, coluna 1</td>
    <td>linha 2, coluna 2</td>
</tr>
</table>
```



```
index.html x
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <body>
7     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8     <a href ="http://www.google.com.br/">Google</a>
9     <p><font size=12 color="red">OBJETIVO:</font>
10    <font face="verdana"><br>capacitar o aluno ao
11        desenvolvimento de projetos práticos
12        utilizando tecnologias avançadas do Java
13        Script e seus frameworks.<br></font>
14    </p>
15
16    <p>
17      <table border="1">
18        <tr>
19          <td>linha 1, coluna 1</td>
20          <td>linha 1, coluna 2</td>
21        </tr>
22        <tr>
23          <td>linha 2, coluna 1</td>
24          <td>linha 2, coluna 2</td>
25        </tr>
26      </table>
27    </p>
28  </body>
29 </html>
```



OBJETIVO:

capacitar o aluno ao desenvolvimento de projetos práticos utilizando tecnologias avançadas do Java Script e seus frameworks.

linha 1, coluna 1	linha 1, coluna 2
linha 2, coluna 1	linha 2, coluna 2

linha 1, coluna 1	linha 1, coluna 2
linha 2, coluna 1	linha 2, coluna 2

Linguagem HTML

COMENTÁRIOS



Linguagem HTML

COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;



Linguagem HTML

COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;
- **EXEMPLO:**



Linguagem HTML

COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;
- EXEMPLO:

<!

>

UCDB

Linguagem HTML

COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;
- EXEMPLO:

```
<!-- TAP1/RAD1 -->
```



Linguagem HTML

COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;
- EXEMPLO:

```
<!-- TAP1/RAD1 -->
```

```
index.html X

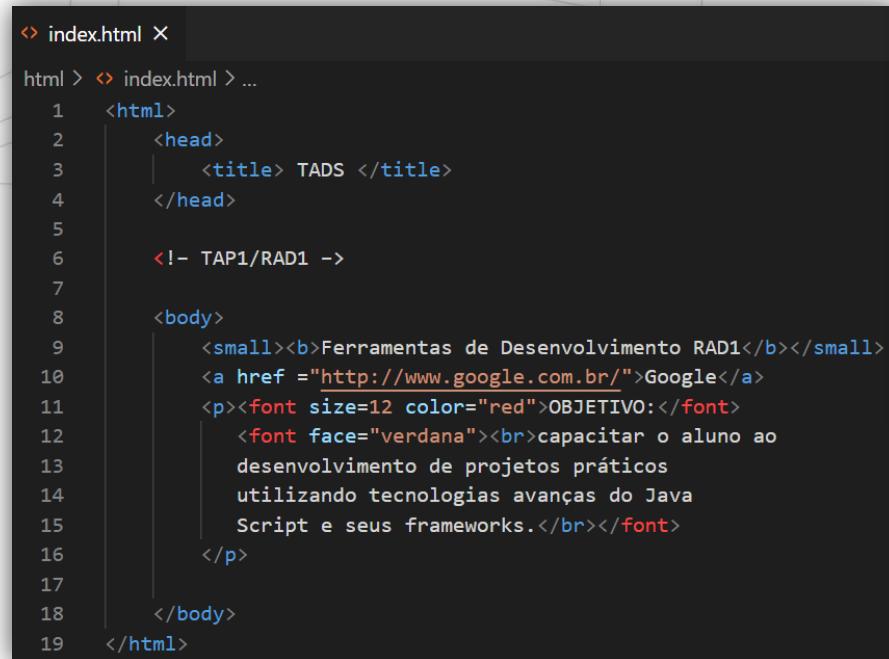
html > index.html > ...
1   <html>
2     <head>
3       <title> TADS </title>
4     </head>
5
6   <!-- TAP1/RAD1 -->
7
8   <body>
9     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
10    <a href ="http://www.google.com.br/">Google</a>
11    <p><font size=12 color="red">OBJETIVO:</font>
12      <font face="verdana"><br>capacitar o aluno ao
13        desenvolvimento de projetos práticos
14        utilizando tecnologias avanças do Java
15        Script e seus frameworks.<br></font>
16    </p>
17
18   </body>
19 </html>
```

Linguagem HTML

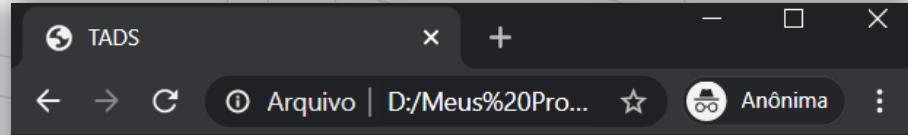
COMENTÁRIOS

- Comentários podem ser inseridos no código-fonte, e só serão acessíveis pelos desenvolvedores que tiverem acesso à edição;
- EXEMPLO:

```
<!-- TAP1/RAD1 -->
```



```
index.html X
html > index.html > ...
1  <html>
2   <head>
3     <title> TADS </title>
4   </head>
5
6   <!-- TAP1/RAD1 -->
7
8   <body>
9     <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
10    <a href ="http://www.google.com.br/">Google</a>
11    <p><font size=12 color="red">OBJETIVO:</font>
12      <font face="verdana"><br>capacitar o aluno ao
13      desenvolvimento de projetos práticos
14      utilizando tecnologias avanças do Java
15      Script e seus frameworks.<br></font>
16    </p>
17
18   </body>
19 </html>
```



OBJETIVO:

capacitar o aluno ao desenvolvimento de projetos práticos utilizando tecnologias avanças do Java Script e seus frameworks.

Linguagem HTML

FORMULÁRIOS



Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;



Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form>
```

```
</form>
```

UCDB

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name
```

```
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input"
```

```
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action  
      </form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp">  
  ...  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text"  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text" name  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text" name="user">  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text" name="user">  
    <input  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text" name="user">  
    <input type  
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">
    Nome do Usuário:
    <input type="text" name="user">
    <input type="submit" >
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">
    Nome do Usuário:
    <input type="text" name="user">
    <input type="submit" value      >
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

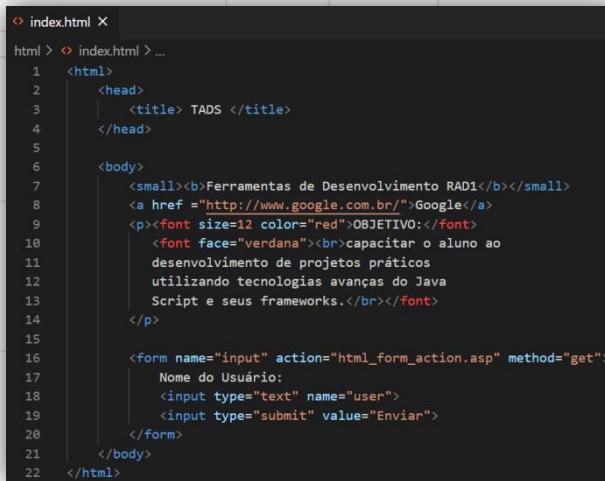
```
<form name="input" action="html_form_action.asp" method="get">
    Nome do Usuário:
    <input type="text" name="user">
    <input type="submit" value="Enviar">
</form>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">
    Nome do Usuário:
    <input type="text" name="user">
    <input type="submit" value="Enviar">
</form>
```



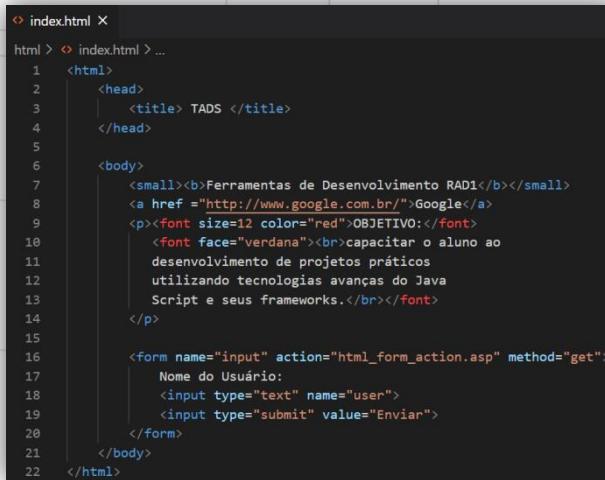
```
index.html x
html > index.html ...
1  <html>
2      <head>
3          <title> TADS </title>
4      </head>
5
6      <body>
7          <small><b>Ferramentas de Desenvolvimento RAD1</b></small>
8          <a href ="http://www.google.com.br/">Google</a>
9          <p><font size=12 color="red">OBJETIVO:</font>
10         <font face="verdana"><br>capacitar o aluno ao
11         desenvolvimento de projetos práticos
12         utilizando tecnologias avançadas.<br></font>
13     </p>
14
15     <form name="input" action="html_form_action.asp" method="get">
16         Nome do Usuário:
17         <input type="text" name="user">
18         <input type="submit" value="Enviar">
19     </form>
20
21 </body>
22 </html>
```

Linguagem HTML

FORMULÁRIOS

- Existem diversos tipos de formulários, que podem coletar diversas informações do usuário;
- Para inserir um formulário simples em uma página, utilizamos a tag `<form>`.
- **EXEMPLO:**

```
<form name="input" action="html_form_action.asp" method="get">  
    Nome do Usuário:  
    <input type="text" name="user">  
    <input type="submit" value="Enviar">  
</form>
```



```
index.html x  
html > index.html ...  
1  <html>  
2      <head>  
3          <title> TADS </title>  
4      </head>  
5  
6      <body>  
7          <small><b>Ferramentas de Desenvolvimento RAD1</b></small>  
8          <a href ="http://www.google.com.br/">Google</a>  
9          <p><font size=12 color="red">OBJETIVO:</font>  
10         <font face="verdana"><br>capacitar o aluno ao  
11             desenvolvimento de projetos práticos  
12             utilizando tecnologias avançadas do Java  
13             Script e seus frameworks.<br></font>  
14         </p>  
15  
16         <form name="input" action="html_form_action.asp" method="get">  
17             Nome do Usuário:  
18             <input type="text" name="user">  
19             <input type="submit" value="Enviar">  
20         </form>  
21     </body>  
22 </html>
```





3.2. Introdução ao CSS

Prof. Me. Marcos Alves
marcos@ucdb.br

CSS

CSS (Folhas de Estilo em Cascata)



CSS

CSS (Folhas de Estilo em Cascata)

- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



CSS

CSS (Folhas de Estilo em Cascata)

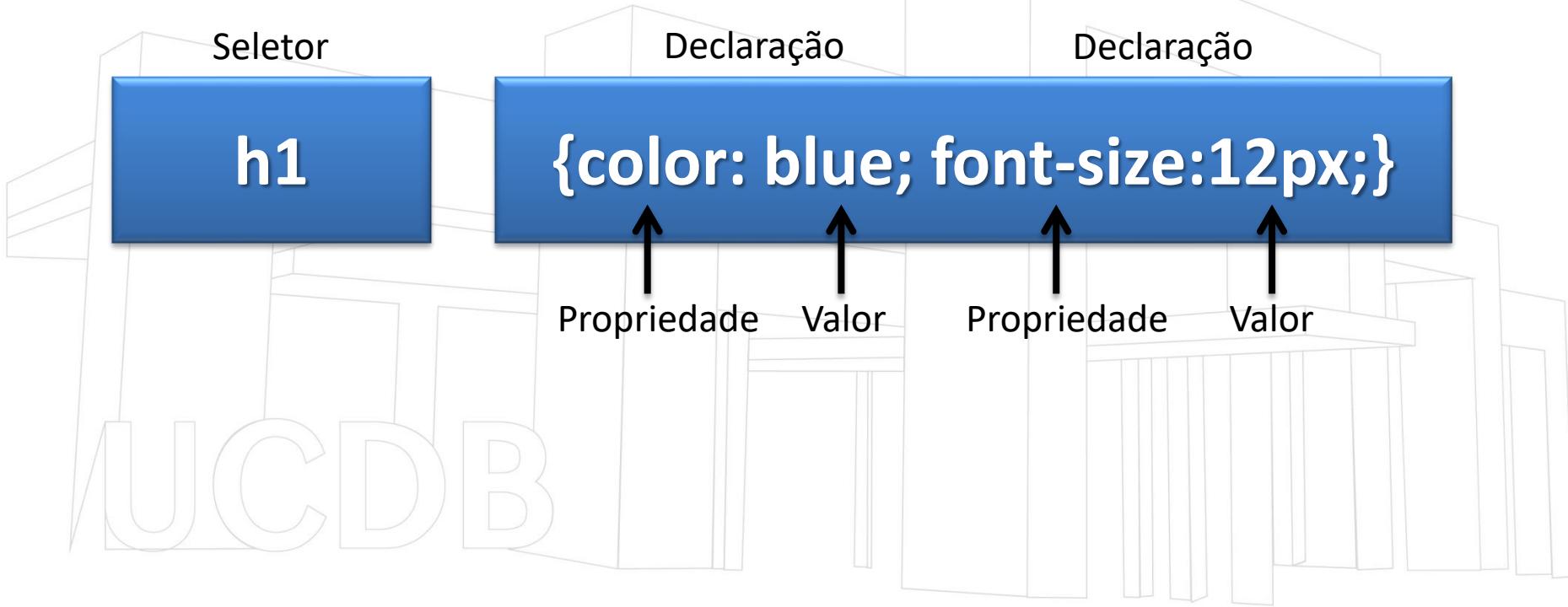
- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



CSS

CSS (Folhas de Estilo em Cascata)

- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



CSS

CSS (Folhas de Estilo em Cascata)

- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



CSS

CSS (Folhas de Estilo em Cascata)

- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



- A folha de estilos pode ser um arquivo de textos simples (alfabeto ISO-Latin1) com a extensão **.css**.
- Para vinculá-lo a uma página HTML, esta deve ter dentro do seu bloco **<head>** ... **</head>** o seguinte descritor:

CSS

CSS (Folhas de Estilo em Cascata)

- Conjunto de regras repassadas ao programa responsável pela formatação de um documento (organização da página, posicionamento de texto, ...);



<LINK REL="stylesheet" HREF="url_do_arquivo.css">

CSS

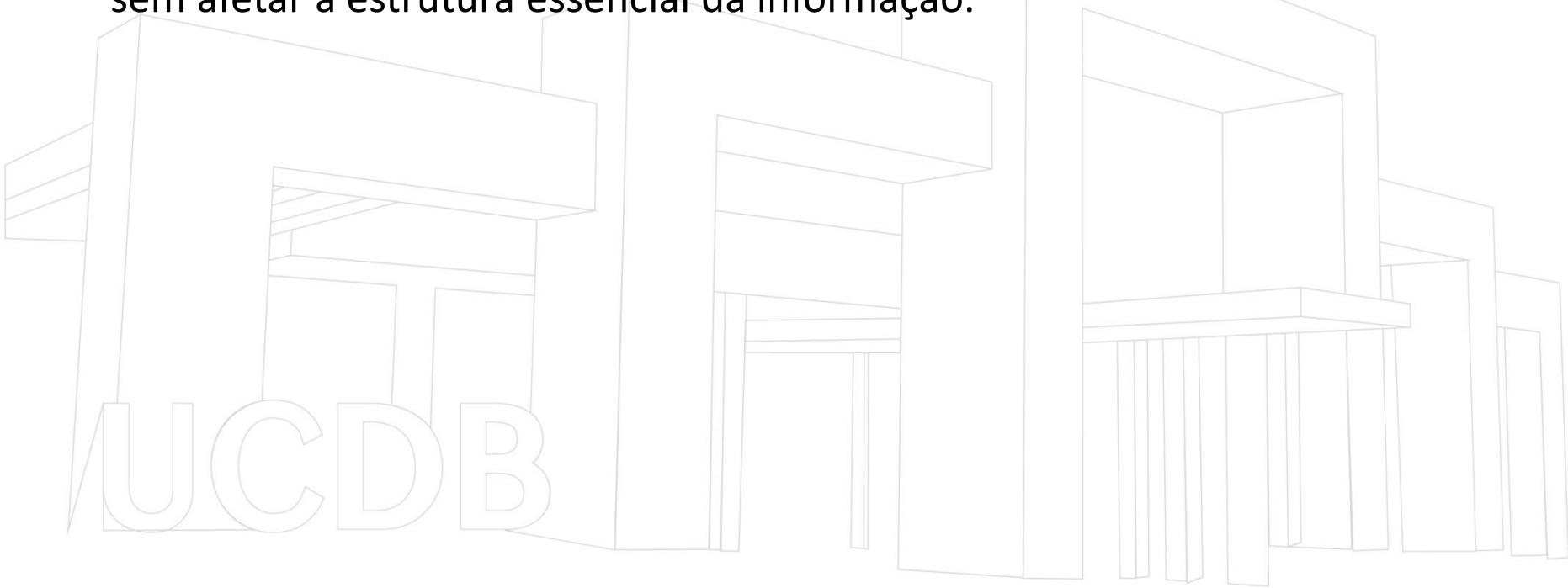
Separação entre Apresentação e Estrutura



CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.



CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.

Controle absoluto da aparência da página

CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.

Controle absoluto da aparência da página

- Grande dificuldade do HTML em, por exemplo, fluir um texto ao redor de uma imagem;

CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.

Controle absoluto da aparência da página

- Grande dificuldade do HTML em, por exemplo, fluir um texto ao redor de uma imagem;
- Já o CSS pode posicionar uma imagem em qualquer lugar, de forma relativa ou absoluta;

CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.

Controle absoluto da aparência da página

- Grande dificuldade do HTML em, por exemplo, fluir um texto ao redor de uma imagem;
- Já o CSS pode posicionar uma imagem em qualquer lugar, de forma relativa ou absoluta;
- Pode-se escolher a posição exata da imagem de back-ground e fazê-la combinar com algo no foreground;

CSS

Separação entre Apresentação e Estrutura

- Com a informação toda armazenada no HTML (estrutura), a apresentação (estilo) seria uma camada a mais, alterando a disposição do texto, cores, etc - sem afetar a estrutura essencial da informação.

Controle absoluto da aparência da página

- Grande dificuldade do HTML em, por exemplo, fluir um texto ao redor de uma imagem;
- Já o CSS pode posicionar uma imagem em qualquer lugar, de forma relativa ou absoluta;
- Pode-se escolher a posição exata da imagem de back-ground e fazê-la combinar com algo no foreground;
- As dimensões e posições são exatas e dadas em unidades conhecidas no mundo da tipografia como pixels, pontos, milímetros.

CSS

Regras, declarações e seletores



CSS

Regras, declarações e seletores

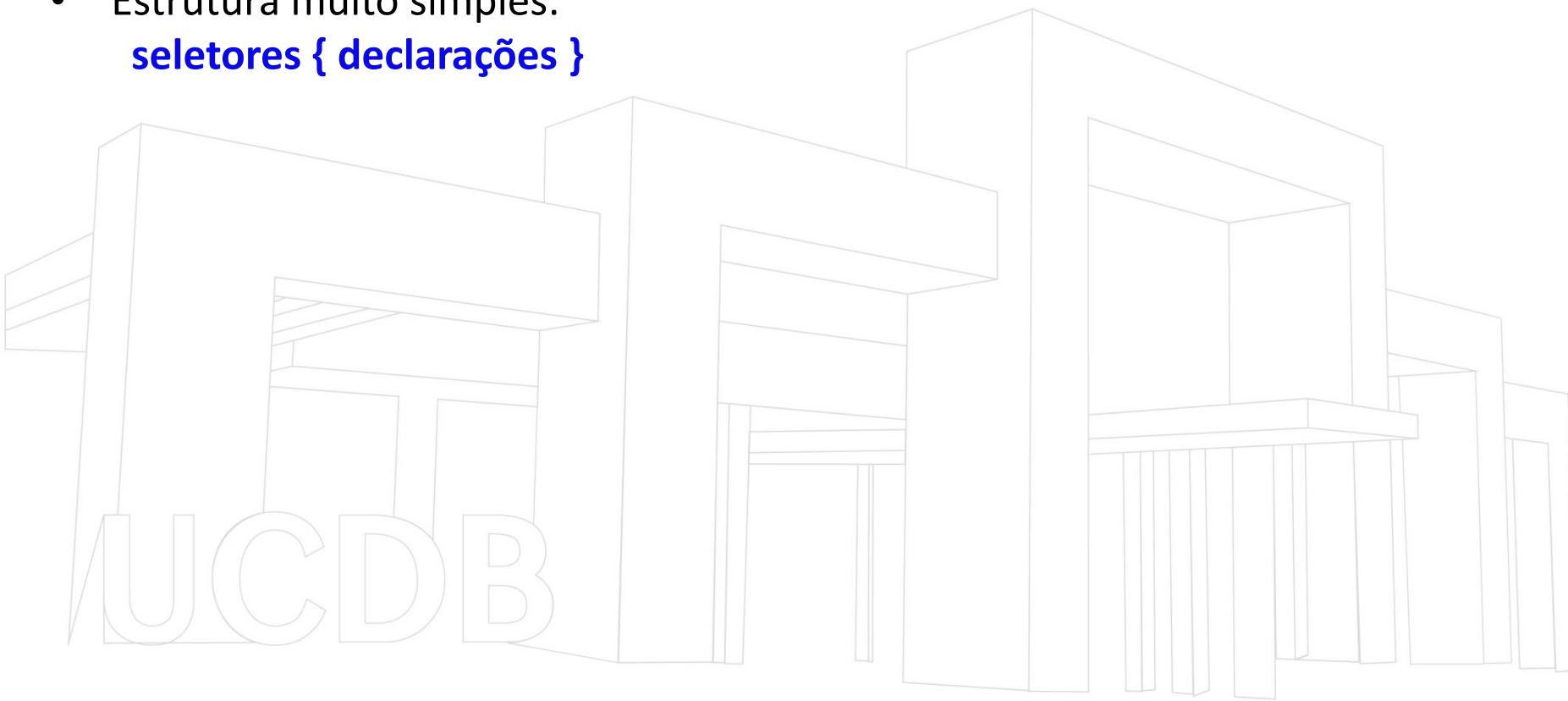
- Estrutura muito simples:



CSS

Regras, declarações e seletores

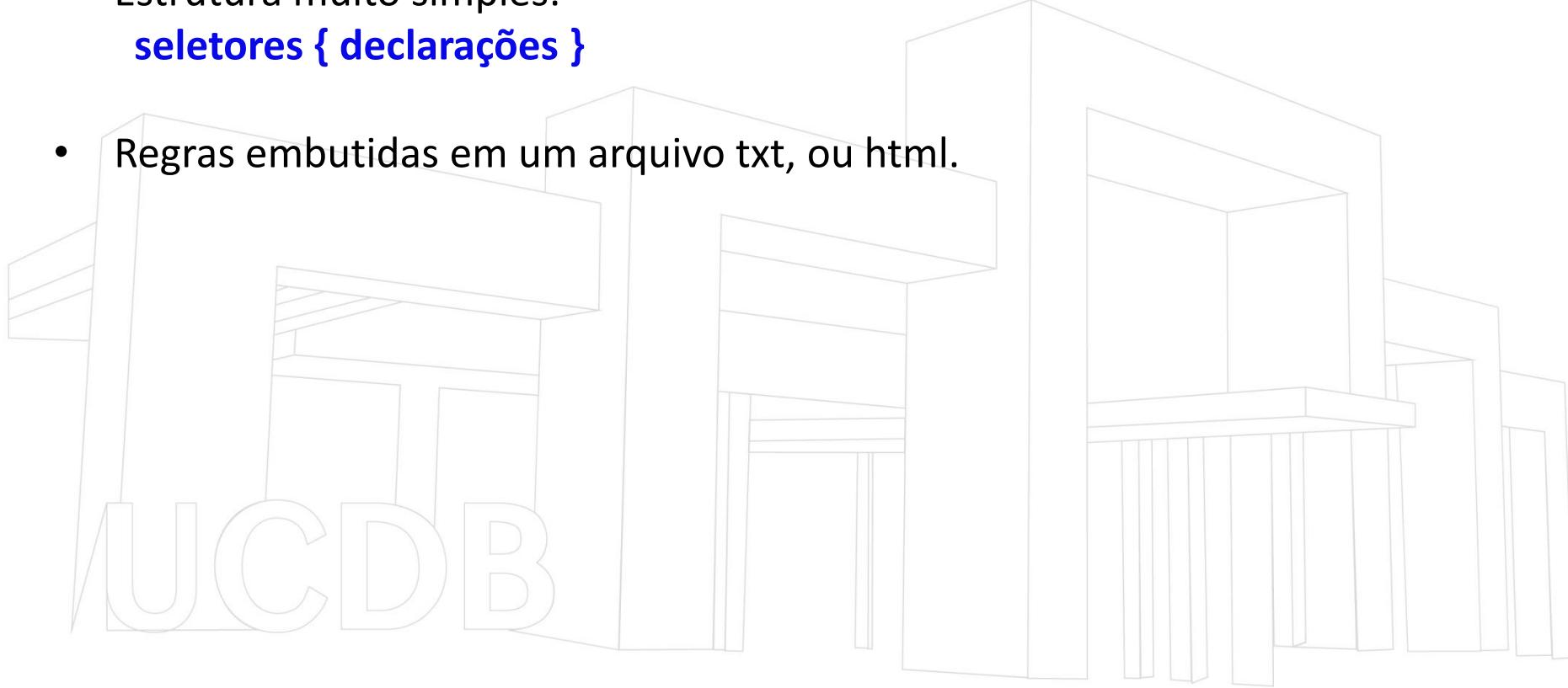
- Estrutura muito simples:
seletores { declarações }



CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

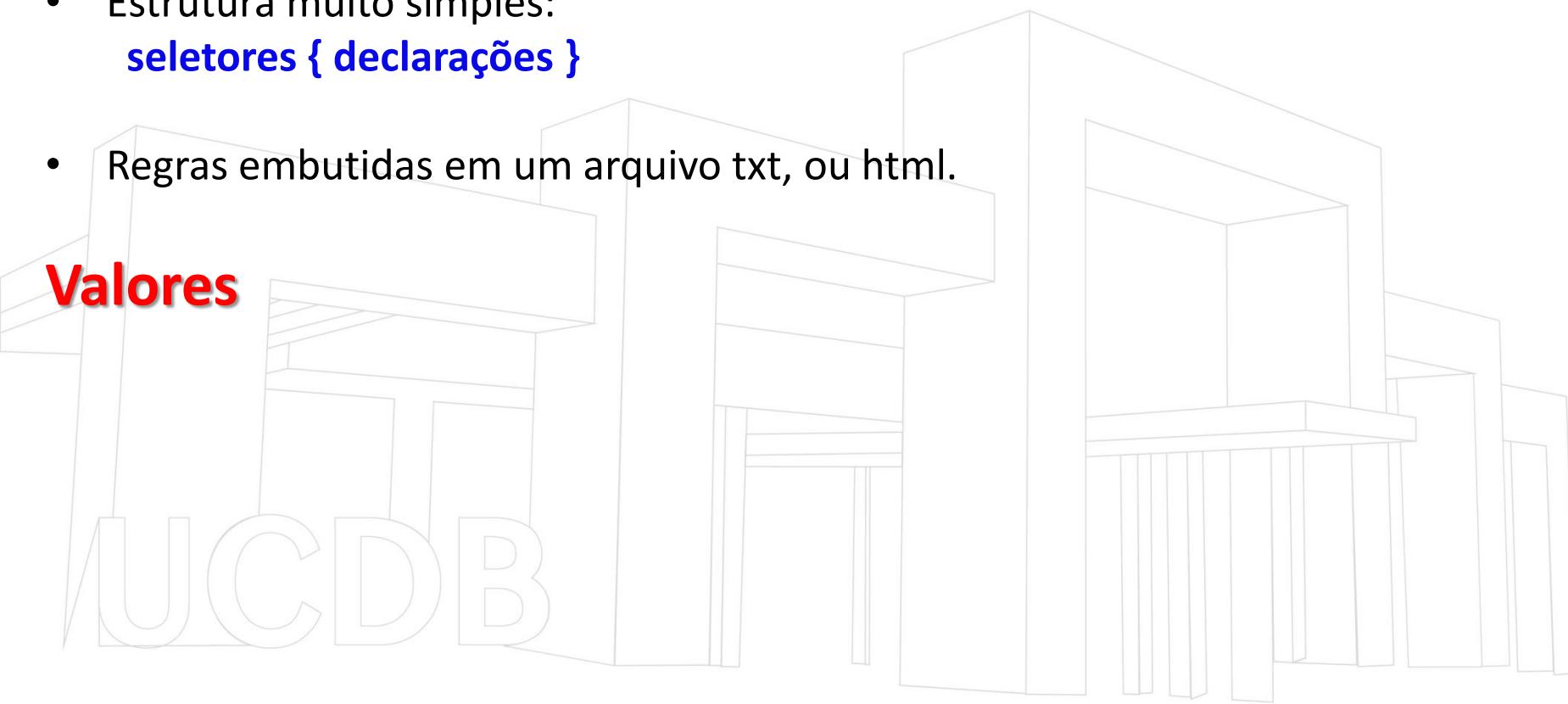


CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores



CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores

- Os valores aplicados às propriedades têm uma sintaxe que depende da propriedade que os usa;

UCDB

CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores

- Os valores aplicados às propriedades têm uma sintaxe que depende da propriedade que os usa;
- Por exemplo, cores → Há quatro maneiras diferentes de especificar cores :

CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores

- Os valores aplicados às propriedades têm uma sintaxe que depende da propriedade que os usa;
- Por exemplo, cores → Há quatro maneiras diferentes de especificar cores :
 - a) Nome do sistema (red, yellow, blue, black, lightGray);

CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores

- Os valores aplicados às propriedades têm uma sintaxe que depende da propriedade que os usa;
- Por exemplo, cores → Há quatro maneiras diferentes de especificar cores :
 - a) Nome do sistema (red, yellow, blue, black, lightGray);
 - b) Código RGB hexadecimal (ff0000, ffff00, 0000ff, 34adfc, 80a7a7) ;

CSS

Regras, declarações e seletores

- Estrutura muito simples:
seletores { declarações }
- Regras embutidas em um arquivo txt, ou html.

Valores

- Os valores aplicados às propriedades têm uma sintaxe que depende da propriedade que os usa;
- Por exemplo, cores → Há quatro maneiras diferentes de especificar cores :
 - a) Nome do sistema (red, yellow, blue, black, lightGray);
 - b) Código RGB hexadecimal (ff0000, ffff00, 0000ff, 34adfc, 80a7a7) ;
 - c) Função rgb() → três argumentos que representam a intensidade dos componentes vermelho (R), verde (G) e azul (B) de uma cor (de 0 a 255).

CSS

Descritores HTML especiais



CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);



CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;

CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;
- Eles são `<div>` e ``;

CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;
- Eles são `<div>` e ``;
- ``:

CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;
- Eles são `<div>` e ``;
- ``:
 - a) Descritor usado dentro de blocos de texto apenas;

CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;
- Eles são `<div>` e ``;
- ``:
 - a) Descritor usado dentro de blocos de texto apenas;
 - b) Chamado descritor em-linha (*inline*) → não quebra a linha antes ou depois;

CSS

Descritores HTML especiais

- São dois descritores estruturais puros (que não definem apresentação específica na folha de estilos nativa do browser);
- Com CSS é possível definir propriedades de apresentação para esses descritores;
- Eles são **<div>** e ****;
- ****:
 - a) Descritor usado dentro de blocos de texto apenas;
 - b) Chamado descritor em-linha (*inline*) → não quebra a linha antes ou depois;
 - c) Se assemelha a descritores como ****, **<i>**, **<small>**, **<a href>** e **<sup>** que servem para formatar texto dentro de parágrafos, células de tabela, etc.

CSS

Descritores HTML especiais

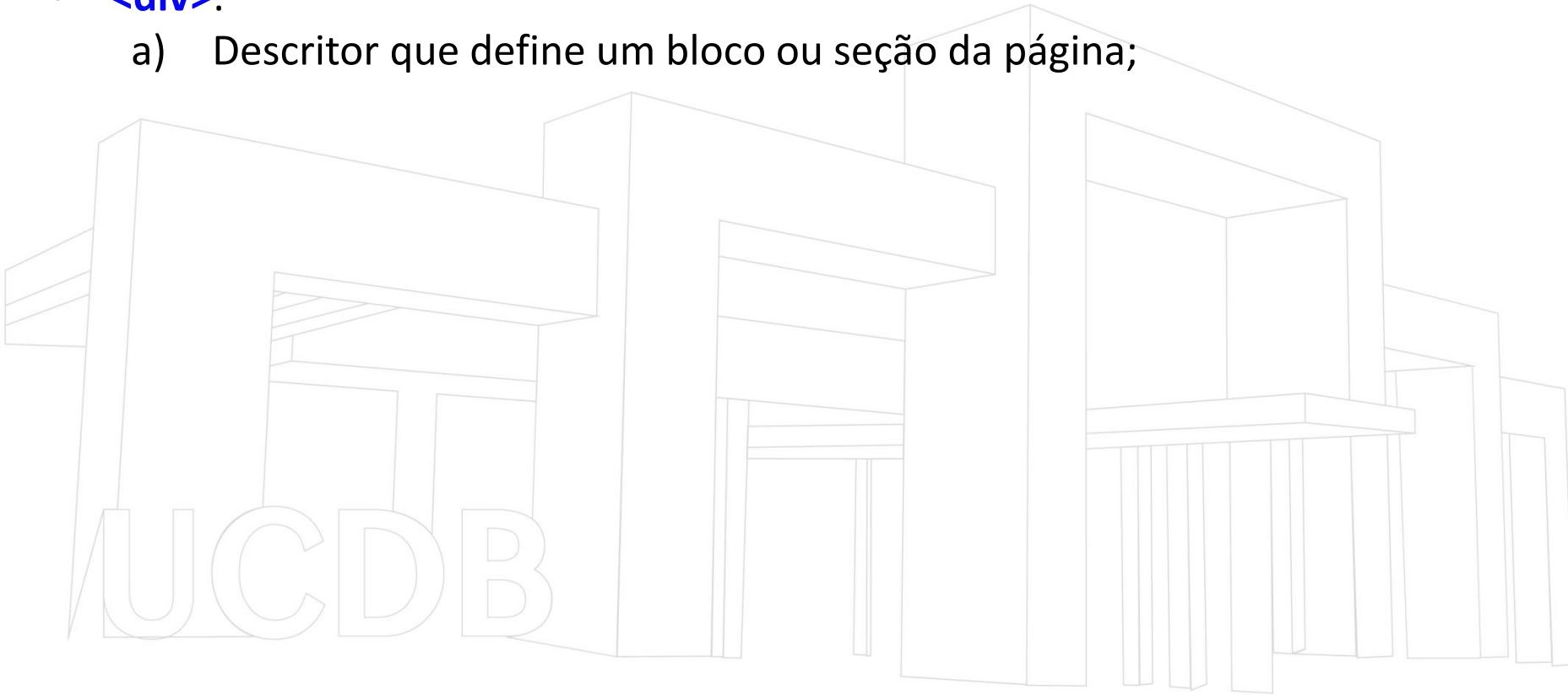
- <div>:



CSS

Descritores HTML especiais

- **<div>**:
 - a) Descritor que define um bloco ou seção da página;



CSS

Descritores HTML especiais

- **<div>**:
 - a) Descritor que define um bloco ou seção da página;
 - b) Pode ser usado para dividir a página em seções e permitir que sejam aplicados estilos específicos a essas seções;

CSS

Descritores HTML especiais

- **<div>**:
 - a) Descritor que define um bloco ou seção da página;
 - b) Pode ser usado para dividir a página em seções e permitir que sejam aplicados estilos específicos a essas seções;
 - c) Descritores de bloco são **<p>**, **<h1>**, **<table>**, etc;

CSS

Descritores HTML especiais

- **<div>**:
 - a) Descritor que define um bloco ou seção da página;
 - b) Pode ser usado para dividir a página em seções e permitir que sejam aplicados estilos específicos a essas seções;
 - c) Descritores de bloco são **<p>**, **<h1>**, **<table>**, etc;
 - d) O **<div>** define um bloco sem função.

CSS

Como Incluir estilos em uma página?



CSS

Como Incluir estilos em uma página?

- Há três formas de aplicar uma folha de estilos a uma página

Web;



CSS

Como Incluir estilos em uma página?

- Há três formas de aplicar uma folha de estilos a uma página Web;
- Estas formas irão determinar a abrangência dos estilos:

CSS

Como Incluir estilos em uma página?

- Há três formas de aplicar uma folha de estilos a uma página

Web;

- Estas formas irão determinar a abrangência dos estilos:
 - a) se afetarão um trecho limitado de uma página;

CSS

Como Incluir estilos em uma página?

- Há três formas de aplicar uma folha de estilos a uma página Web;
- Estas formas irão determinar a abrangência dos estilos:
 - a) se afetarão um trecho limitado de uma página;
 - b) se a toda a página;

CSS

Como Incluir estilos em uma página?

- Há três formas de aplicar uma folha de estilos a uma página Web;
- Estas formas irão determinar a abrangência dos estilos:
 - a) se afetarão um trecho limitado de uma página;
 - b) se a toda a página;
 - c) ou se irão afetar todo um site.

CSS

Como Incluir estilos em uma página?

- PRIMEIRA FORMA



CSS

Como Incluir estilos em uma página?

- **PRIMEIRA FORMA** → Depois que as definições estiverem prontas e gravadas em um arquivo com extensão .css, o estilo estará pronto para ser usado;



CSS

Como Incluir estilos em uma página?

- **PRIMEIRA FORMA** → Depois que as definições estiverem prontas e gravadas em um arquivo com extensão .css, o estilo estará pronto para ser usado;
- Pode ser importado através do descritor LINK:



CSS

Como Incluir estilos em uma página?

- **PRIMEIRA FORMA** → Depois que as definições estiverem prontas e gravadas em um arquivo com extensão .css, o estilo estará pronto para ser usado;
- Pode ser importado através do descritor LINK:

```
<head> (...)  
  <link rel=stylesheet  
        href="http://internet-name/mystyles.css"  
        type="text/css"  
  >  
</head>
```

CSS

Como Incluir estilos em uma página?

- **PRIMEIRA FORMA** → Depois que as definições estiverem prontas e gravadas em um arquivo com extensão .css, o estilo estará pronto para ser usado;
- Pode ser importado através do descritor LINK:

```
<head> (...)  
  <link rel=stylesheet  
        href="http://internet-name/mystyles.css"  
        type="text/css"  
  >  
</head>
```

- O `<link>` não tem descritor de fechamento e deve ser usado dentro do bloco `<head>`.

CSS

Como Incluir estilos em uma página?

- SEGUNDA FORMA



CSS

Como Incluir estilos em uma página?

- **SEGUNDA FORMA** → estilos sejam aplicados localmente, em uma única página, embutindo uma folha de estilos diretamente na página HTML dentro de um bloco formado pelos descritores `<style>` e `</style>`:



CSS

Como Incluir estilos em uma página?

- **SEGUNDA FORMA** → estilos sejam aplicados localmente, em uma única página, embutindo uma folha de estilos diretamente na página HTML dentro de um bloco formado pelos descritores `<style>` e `</style>`:

```
<style type="text/css">
    p { font: 12pt "Times New Roman" bold; color: red }
    i { color: black }
</style>
```

CSS

Como Incluir estilos em uma página?

- As propriedades declaradas no bloco `<style>` sobrepõem qualquer propriedades anteriores do elemento (inclusive as de uma folha de estilos importada, se houver);



CSS

Como Incluir estilos em uma página?

- As propriedades declaradas no bloco `<style>` sobrepõem qualquer propriedades anteriores do elemento (inclusive as de uma folha de estilos importada, se houver);
- É comum colocar todo o código entre comentários HTML (`<!-- e -->`) para proteger browsers antigos da exibição indesejada do código:

CSS

Como Incluir estilos em uma página?

- As propriedades declaradas no bloco `<style>` sobrepõem qualquer propriedades anteriores do elemento (inclusive as de uma folha de estilos importada, se houver);
- É comum colocar todo o código entre comentários HTML (`<!--` e `-->`) para proteger browsers antigos da exibição indesejada do código:

```
<style type="text/css">
  <!--
    p { font: 12pt "Times New Roman" bold;
        color: red }
    i { color: black }
  -->
</style>
```

CSS

Como Incluir estilos em uma página?

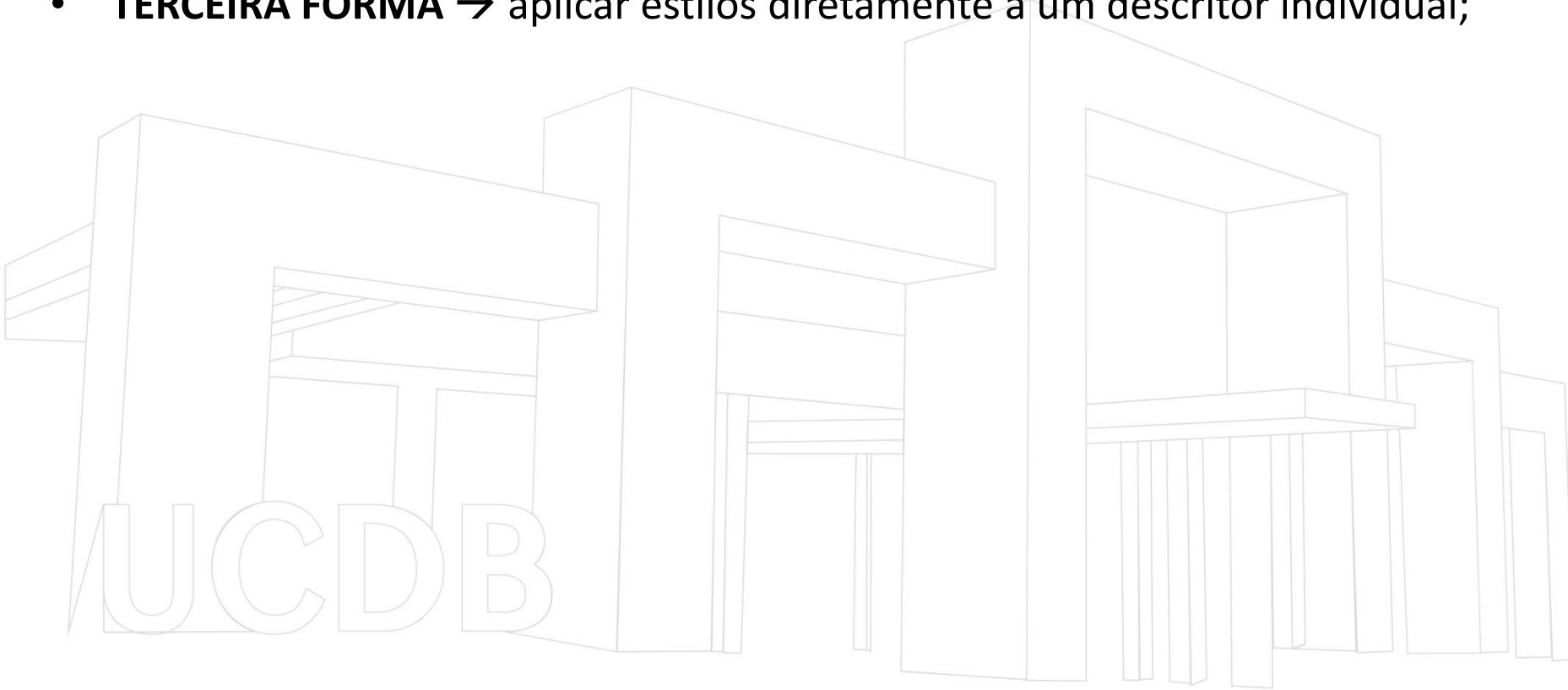
- TERCEIRA FORMA



CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;



CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;
- Para tanto deve-se usar o atributo STYLE em quase qualquer descritor;



CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;
- Para tanto deve-se usar o atributo STYLE em quase qualquer descritor;
- Este método não corresponde exatamente a uma "folha" de estilos, pois os estilos aplicados não são **reaproveitáveis**;

CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;
- Para tanto deve-se usar o atributo STYLE em quase qualquer descritor;
- Este método não corresponde exatamente a uma "folha" de estilos, pois os estilos aplicados não são **reaproveitáveis**;
- Permite alterar a aparência de um único descritor, de um conjunto deles ou de um bloco de informações da página.

CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;
- Para tanto deve-se usar o atributo STYLE em quase qualquer descritor;
- Este método não corresponde exatamente a uma "folha" de estilos, pois os estilos aplicados não são **reaproveitáveis**;
- Permite alterar a aparência de um único descritor, de um conjunto deles ou de um bloco de informações da página.
- Exemplo:

CSS

Como Incluir estilos em uma página?

- **TERCEIRA FORMA** → aplicar estilos diretamente a um descritor individual;
- Para tanto deve-se usar o atributo STYLE em quase qualquer descritor;
- Este método não corresponde exatamente a uma "folha" de estilos, pois os estilos aplicados não são **reaproveitáveis**;
- Permite alterar a aparência de um único descritor, de um conjunto deles ou de um bloco de informações da página.
- Exemplo:

```
<p style="color: green; font-size: 12pt"> Este texto </p>
```

CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:
 - No arquivo estilos.css:



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:
 - No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
color: red }
```



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:
 - No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:

- No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```

- No começo do arquivo HTML:



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:

- No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```

- No começo do arquivo HTML:

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

CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:

- No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```

- No começo do arquivo HTML:

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

- Mais à frente no arquivo HTML:



CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:

- No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```

- No começo do arquivo HTML:

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

- Mais à frente no arquivo HTML:

```
<h1 style="color: navy">Auto da Comadecida</h1>
```

CSS

Como Incluir estilos em uma página?

- É possível utilizar um ou mais métodos ao mesmo tempo;
- Exemplo:

- No arquivo estilos.css:

```
p {font: 12pt "Times New Roman" bold;  
    color: red }  
h1 {color: black }
```

- No começo do arquivo HTML:

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

- Mais à frente no arquivo HTML:

```
<h1 style="color: navy">Auto da Compadecida</h1>  
<p style="color: black">Ariano Suassuna (Recife, 1955)</p>
```

CSS

Como Incluir estilos em uma página?

- **QUAL O RESULTADO?**



CSS

Como Incluir estilos em uma página?

- **QUAL O RESULTADO?**



CSS

Classes e IDs



CSS

Classes e IDs

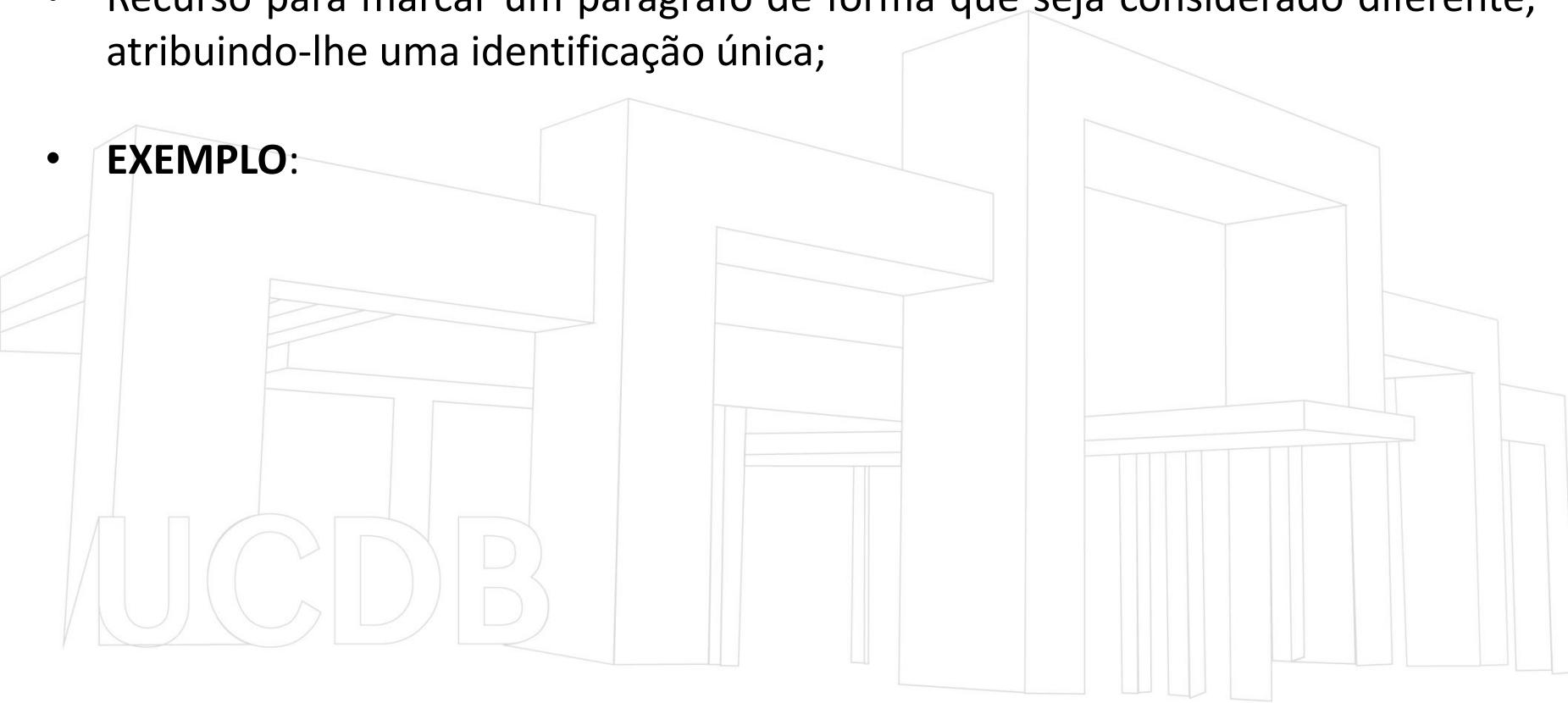
- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;



CSS

Classes e IDs

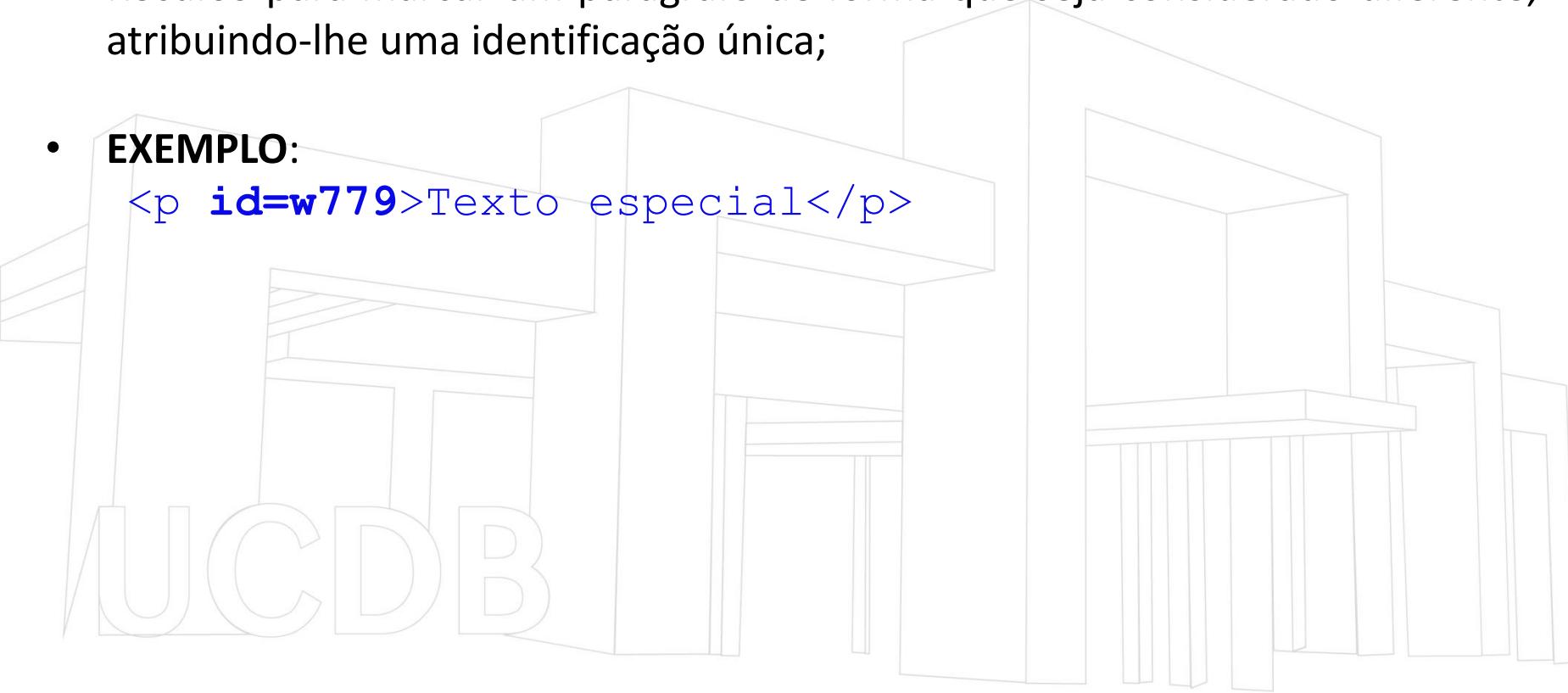
- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- **EXEMPLO:**



CSS

Classes e IDs

- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- EXEMPLO:
`<p id=w779>Texto especial</p>`



CSS

Classes e IDs

- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- **EXEMPLO:**
`<p id=w779>Texto especial</p>`
- Para alterar as características deste parágrafo agora, pode-se usar o seu ID como seletor, da forma:

CSS

Classes e IDs

- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- **EXEMPLO:**
`<p id=w779>Texto especial</p>`
- Para alterar as características deste parágrafo agora, pode-se usar o seu ID como seletor, da forma:
`#w779 {color: cyan }`

CSS

Classes e IDs

- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- EXEMPLO:
`<p id=w779>Texto especial</p>`
- Para alterar as características deste parágrafo agora, pode-se usar o seu ID como seletor, da forma:
`#w779 {color: cyan }`
- Se isto estiver em um arquivo CSS, todas as páginas que o usam e que tiverem um elemento com o ID **#w779** serão afetadas;

CSS

Classes e IDs

- Recurso para marcar um parágrafo de forma que seja considerado diferente, atribuindo-lhe uma identificação única;
- **EXEMPLO:**
`<p id=w779>Texto especial</p>`
- Para alterar as características deste parágrafo agora, pode-se usar o seu ID como seletor, da forma:
`#w779 {color: cyan }`
- Se isto estiver em um arquivo CSS, todas as páginas que o usam e que tiverem um elemento com o ID `#w779` serão afetadas;
- Se houver mais de um com o mesmo ID apenas o primeiro será afetado.

CSS

Classes e IDs

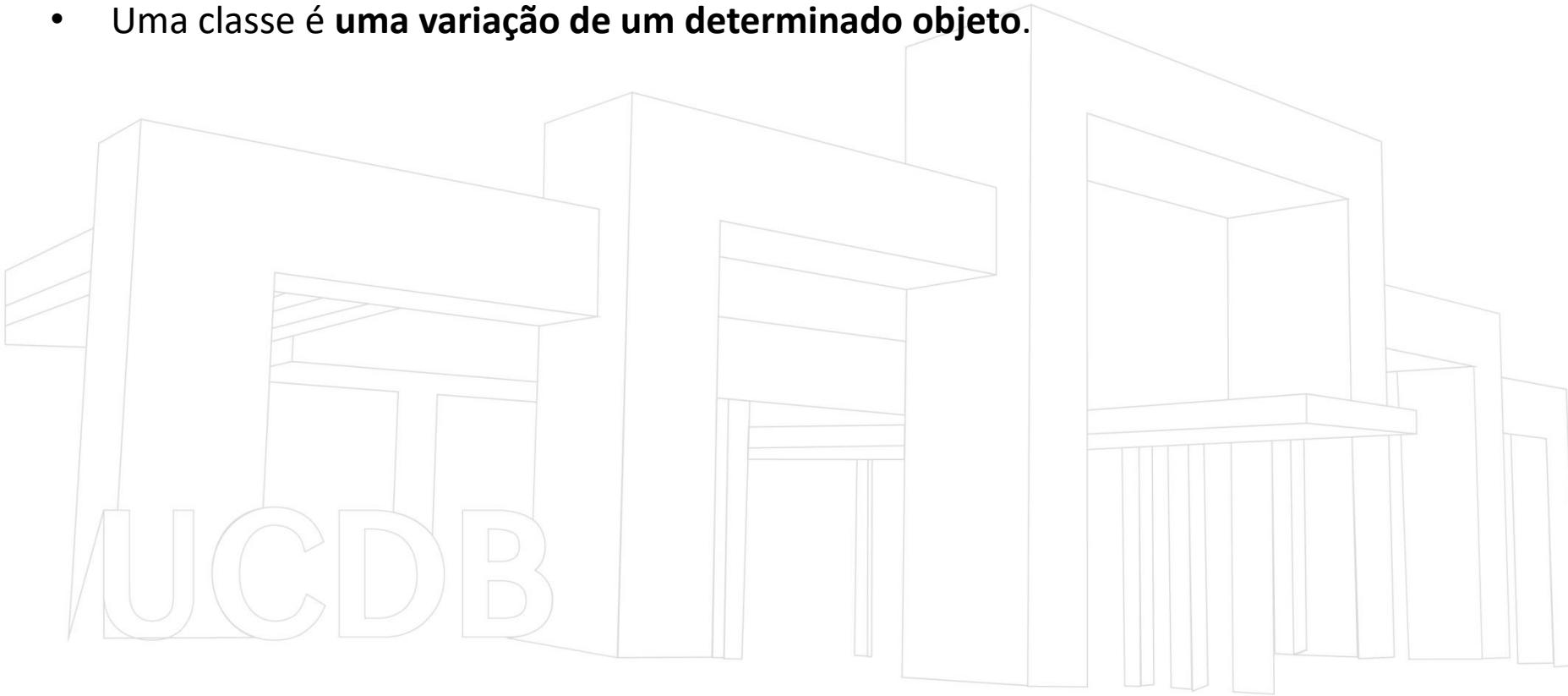
- Melhor que usar ID é agrupar características semelhantes em classes;



CSS

Classes e IDs

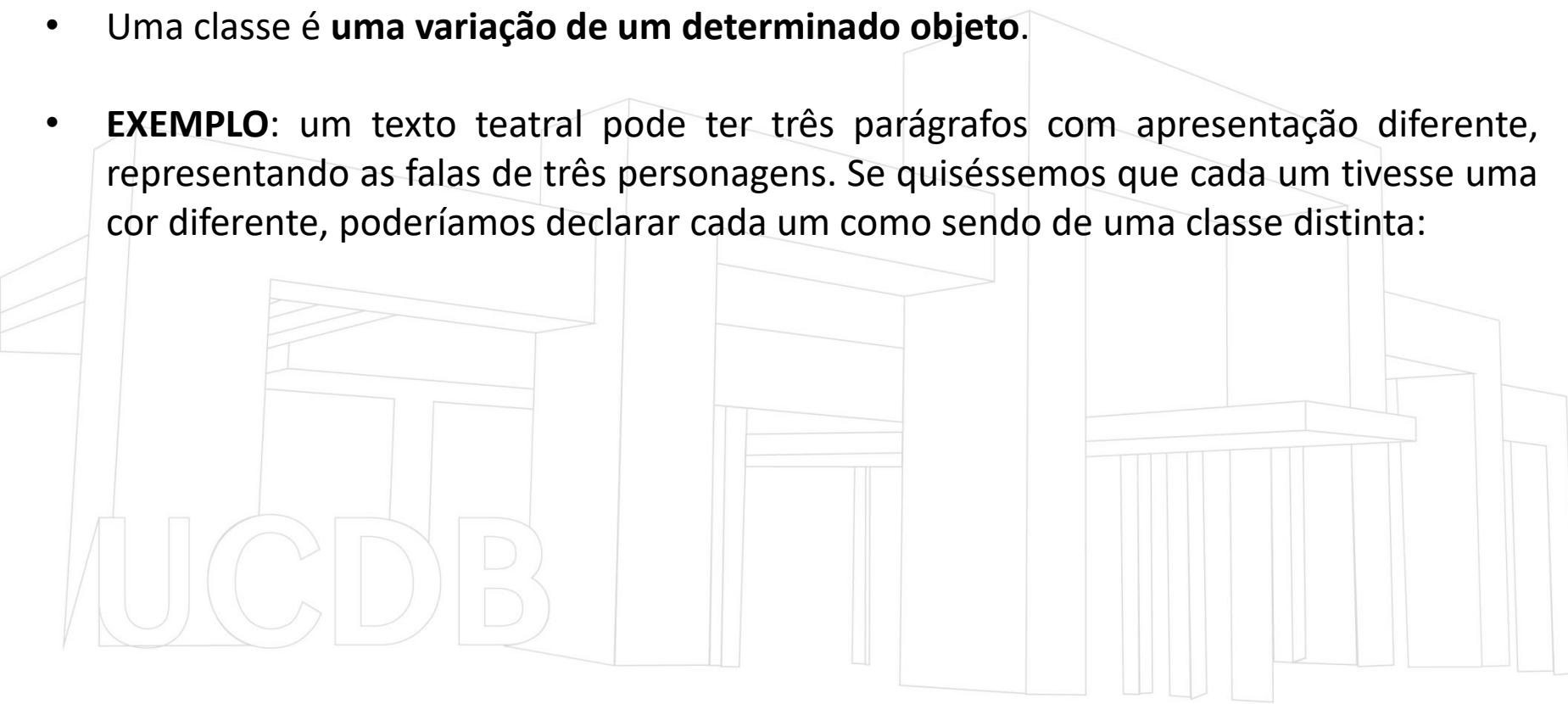
- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**



CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:



CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:
`<p class=padre> Eu retiro o que disse, João </p>`

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
```

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
```

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
<p class=padre> Enterrado latindo, Senhor Bispo, Au, au, au, não
sabe? </p>
```

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
<p class=padre> Enterrado latindo, Senhor Bispo, Au, au, au, não
sabe? </p>
```

- Para dar a cada parágrafo de um mesmo personagem (mesma classe) os mesmos atributos, usa-se:

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
<p class=padre> Enterrado latindo, Senhor Bispo, Au, au, au, não
sabe? </p>
```

- Para dar a cada parágrafo de um mesmo personagem (mesma classe) os mesmos atributos, usa-se:

```
p.grilo { color: maroon }
```

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
<p class=padre> Enterrado latindo, Senhor Bispo, Au, au, au, não
sabe? </p>
```

- Para dar a cada parágrafo de um mesmo personagem (mesma classe) os mesmos atributos, usa-se:

```
p.grilo { color: maroon }
p.padre { color: black }
```

CSS

Classes e IDs

- Melhor que usar ID é agrupar características semelhantes em classes;
- Uma classe é **uma variação de um determinado objeto.**
- **EXEMPLO:** um texto teatral pode ter três parágrafos com apresentação diferente, representando as falas de três personagens. Se quiséssemos que cada um tivesse uma cor diferente, poderíamos declarar cada um como sendo de uma classe distinta:

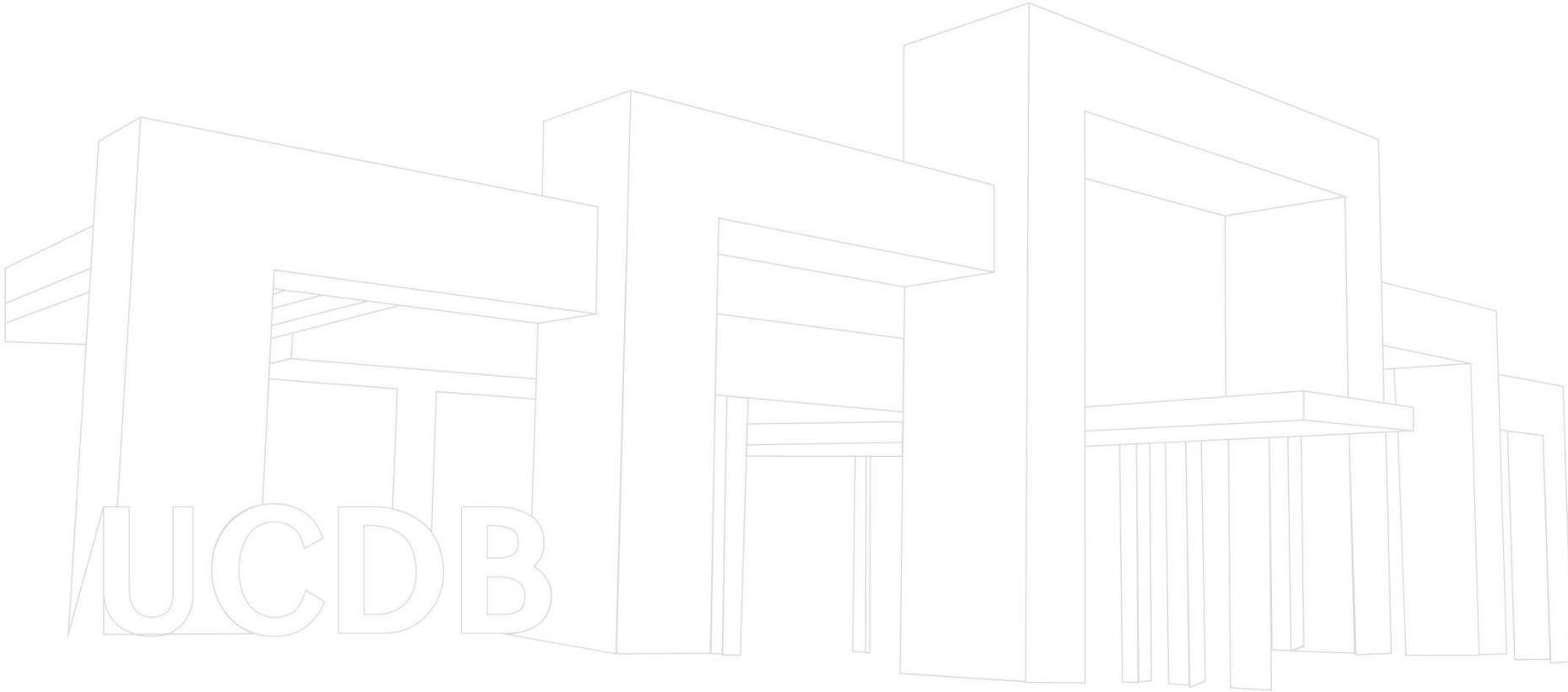
```
<p class=padre> Eu retiro o que disse, João </p>
<p class=grilo> Retirando ou não retirando, o fato é que o
cachorro enterrou-se em latim </p>
<p class=bispo> Um cachorro? Enterrado em latim? </p>
<p class=padre> Enterrado latindo, Senhor Bispo, Au, au, au, não
sabe? </p>
```

- Para dar a cada parágrafo de um mesmo personagem (mesma classe) os mesmos atributos, usa-se:

```
p.grilo { color: maroon }
p.padre { color: black }
p.bispo { color: navy }
```

CSS

Styling Background



CSS

Styling Background

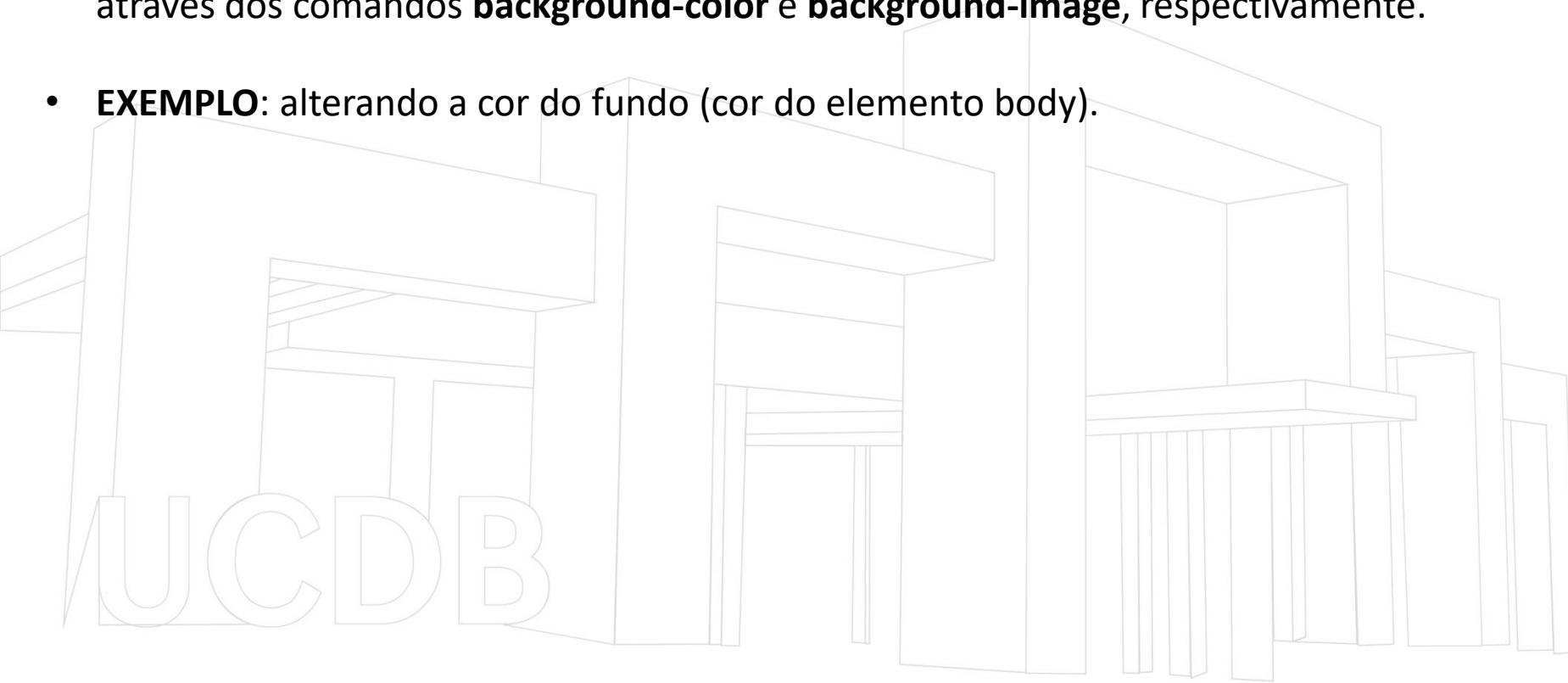
- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.



CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- **EXEMPLO:** alterando a cor do fundo (cor do elemento body).

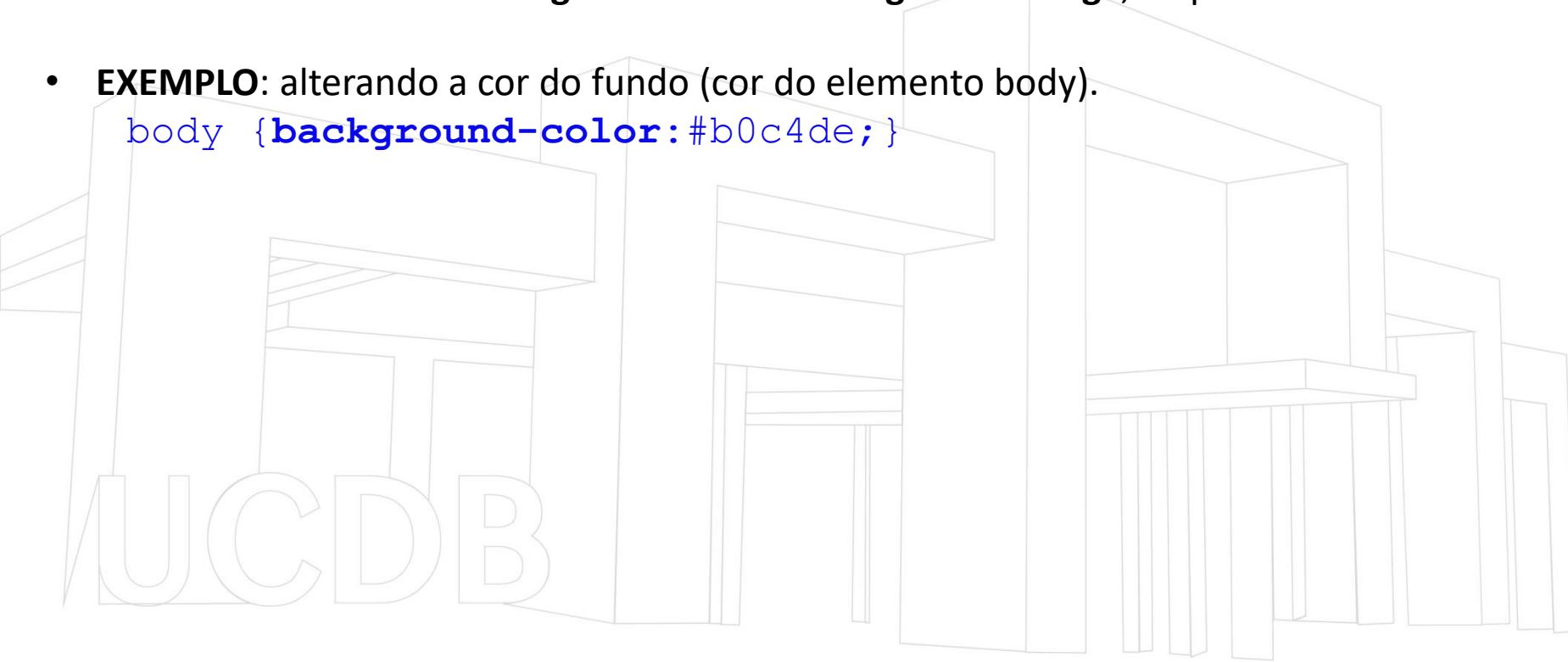


CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de; }
```

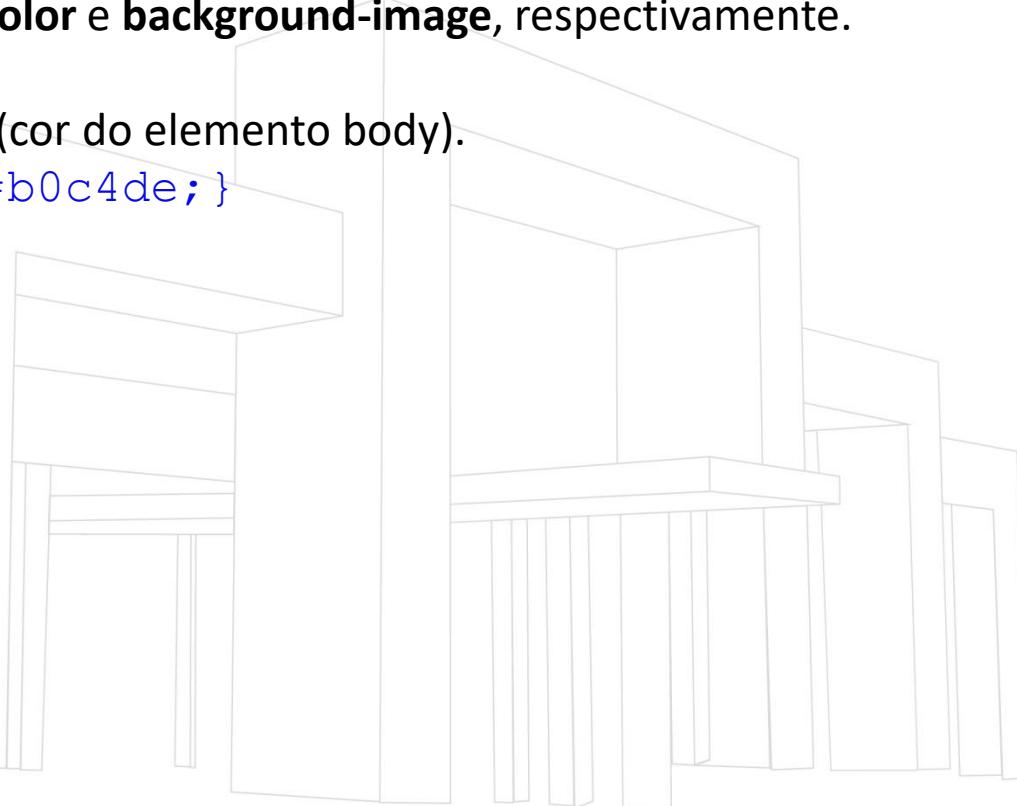


CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de;}
```



CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de;}
```



- As cores podem ser definidas de três maneiras diferentes:

CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de;}
```



- As cores podem ser definidas de três maneiras diferentes:
 - pelo nome da cor em inglês (ex: red);

CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de;}
```



- As cores podem ser definidas de três maneiras diferentes:
 - pelo nome da cor em inglês (ex: red);
 - por um valor RGB (Red, Green, Blue) (ex: rgb(255,0,0);

CSS

Styling Background

- As propriedades que podem ser modificadas e, ou, acrescentadas são **cor** e **imagem**, através dos comandos **background-color** e **background-image**, respectivamente.
- EXEMPLO:** alterando a cor do fundo (cor do elemento body).

```
body {background-color: #b0c4de;}
```



- As cores podem ser definidas de três maneiras diferentes:
 - pelo nome da cor em inglês (ex: red);
 - por um valor RGB (Red, Green, Blue) (ex: rgb(255,0,0);
 - ou pelo valor hexadecimal de uma cor (ex: #ff0000).

CSS

Styling Text



CSS

Styling Text

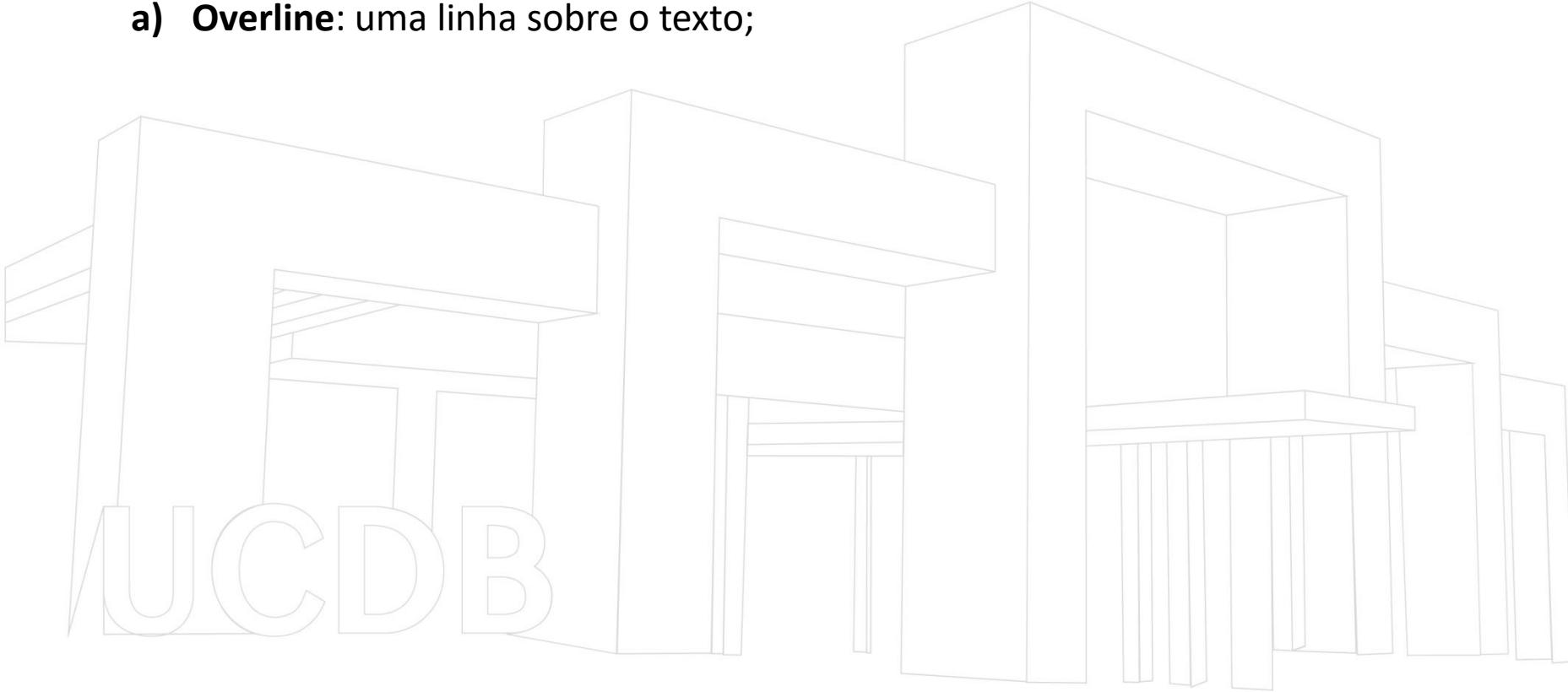
- DECORAÇÃO



CSS

Styling Text

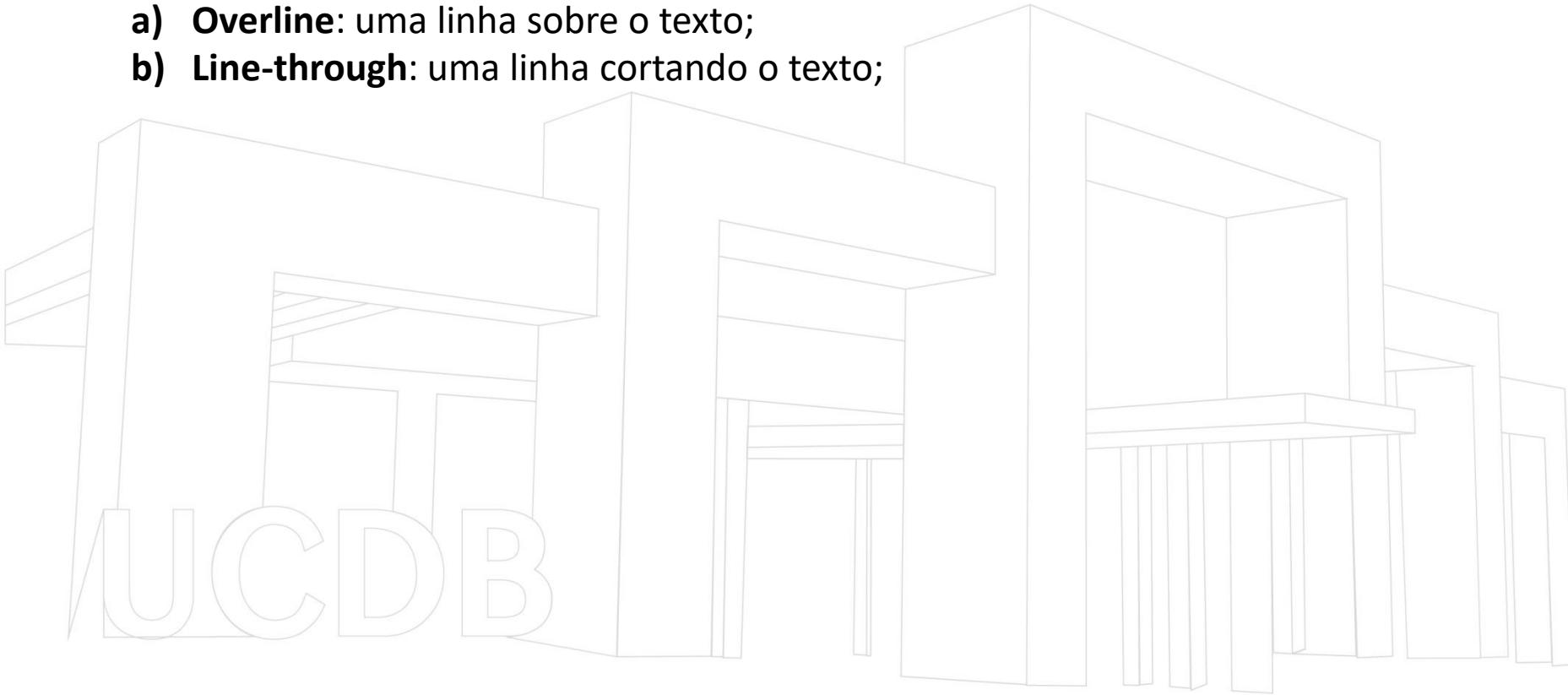
- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;



CSS

Styling Text

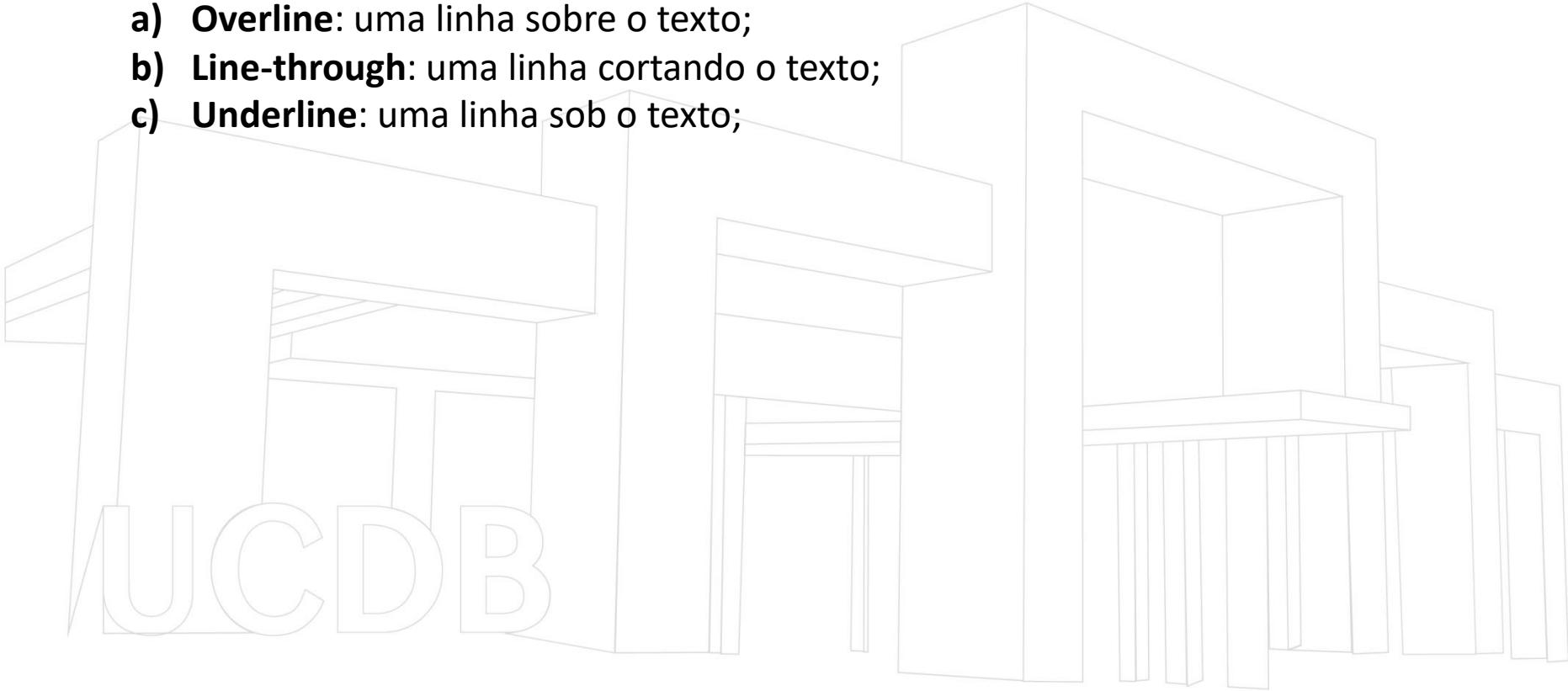
- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;



CSS

Styling Text

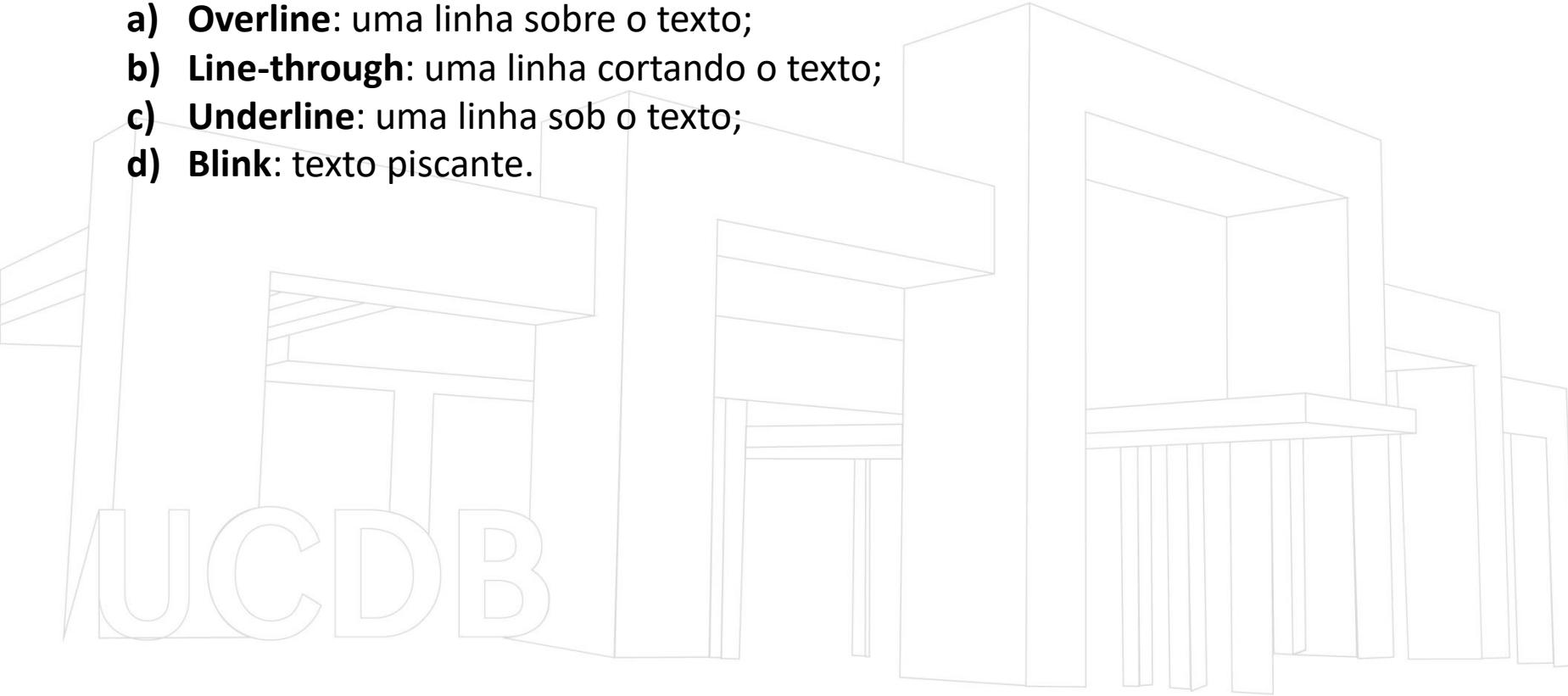
- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;



CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

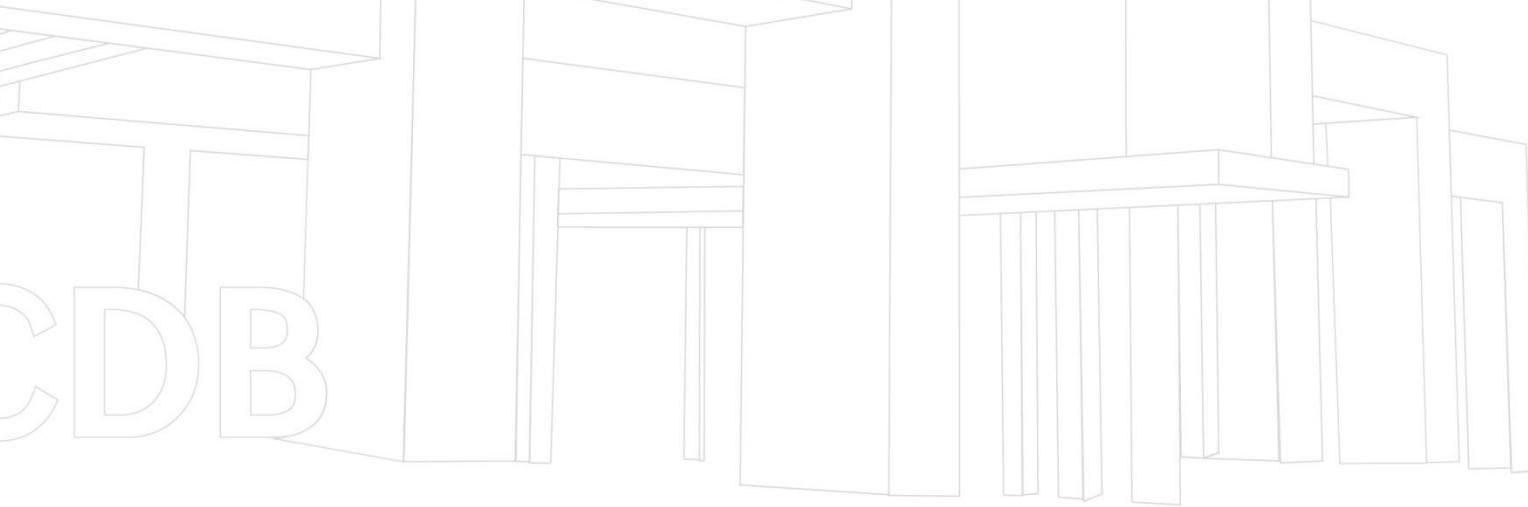


CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

EXEMPLO:



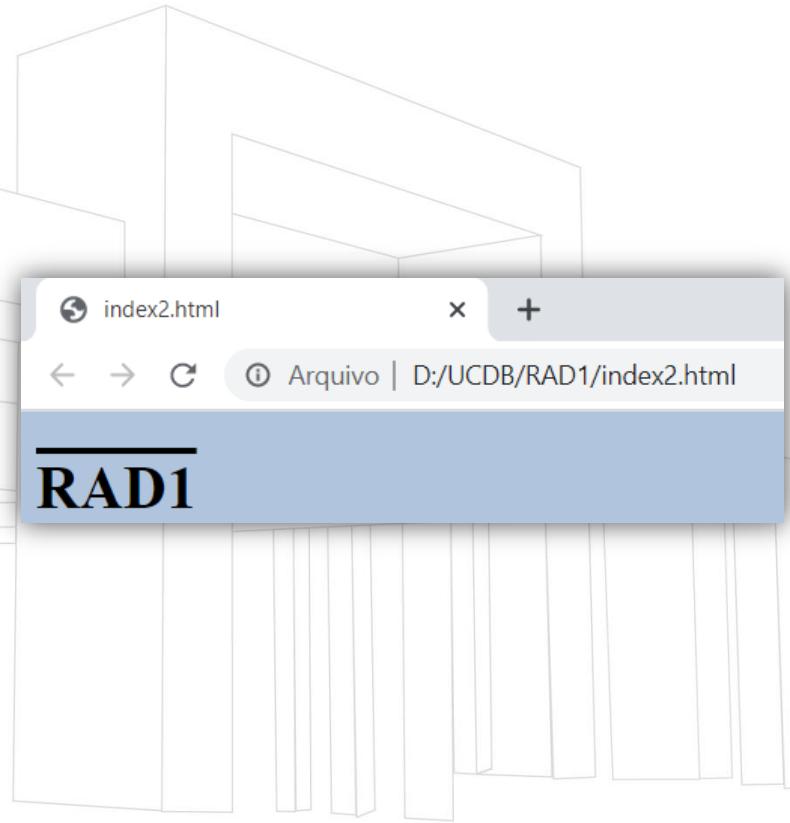
CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

EXEMPLO:

```
h1 {text-decoration:overline;}
```



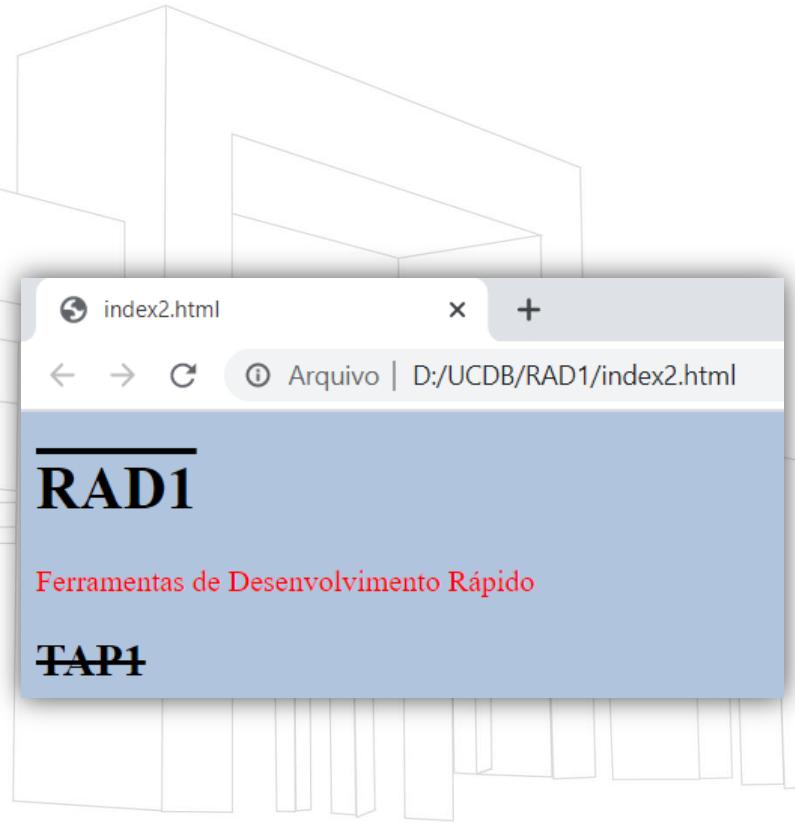
CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

EXEMPLO:

```
h1{text-decoration:overline;}  
h2{text-decoration:line-through;}
```



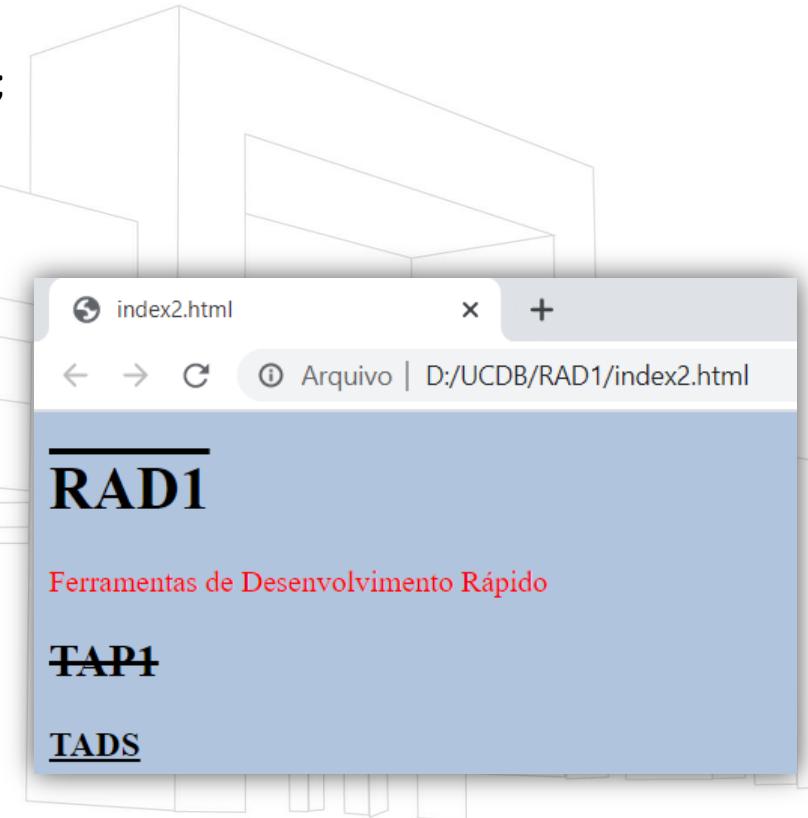
CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

EXEMPLO:

```
h1{text-decoration:overline;}  
h2{text-decoration:line-through;}  
h3{text-decoration:underline;}
```



CSS

Styling Text

- DECORAÇÃO
 - a) Overline: uma linha sobre o texto;
 - b) Line-through: uma linha cortando o texto;
 - c) Underline: uma linha sob o texto;
 - d) Blink: texto piscante.

EXEMPLO:

```
h1{text-decoration:overline;}  
h2{text-decoration:line-through;}  
h3{text-decoration:underline;}  
h4 {text-decoration:blink;}
```



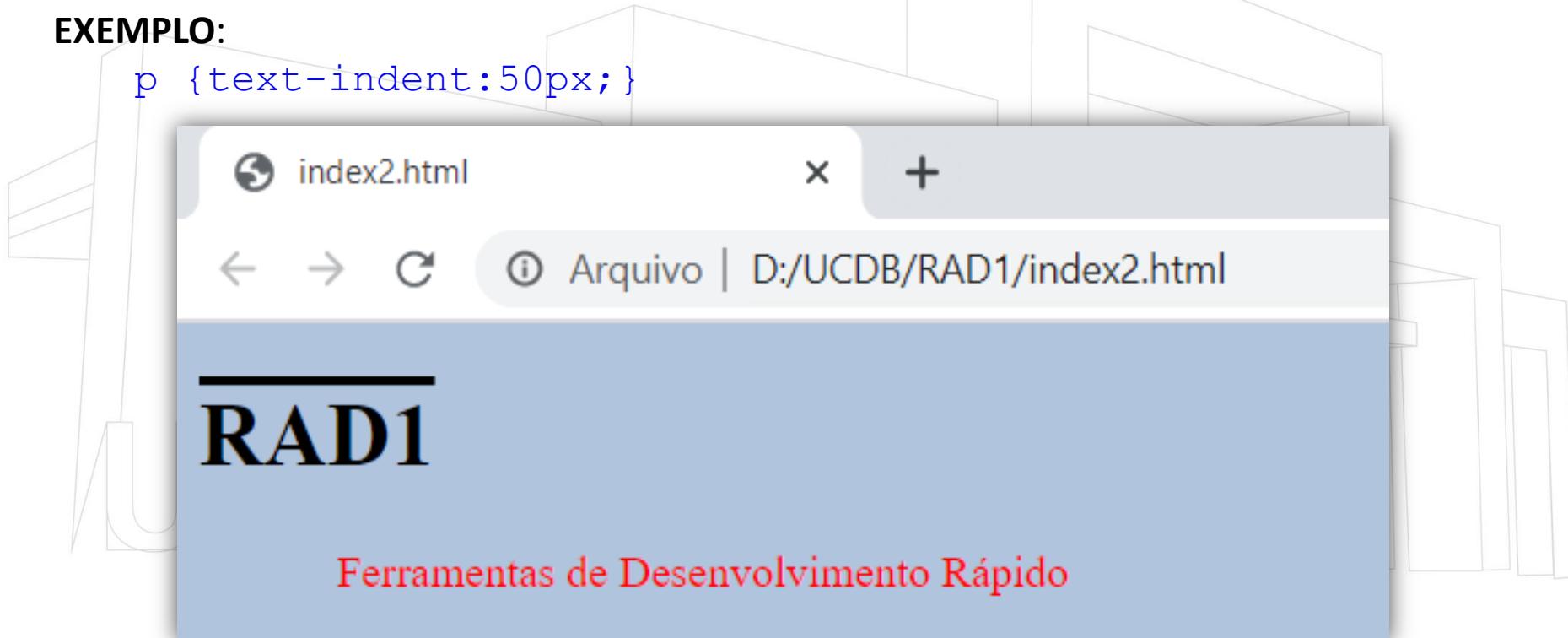
CSS

Styling Text

- **INDENTAÇÃO**
- espaço entre a primeira palavra de um parágrafo e a margem.
-

EXEMPLO:

```
p {text-indent:50px;}
```





3.3. Introdução ao JavaScript

Prof. Me. Marcos Alves
marcos@ucdb.br

JavaScript

Escopo de Variáveis



JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: *let*, *const* e *var*;



JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: *let*, *const* e *var*;

Palavra-chave	const	let	var
Escopo global	NÃO	NÃO	SIM

JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: *let*, *const* e *var*;

Palavra-chave	const	let	var
Escopo global	NÃO	NÃO	SIM
Escopo de função	SIM	SIM	SIM

JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: *let*, *const* e *var*;

Palavra-chave	<code>const</code>	<code>let</code>	<code>var</code>
Escopo global	NÃO	NÃO	SIM
Escopo de função	SIM	SIM	SIM
Escopo de bloco	SIM	SIM	NÃO

JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: *let*, *const* e *var*;

Palavra-chave	<i>const</i>	<i>let</i>	<i>var</i>
Escopo global	NÃO	NÃO	SIM
Escopo de função	SIM	SIM	SIM
Escopo de bloco	SIM	SIM	NÃO
Pode ser alterado	NÃO	SIM	SIM

JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: ***let*, *const* e *var***;

Palavra-chave	const	let	var
Escopo global	NÃO	NÃO	SIM
Escopo de função	SIM	SIM	SIM
Escopo de bloco	SIM	SIM	NÃO
Pode ser alterado	NÃO	SIM	SIM

- *Hoisting* → variáveis declaradas com a palavra-chave ***var*** podem ser utilizadas mesmo antes de sua declaração;

JavaScript

Escopo de Variáveis

- Está associado aos conceito de visibilidade, em três formas: ***let***, ***const*** e ***var***;

Palavra-chave	<i>const</i>	<i>let</i>	<i>var</i>
Escopo global	NÃO	NÃO	SIM
Escopo de função	SIM	SIM	SIM
Escopo de bloco	SIM	SIM	NÃO
Pode ser alterado	NÃO	SIM	SIM

- *Hoisting* → variáveis declaradas com a palavra-chave ***var*** podem ser utilizadas mesmo antes de sua declaração;
- Variáveis criadas com ***let*** só podem ser utilizadas após sua declaração, pois, apesar de serem elevadas, elas não são inicializadas.

JavaScript

Estruturas Condicionais



JavaScript

Estruturas Condicionais

```
if (condition) {  
}  
}
```



JavaScript

Estruturas Condicionais

```
if (condition) {  
}
```

```
if (condition) {  
  
} else {  
  
}
```

UCDB

JavaScript

Estruturas Condicionais

```
if (condition) {  
}  
}
```

```
if (condition) {  
  
} else {  
  
}
```

```
if ( preco > 100 ) {  
    console.log("Desconto de 10% liberado");  
} else if ( preco > 50 ) {  
    console.log("Desconto de 5% liberado");  
} else {  
    console.log("Nenhum desconto foi liberado");  
}
```

UCDB

JavaScript

Estruturas Condicionais

```
if (condition) {  
}  
}
```

```
if (condition) {  
  
} else {  
  
}
```

```
if ( preco > 100 ) {  
    console.log("Desconto de 10% liberado");  
} else if ( preco > 50 ) {  
    console.log("Desconto de 5% liberado");  
} else {  
    console.log("Nenhum desconto foi liberado");  
}
```

```
( condicao ? código1 : código2 )
```

JavaScript

Funções



JavaScript

Funções

```
function name(params) {  
}  
}
```



JavaScript

Funções

```
function name(params) {  
}  
/* Declare a função 'minhaFunção' */
```

```
function minhaFuncao(objeto) {  
    | | objeto.marca = "FIAT";  
}  
  
/*  
 * Declare a variável 'meucarro';  
 * crie e inicialize um novo Objeto;  
 * atribua referência para 'meucarro'  
 */  
var meucarro = {  
    marca: "FORD",  
    modelo: "TAURUS",  
    ano: 1998  
};
```



JavaScript

Funções

```
function name(params) {  
}  
/* Declare a função 'minhaFunção' */
```

```
function minhaFuncao(objeto) {  
    objeto.marca = "FIAT";  
  
    /*  
     * Declare a variável 'meucarro';  
     * crie e inicialize um novo Objeto;  
     * atribua referência para 'meucarro'  
     */  
  
    var meucarro = {  
        marca: "FORD",  
        modelo: "TAURUS",  
        ano: 1998  
    };  
}
```

```
/* Exibe 'FORD' */  
console.log(meucarro.marca);  
  
/* Passe a referência do objeto para a função */  
minhaFuncao(meucarro);  
  
/*  
 * Exibe 'FIAT' como valor para a propriedade 'marca'  
 * do objeto, mudado pela função.  
 */  
console.log(meucarro.marca);
```

JavaScript

Estruturas de Repetição



JavaScript

Estruturas de Repetição

```
while (condition) {  
}  
}
```



JavaScript

Estruturas de Repetição

```
while (condition) {  
}
```

```
do {  
}  
} while (condition);
```

UCDB

JavaScript

Estruturas de Repetição

```
while (condition) {  
}
```

```
for ( let index = 0; index < length; index++ ) {  
}
```

```
do {  
}  
} while (condition);
```

UCDB

JavaScript

Estruturas de Repetição

```
while (condition) {  
}  
}  
{
```

```
do {  
  
} while (condition);
```

```
for ( let index = 0; index < length; index++ ) {  
  
}
```

```
array.forEach(element => {  
  
});  
  
cores = ["Verde", "Amarelo", "Azul", "Branco"];  
function imprimir(item) {  
    console.log(item);  
}  
  
cores.forEach(imprimir);
```

UCDB

JavaScript

Arrays



JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```



JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

UCDB

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Remover um item do final

```
carros.pop();
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Remover um item do final

```
carros.pop();
```

- Remover um item do início

```
carros.shift();
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Remover um item do final

```
carros.pop();
```

- Remover um item do início

```
carros.shift();
```

- Adicionar um item ao início

```
carros.unshift( 'Opala' );
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Remover um item do final

```
carros.pop();
```

- Remover um item do início

```
carros.shift();
```

- Adicionar um item ao início

```
carros.unshift( 'Opala' );
```

- Remover um item pela posição

```
carros.splice(start: number, deleteCount?: number);
```

JavaScript

Arrays

- Criar um Array

```
var carros = [ 'Fusca', 'Brasilia'];

console.log(carros.length);
```

- Acessar um item (*index*) do Array

```
var primeiro = carros[ 0 ]; // Fusca
var ultimo = carros[ carros.length - 1 ];
```

- Iterar um Array

```
carros.forEach(function (item, indice) {
  console.log(item, indice);
});
// Fusca 0
// Brasilia 1
```

- Adicionar um item ao final

```
carros.push( 'Belina' );
```

- Remover um item do final

```
carros.pop();
```

- Remover um item do início

```
carros.shift();
```

- Adicionar um item ao início

```
carros.unshift( 'Opala' );
```

- Remover um item pela posição

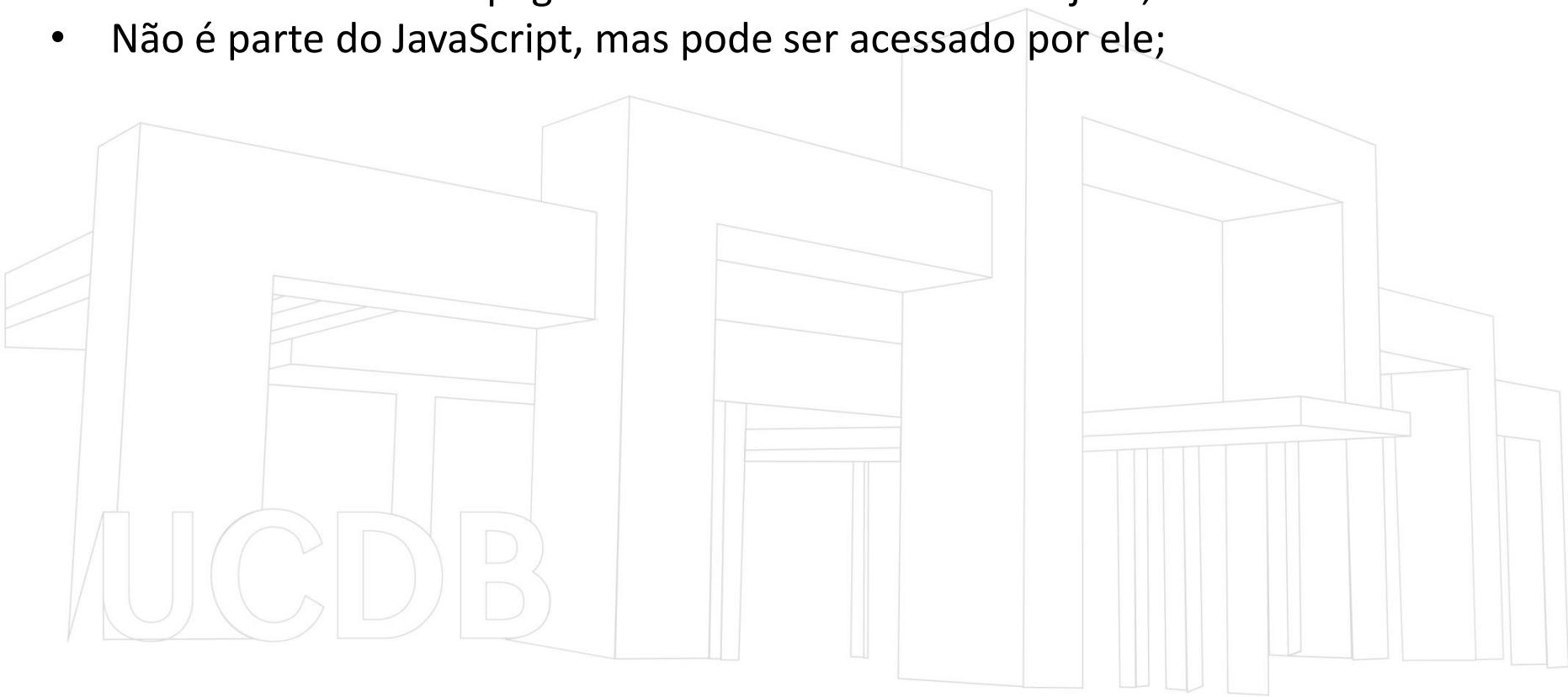
```
carros.splice(start: number, deleteCount?: number);
```

```
var removedItem = carros.splice( 2, 1 );
```

JavaScript

Manipulação do DOM (Document Object Model)

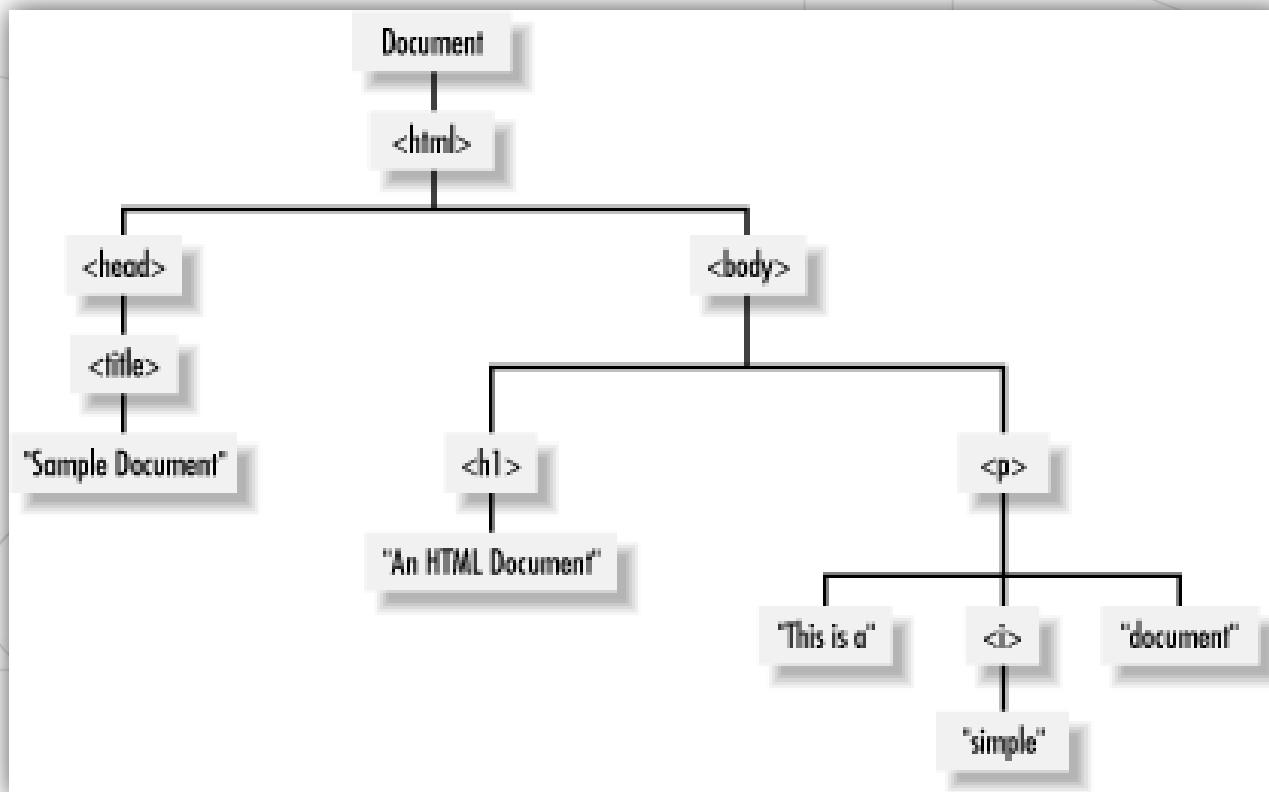
- Pode ser visto como página WEB “convertida” em objeto;
- Não é parte do JavaScript, mas pode ser acessado por ele;



JavaScript

Manipulação do DOM (Document Object Model)

- Pode ser visto como página WEB “convertida” em objeto;
- Não é parte do JavaScript, mas pode ser acessado por ele;



JavaScript

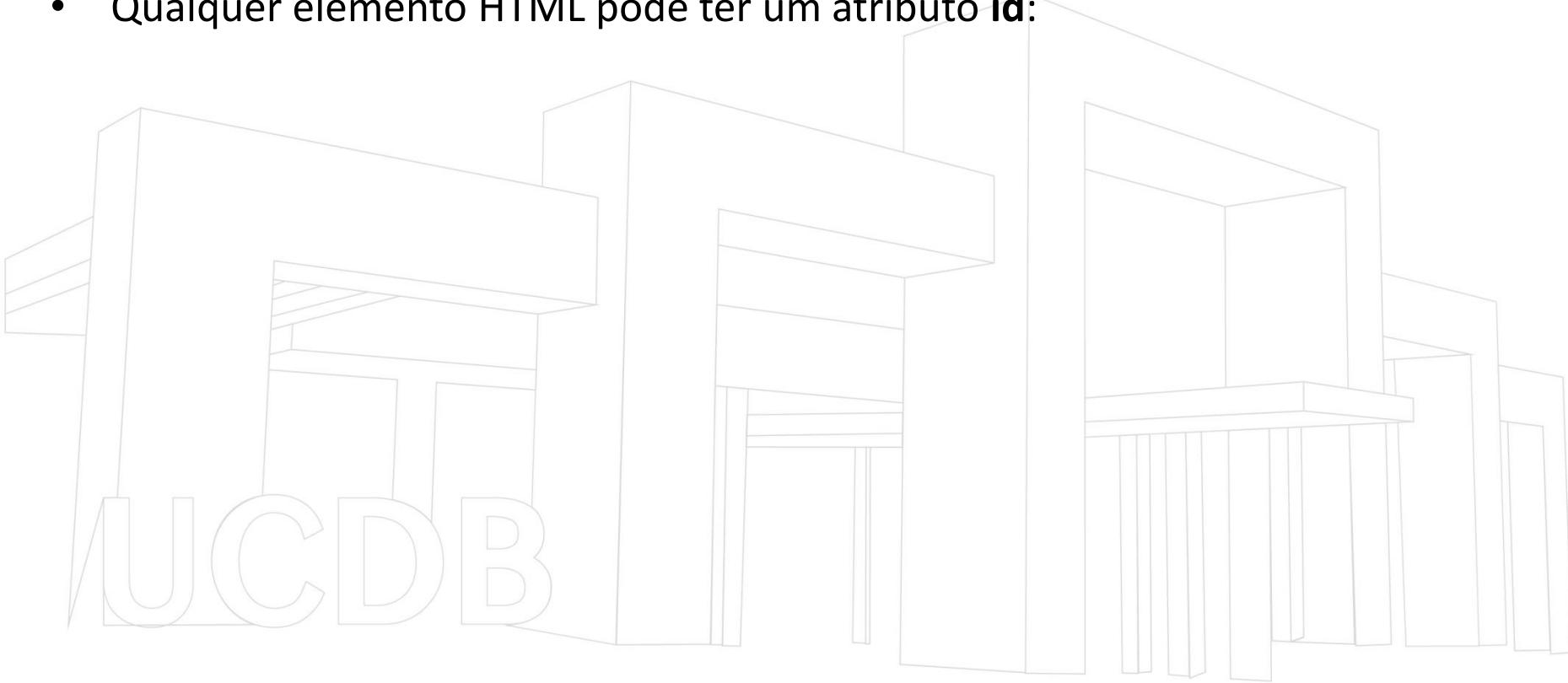
DOM: Métodos diretos para acessar o documento



JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

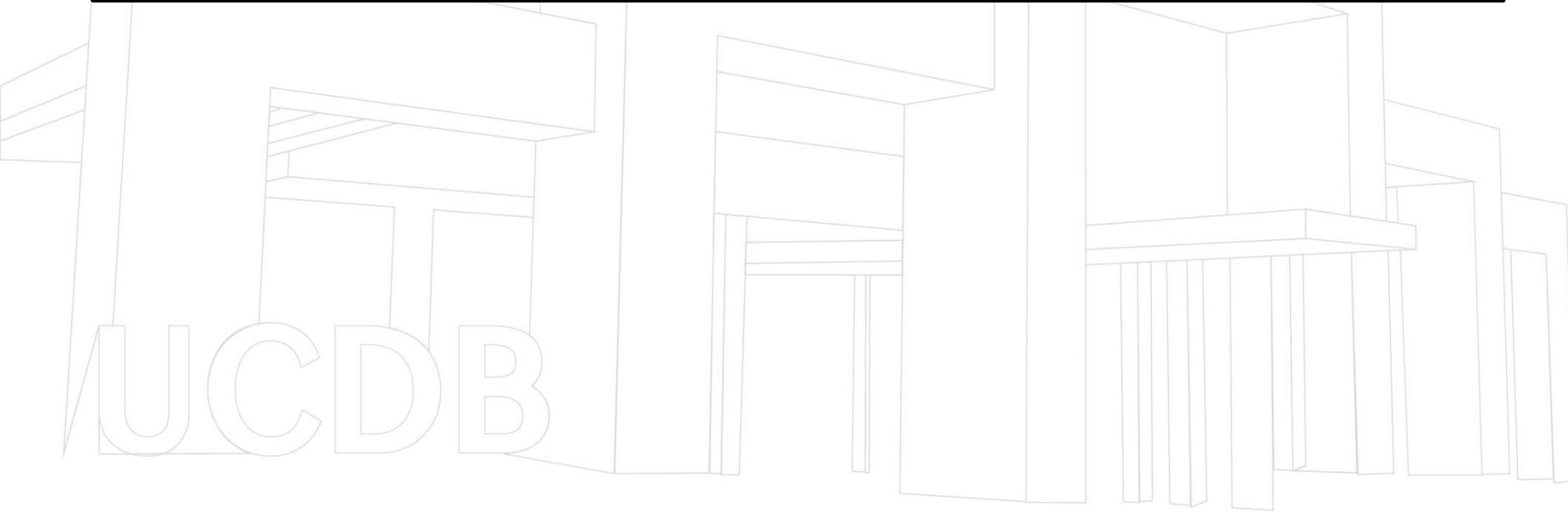


JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```



JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```

- Outros métodos de nomes e funções intuitivas:

JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```

- Outros métodos de nomes e funções intuitivas:

- [getElementsByName\(\)](#)

JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```

- Outros métodos de nomes e funções intuitivas:

- [getElementsByName\(\)](#)
- [getElementsByTagName\(\)](#)

JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```

- Outros métodos de nomes e funções intuitivas:

- [getElementsByName\(\)](#)
- [getElementsByTagName\(\)](#)
- [getElementsByClassName\(\)](#)

JavaScript

DOM: Métodos diretos para acessar o documento

- Qualquer elemento HTML pode ter um atributo **id**:

```
var elemento = document.getElementById("identificacao");
```

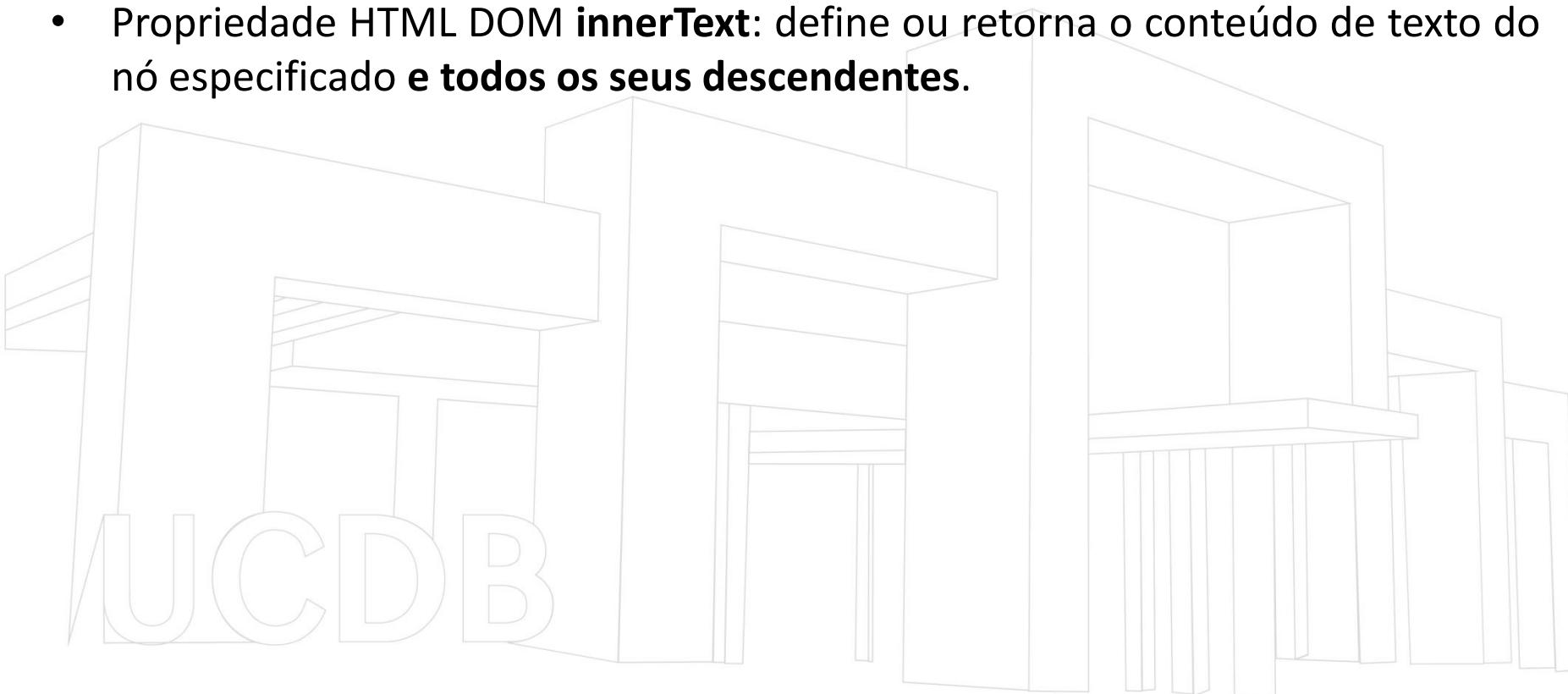
- Outros métodos de nomes e funções intuitivas:

- [getElementsByName\(\)](#)
- [getElementsByTagName\(\)](#)
- [getElementsByClassName\(\)](#)
- [querySelectorAll\(\)](#)

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.



JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

```
let txt = node.innerText
```

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

```
let txt = node.innerText
```

- b) Defina o conteúdo de texto de um nó:

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

```
let txt = node.innerText
```

- b) Defina o conteúdo de texto de um nó:

```
node.innerText = text
```

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM **innerText**: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

```
let txt = node.innerText
```

- b) Defina o conteúdo de texto de um nó:

```
node.innerText = text
```

Valores de propriedade

JavaScript

DOM: Métodos diretos para acessar o documento

- Propriedade HTML DOM `innerText`: define ou retorna o conteúdo de texto do nó especificado **e todos os seus descendentes**.

Sintaxe

- a) Retorne o conteúdo de texto de um nó:

```
let txt = node.innerText
```

- b) Defina o conteúdo de texto de um nó:

```
node.innerText = text
```

Valores de propriedade

Valor	Tipo	Descrição
texto	String	conteúdo do nó especificado (texto)

JavaScript

EXEMPLO

JS

```
s_exe > JS cadastro.js > ⚒ cadastrarViolao
1  var arrViolao = [];
2
3  function cadastrarViolao(){
4      //ler dados da página
5      let nome = document.getElementById('nome').value;
6      let violao = document.getElementById('violao').value;
7
8      //avisar preenchimento obrigatório
9      if( nome == ''){
10          alert('Preencha as informações: nome')
11      } else if (violao == ''){
12          alert('Preencha as informações: violão ')
13      } else {
14          //criar objetos
15          let vl = {marca: nome, modelo: violao};
16
17          //limpar campos na página
18          document.getElementById('nome').value = '';
19          document.getElementById('violao').value = '';
20
21          //adicionar os arrays
22          arrViolao.push(vl);
23      }
24 }
```

JavaScript

EXEMPLO

```
26  function listarViolao(){
27      // criar variavel
28      let saidaHTML = '';
29      //criar o for
30      for(i=0; i < arrViolao.length; i++){
31          |  saidaHTML = saidaHTML + `<div> ${arrViolao[i].marca} ${arrViolao[i].modelo} </div>`;
32      }
33      document.getElementById('saida').innerHTML = saidaHTML;
34 }
```

```
36  function limparViolao(){
37      if (arrViolao){
38          |  arrViolao = [];
39          |  document.getElementById('saida').innerHTML = '';
40      }
41 }
```

JavaScript

EXEMPLO

HTML

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Cadastro</title>
7      <script src='cadastro.js'></script>
8      <link rel='stylesheet' href='cadastro.css'>
9
10 </head>
11 <body>
12     <div id='box'>
13         <div>
14             <img src='https://i.pinimg.com/originals/f9/18/9c/f9189ce3f7e6161f76b5bfb904ac1e68.jpg'>
15         </div>
16         <div><h1>Cadastrar Violão</h1></div>
17
18         <div><input id='nome' placeholder='Nome'></div>
19         <div><input id='violao' placeholder="Violão"></div>
20
21         <button onclick = 'cadastrarViolao()'>Cadastrar</button>
22         <button onclick = 'listarViolao()'>Listar</button>
23         <button onclick = 'limparViolao()'>Limpar campos</button>
24
25         <div id='saida'></div>
26     </div>
27
28     <div id='box2'>
29         <div id='link'>
30             <a href="projetos.html">
31                 Ver mais cadastros
32             </a>
33         </div>
34     </div>
35 </div>
36
37
38 </body>
39 </html>
```