



# Programação para Internet

---

## Módulo 1

### Introdução à Linguagem HTML

Prof. Dr. Daniel A. Furtado - FACOM/UFU

Conteúdo protegido por direito autoral, nos termos da Lei nº 9 610/98

A cópia, reprodução ou apropriação deste material, total ou parcialmente, é proibida pelo autor

# Conteúdo do Módulo

- Introdução ao desenvolvimento Web com a HTML5
- Estrutura de um documento HTML
- Elementos e recursos mais comuns da linguagem
- Colocando página online com serviço de hospedagem gratuito
- Categorias de elementos e conteúdo embutido
- Elementos semânticos de estrutura e layout
- Validação do código HTML e entidades
- Formulários

# O que é HTML?

- Criação de páginas Web
- Hypertext Markup Language (Linguagem de Marcação de Hipertexto)
- Definição do conteúdo da página (títulos, parágrafos, links, imagens, etc.)
- Estilos de exibição da página são deixados para a CSS
- Arquivo com extensão .html (ou .htm)
- Processada pelo navegador de Internet
- NÃO é linguagem de programação
- Comumente utilizada com outras tecnologias como CSS, JavaScript, etc.

# Breve Histórico

- 1989: Nascimento da Web no CERN (proposta de Tim Berners Lee)
- 1991: Surgimento da HTML (Tim publica o documento HTML Tags)
- 1994: Tim cria o consórcio W3C (World Wide Web Consortium)
- 1999: HTML 4.01 (recomendação W3C)
- 2000: XHTML 1.0 (eXtensible Hypertext Markup Language: reformulação da HTML com base na XML)
- 2004: Criação do WHATWG (Web Hypertext Application Technology Working Group)
- 2008: HTML5 (WHATWG)
- 2014: HTML5 (recomendação W3C)
- 2019: W3C e WHATWG assinam acordo
  - WHATWG se torna o principal responsável pelo desenvolvimento da HTML, de forma contínua, seguindo seu padrão conhecido como *living standard*
  - <https://html.spec.whatwg.org/print.pdf>

# Primeira Página HTML

## Código HTML

```
<!DOCTYPE html>
<html>

  <head>
    <title>Titulo da Pagina</title>
  </head>

  <body>

    <h1>Primeira Pagina</h1>
    <p>Ola, mundo!</p>

  </body>

</html>
```

## Resultado no Navegador



# Elementos HTML

```
<!DOCTYPE html>
<html>

  <head>
    <title>Titulo da Pagina</title>
  </head>

  <body>

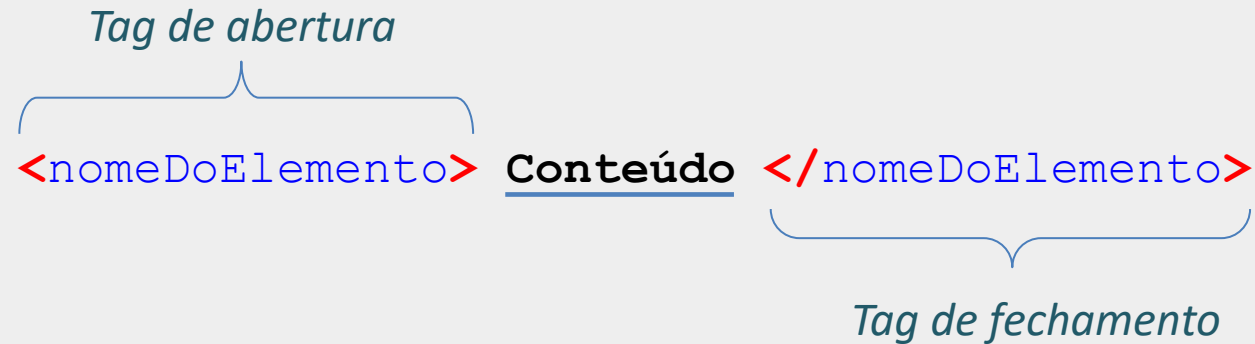
    <h1>Primeira Pagina</h1>
    <p>Ola, mundo!</p>

  </body>

</html>
```

Um documento HTML é composto por uma coleção de **elementos** HTML, onde cada elemento exerce um papel na página. No exemplo acima temos os elementos **html**, **head**, **title**, **body**, **h1** e **p**

# Elementos HTML



- A maioria dos elementos é inserida colocando-se uma **tag de abertura**, um **conteúdo** e uma **tag de fechamento**
- Mas há também elementos sem conteúdo (elementos vazios)
  - Sem tag de fechamento (fechamento implícito)
  - Um exemplo é o `<img>`, para inserção de imagem
  - **OBS:** sintaxe da XHTML, com barra no final (`<img/>`), não é recomendada na HTML5

# Estrutura Básica de um Documento HTML

*Início do documento HTML*

```
<!DOCTYPE html>  
<html>
```

*Cabeçalho do documento*

```
  <head>  
    <title>Titulo da Pagina</title>  
  </head>
```

*Corpo do documento*

```
  <body>  
    <h1>Primeira Pagina</h1>  
    <p>Ola, mundo!</p>
```

*Fim do documento HTML*

```
  </body>  
</html>
```

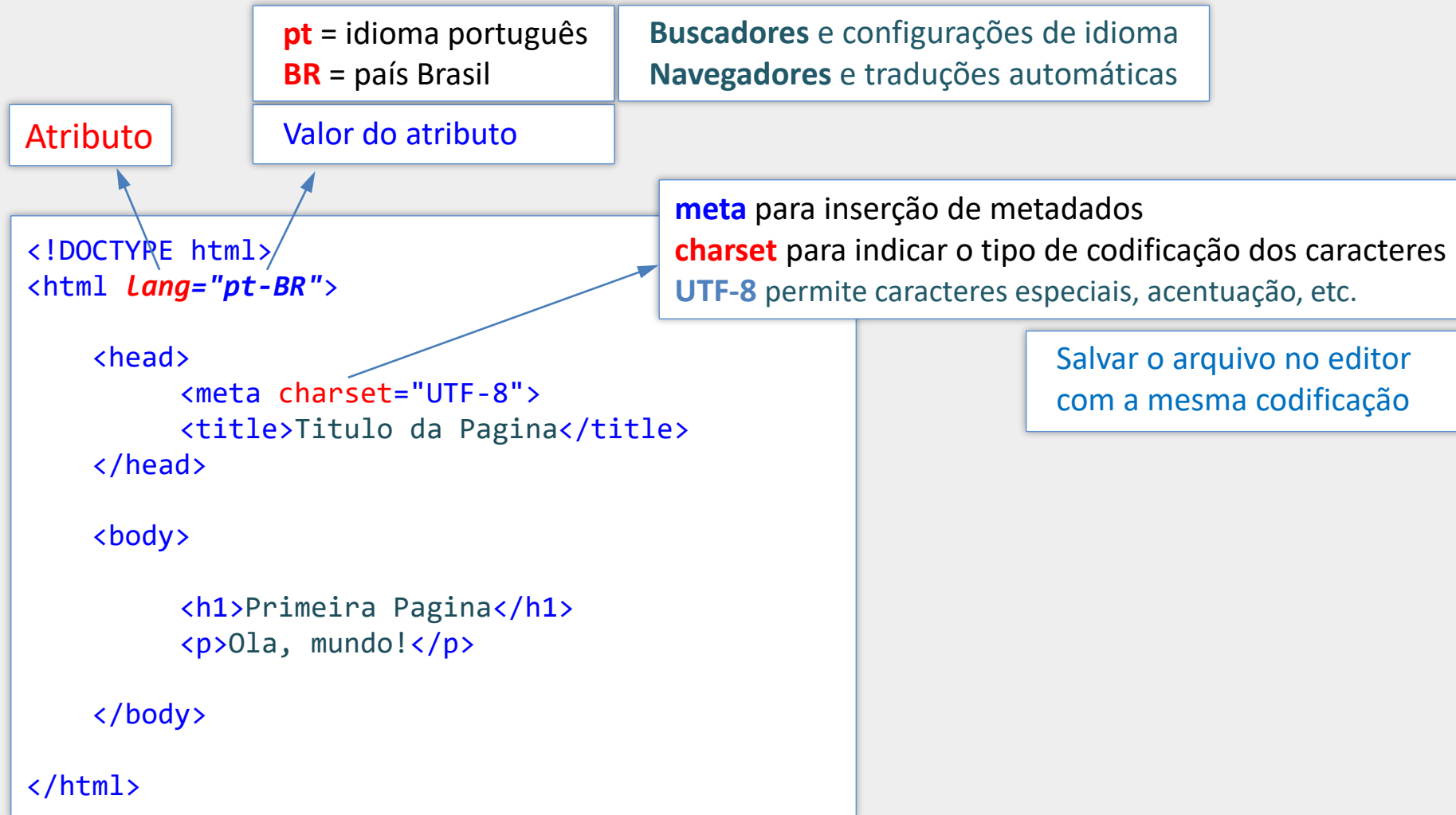
Indica que é um documento HTML5

O cabeçalho, `<head>`, normalmente inclui dados de configuração do documento como título, tipo de codificação dos caracteres, links com outros arquivos etc. Não aparece na página. O título, `<title>`, aparecerá na aba do navegador e nos resultados dos buscadores

O corpo, `<body>`, é a região onde se coloca o conteúdo propriamente dito da página HTML, o qual será apresentado ao usuário.



# Estrutura Básica de um Documento HTML



# Hierarquia de Elementos e Árvore DOM

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

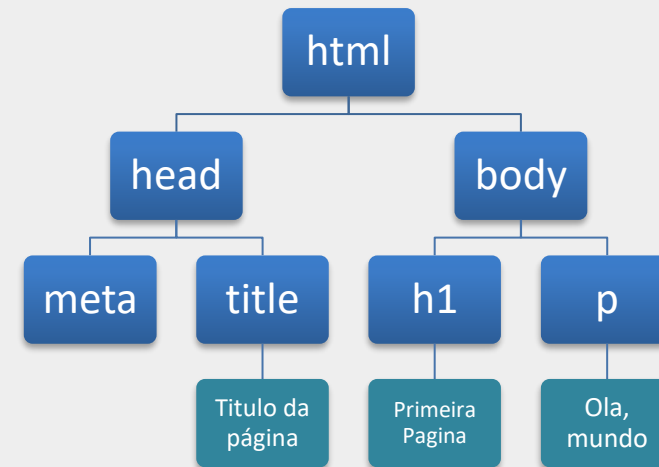
  <head>
    <meta charset="UTF-8">
    <title>Titulo da Pagina</title>
  </head>

  <body>

    <h1>Primeira Pagina</h1>
    <p>Ola, mundo!</p>

  </body>

</html>
```



Árvore DOM (simplificada)

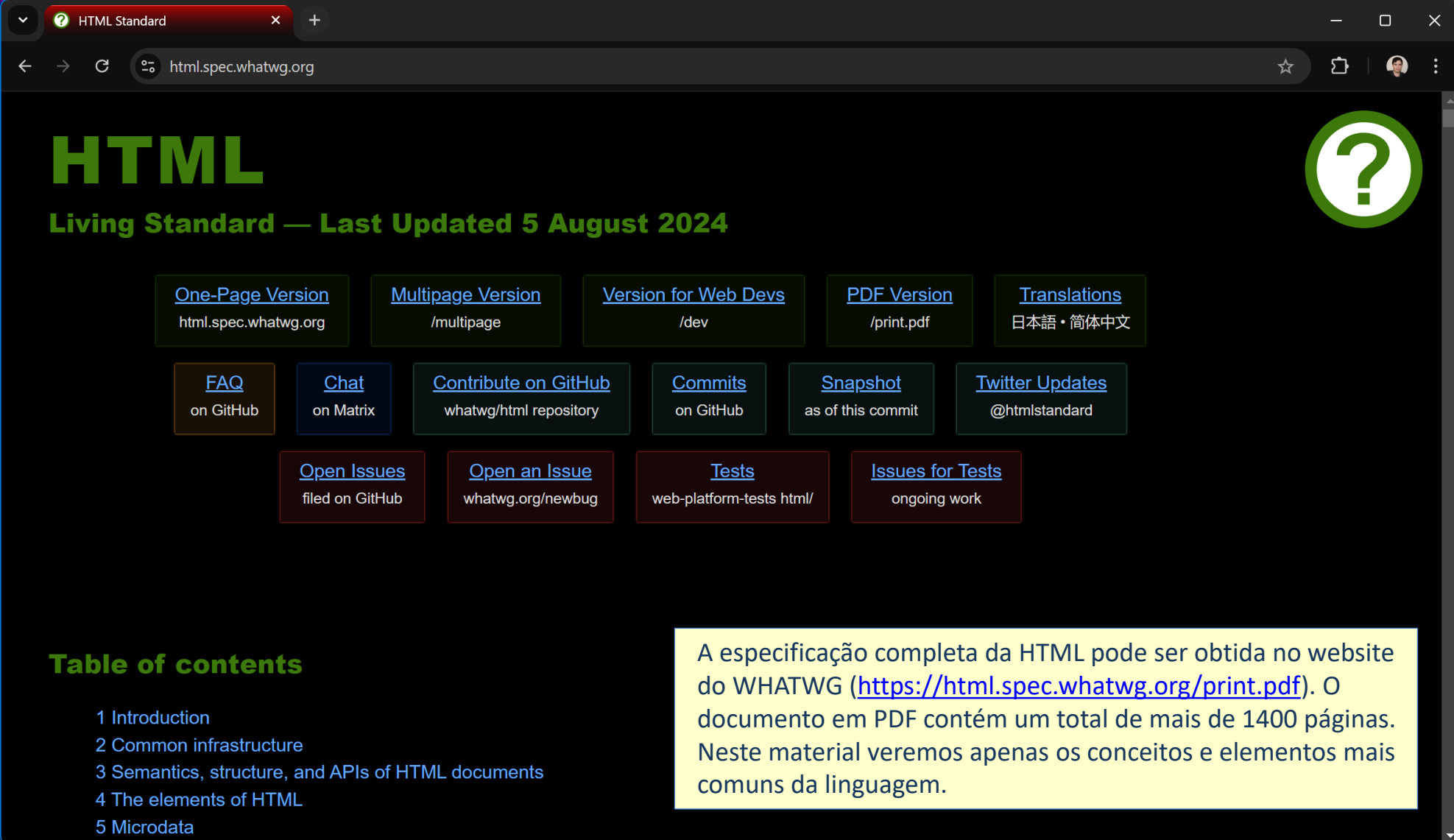
# Observações sobre Elementos e Atributos

- Elementos podem ter vários atributos
  - ``
- Alguns atributos não precisam de valor explícito (atributo vazio)
  - `<input type="text" disabled>`  
é equivalente a
    - `<input type="text" disabled="">`
- Aspas simples ou duplas são aceitas, mas aspas duplas são mais comuns
  - `<meta charset="UTF-8">`
  - `<meta charset='UTF-8'>`
- Em alguns casos pode-se colocar o valor sem aspas

# Observações sobre Elementos e Atributos

- HTML não é sensível à caixa para nomes de elementos e atributos
  - W3C recomenda letras minúsculas
- A tag de fechamento pode ser omitida em alguns casos
  - Porém não é recomendado pelo W3C
- A indentação é recomendada
  - No Visual Studio Code, pressione SHIFT + ALT + F

# Especificação da HTML no Website do WHATWG



The screenshot shows the homepage of the HTML Standard website. The page features a dark theme with green and blue accents. At the top, the word "HTML" is displayed in large green letters, followed by the subtitle "Living Standard — Last Updated 5 August 2024". A green circular logo with a white question mark is positioned in the top right corner. Below the header, there are several navigation buttons arranged in a grid. These buttons provide links to different versions of the specification (One-Page, Multipage, Version for Web Devs, PDF), translations (Japanese and Simplified Chinese), and community resources (FAQ, Chat, GitHub repository, Commits, Snapshot, Twitter updates, Open Issues, and Tests). At the bottom left, a "Table of contents" section lists the main sections of the specification: Introduction, Common infrastructure, Semantics, structure, and APIs of HTML documents, The elements of HTML, and Microdata. A yellow callout box on the right side of the page contains text explaining that the complete HTML specification can be found in a PDF document on the WHATWG website, which is over 1400 pages long.

**HTML**  
Living Standard — Last Updated 5 August 2024

[One-Page Version](#)  
html.spec.whatwg.org

[Multipage Version](#)  
/multipage

[Version for Web Devs](#)  
/dev

[PDF Version](#)  
/print.pdf

[Translations](#)  
日本語・简体中文

[FAQ](#)  
on GitHub

[Chat](#)  
on Matrix

[Contribute on GitHub](#)  
whatwg/html repository

[Commits](#)  
on GitHub

[Snapshot](#)  
as of this commit

[Twitter Updates](#)  
@htmlstandard

[Open Issues](#)  
filed on GitHub

[Open an Issue](#)  
whatwg.org/newbug

[Tests](#)  
web-platform-tests html/

[Issues for Tests](#)  
ongoing work

**Table of contents**

- 1 Introduction
- 2 Common infrastructure
- 3 Semantics, structure, and APIs of HTML documents
- 4 The elements of HTML
- 5 Microdata

A especificação completa da HTML pode ser obtida no website do WHATWG (<https://html.spec.whatwg.org/print.pdf>). O documento em PDF contém um total de mais de 1400 páginas. Neste material veremos apenas os conceitos e elementos mais comuns da linguagem.

# Elementos Textuais Básicos da HTML

## **Faculdade de Computação da UFU**

A Faculdade de Computação (FACOM) da Universidade Federal de Uberlândia foi criada em 2000, a partir do extinto Departamento de Informática (DEINF), criado em 1988, no âmbito do CETEC, com docentes provenientes dos departamentos de Engenharia Elétrica e de Matemática.

No início dos anos 2000 foi criado na Faculdade o Programa de Pós-Graduação em Ciência da Computação.

### **Cursos de Graduação**

A FACOM oferece os cursos de Bacharelado em Ciência da Computação, em Uberlândia, e Bacharelado em Sistemas de Informação, em Uberlândia e Monte Carmelo.

### **Cursos de Pós-Graduação**

A FACOM oferece também os cursos de Mestrado Acadêmico e Doutorado em Ciência da Computação.

*Av. João Naves de Ávila, 2121, Santa Mônica  
Uberlândia, MG  
CEP 38400-902*

# Títulos e Parágrafos

`<p>`

- Inserção de parágrafos de texto
- Começam com nova linha
- Espamento antes e depois

`<h1>`

- Para títulos principais (de nível 1)
- Por padrão, exibido em fonte maior e negrito

`<h2>`

- Para subtítulos (de nível 2)

`<h3>`, `<h4>`, `<h5>`, `<h6>`

- Para títulos de nível 3, 4, 5 e 6, respectivamente

# Elementos <abbr>, <address>, <pre> e <br>

## <abbr>

- Para abreviações ou acrônimos
- Atributo **title** – significado da abreviação

## <address>

- Para informações de contato (geral ou específico)

## <pre>

- Texto com manutenção de espaços e quebras de linha
- Situações especiais

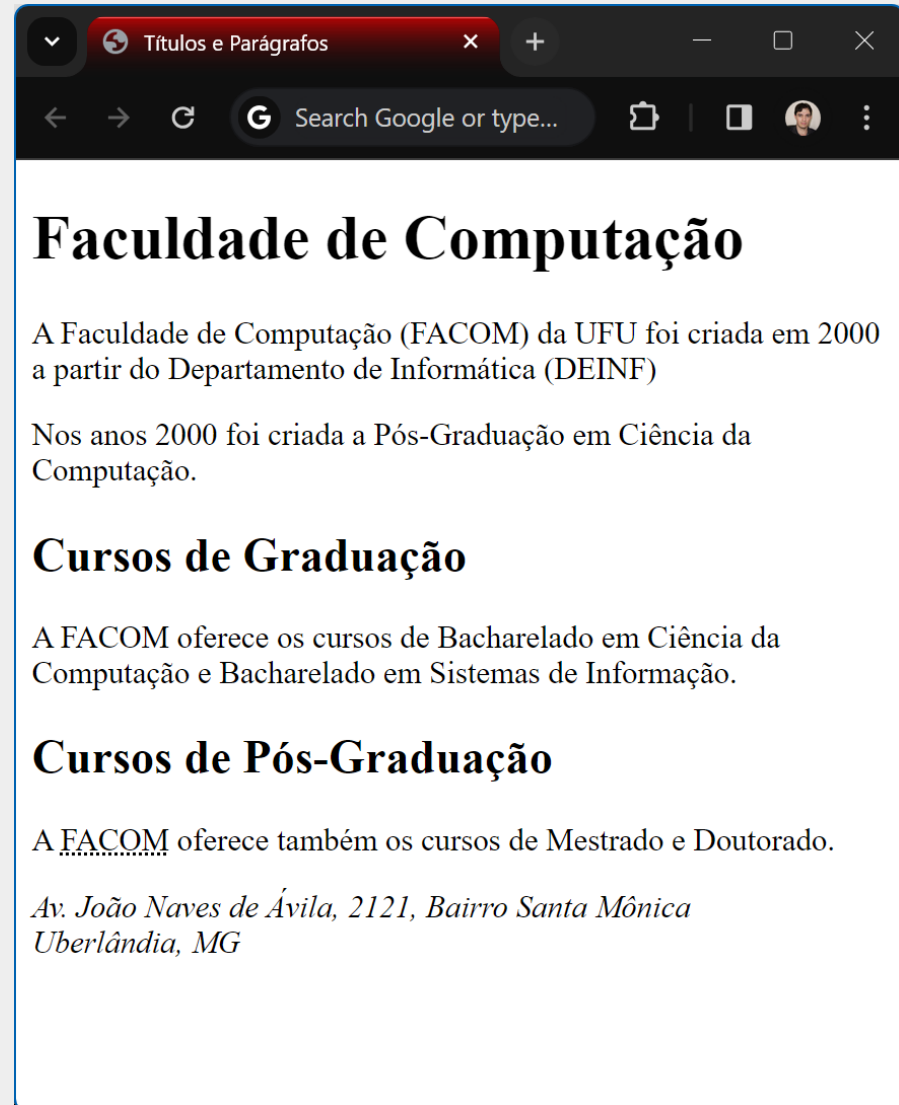
## <br>

- Quebra de linha (apenas para situações especiais)
- Deve ser evitado



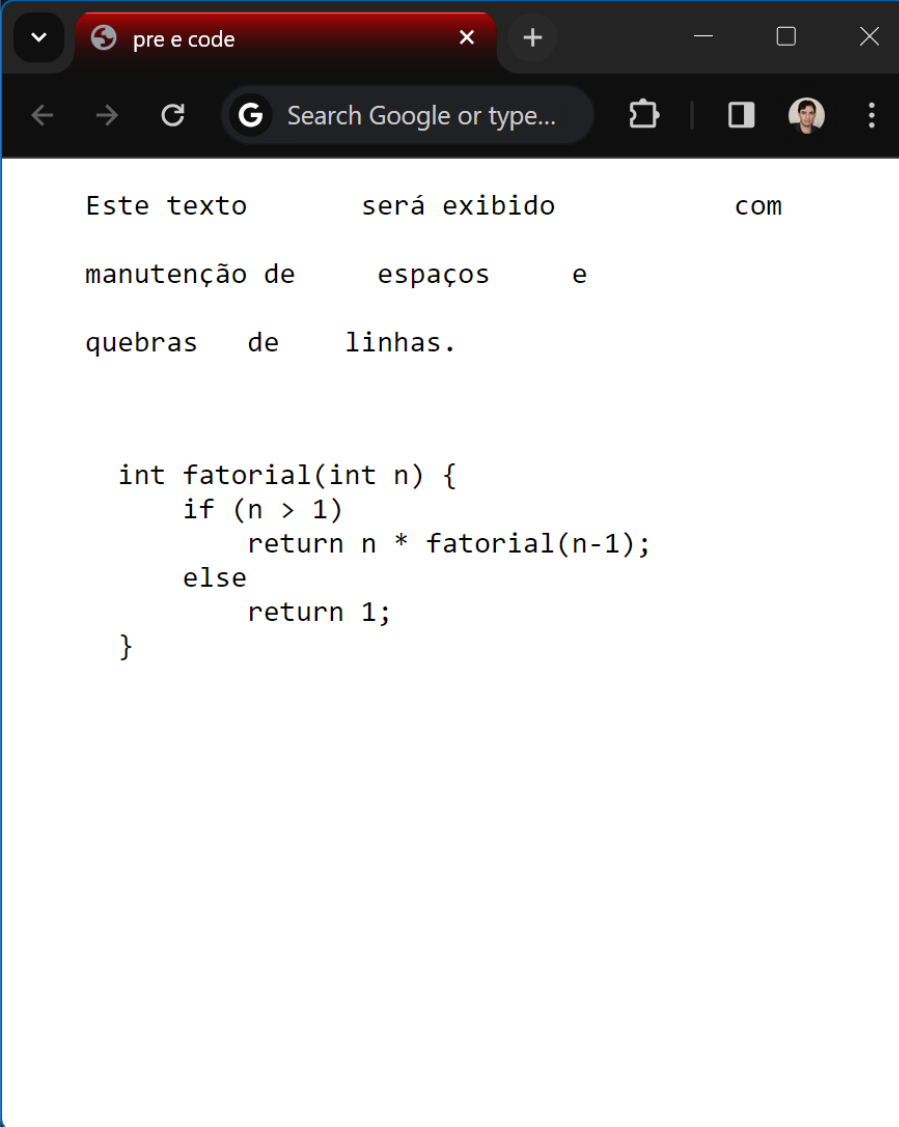
# Exemplo de Elementos Textuais

```
...
<body>
  <h1>Faculdade de Computação da UFU</h1>
  <p>A Faculdade de Computação (FACOM) da UFU foi criada em
    2000 a partir do Departamento de Informática (DEINF)</p>
  <p>Nos anos 2000 foi criada a Pós-Graduação
    em Ciência da Computação.</p>
  <h2>Cursos de Graduação</h2>
  <p>A FACOM oferece os cursos de Bacharelado em Ciência da
    Computação e Bacharelado em Sistemas de Informação.</p>
  <h2>Cursos de Pós-Graduação</h2>
  <p>A <abbr title="Faculdade de Computação">FACOM</abbr>
    oferece também os cursos de Mestrado e Doutorado.</p>
  <address>
    Av. João Naves de Ávila, 2121, Bairro Santa Mônica <br>
    Uberlândia, MG
  </address>
  <!-- Este é um comentário em HTML e não aparecerá -->
</body>
</html>
```



# Exemplo de Elementos Textuais

```
<body>
  <pre>
    Este texto      será exibido      com
    manutenção de   espaços          e
    quebras de     linhas.
  </pre>
  <pre>
    <code>
      int fatorial(int n) {
        if (n > 1)
          return n * fatorial(n-1);
        else
          return 1;
      }
    </code>
  </pre>
</body>
```



```
pre e code
Search Google or type...
Este texto      será exibido      com
manutenção de   espaços          e
quebras de     linhas.

int fatorial(int n) {
  if (n > 1)
    return n * fatorial(n-1);
  else
    return 1;
}
```

# Citações com <q>, <blockquote> e <cite>

## <q>

- Citações curtas
- Apresentada com aspas

## <blockquote>

- Citações longas
- Apresentada com recuo

## <cite>

- Citação de títulos de trabalhos criativos como nomes de livros, filmes etc.
- Apresentada em itálico

## <hr>

- Mudança temática a nível de parágrafo
- Apresentado como barra horizontal (padrão)

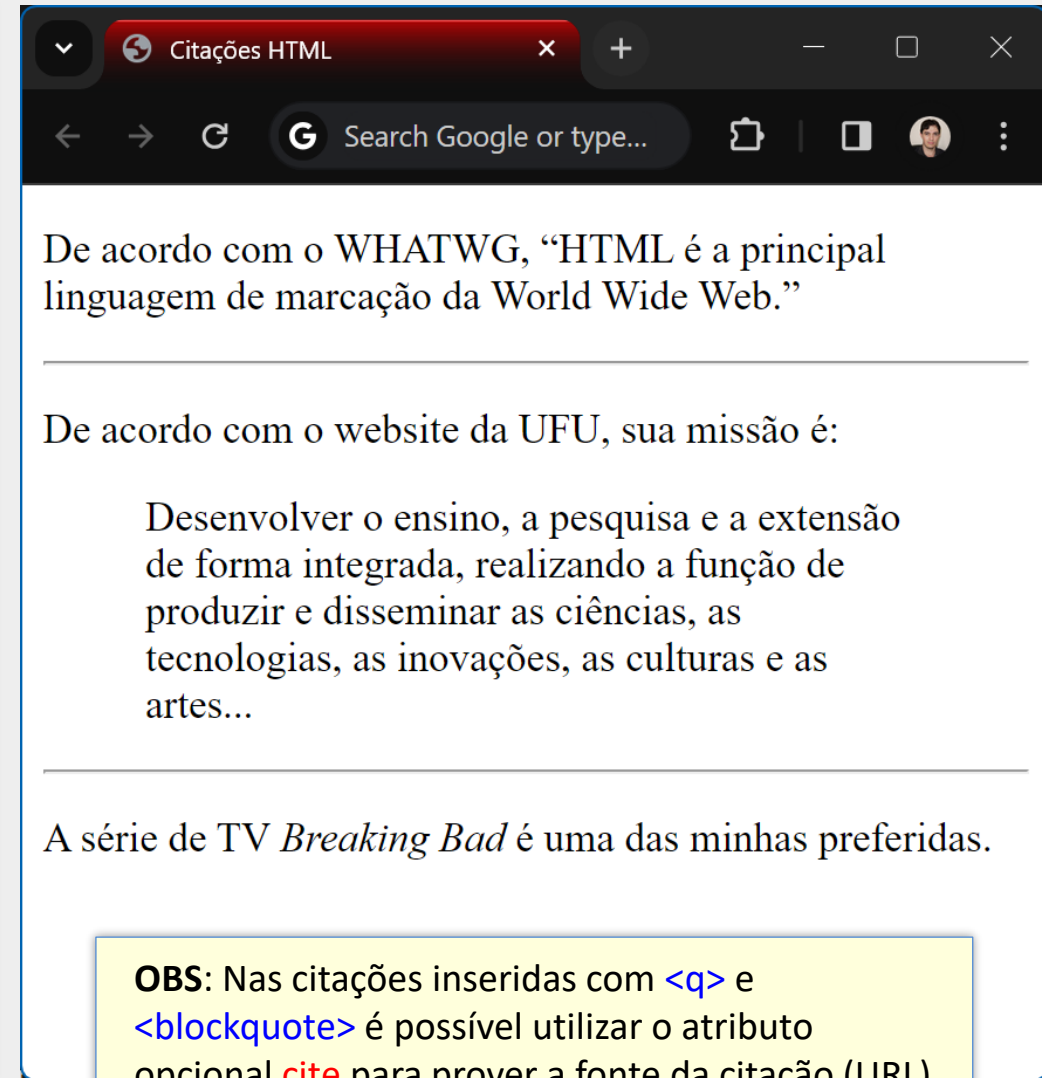
# Exemplo de Citações

```
<body>
  <p>De acordo com o WHATWG,
  | <q>HTML é a principal linguagem de
  | | marcação da World Wide Web.</q>
  </p>

  <hr>
  <p>De acordo com o website da UFU, sua missão é:</p>
  <blockquote>
  | Desenvolver o ensino, a pesquisa e a extensão
  | de forma integrada, realizando a função de
  | produzir e disseminar as ciências, as tecnologias,
  | as inovações, as culturas e as artes...
  </blockquote>

  <hr>
  <p>A série de TV <cite>Breaking Bad</cite>
  | é uma das minhas preferidas.</p>

  <!-- hr cria uma mudança temática no conteúdo -->
</body>
```



De acordo com o WHATWG, “HTML é a principal linguagem de marcação da World Wide Web.”

---

De acordo com o website da UFU, sua missão é:

Desenvolver o ensino, a pesquisa e a extensão de forma integrada, realizando a função de produzir e disseminar as ciências, as tecnologias, as inovações, as culturas e as artes...

---

A série de TV *Breaking Bad* é uma das minhas preferidas.

**OBS:** Nas citações inseridas com `<q>` e `<blockquote>` é possível utilizar o atributo opcional `cite` para prover a fonte da citação (URL)

# Elementos `<b>` e `<strong>`

## `<b>`

- Historicamente: `<b>` de **bold**
- Na HTML5: utilizado para chamar a atenção sem adicionar forte importância
- Usar apenas quando não houver elemento mais apropriado

## `<strong>`

- Define o texto como de **forte importância** ou urgência
- Situação em que o texto é lido com **entonação forte**
- Pode ser explorado por tecnologias assistivas e de acessibilidade

Por padrão, o conteúdo de `<b>` e `<strong>` aparecerão em negrito

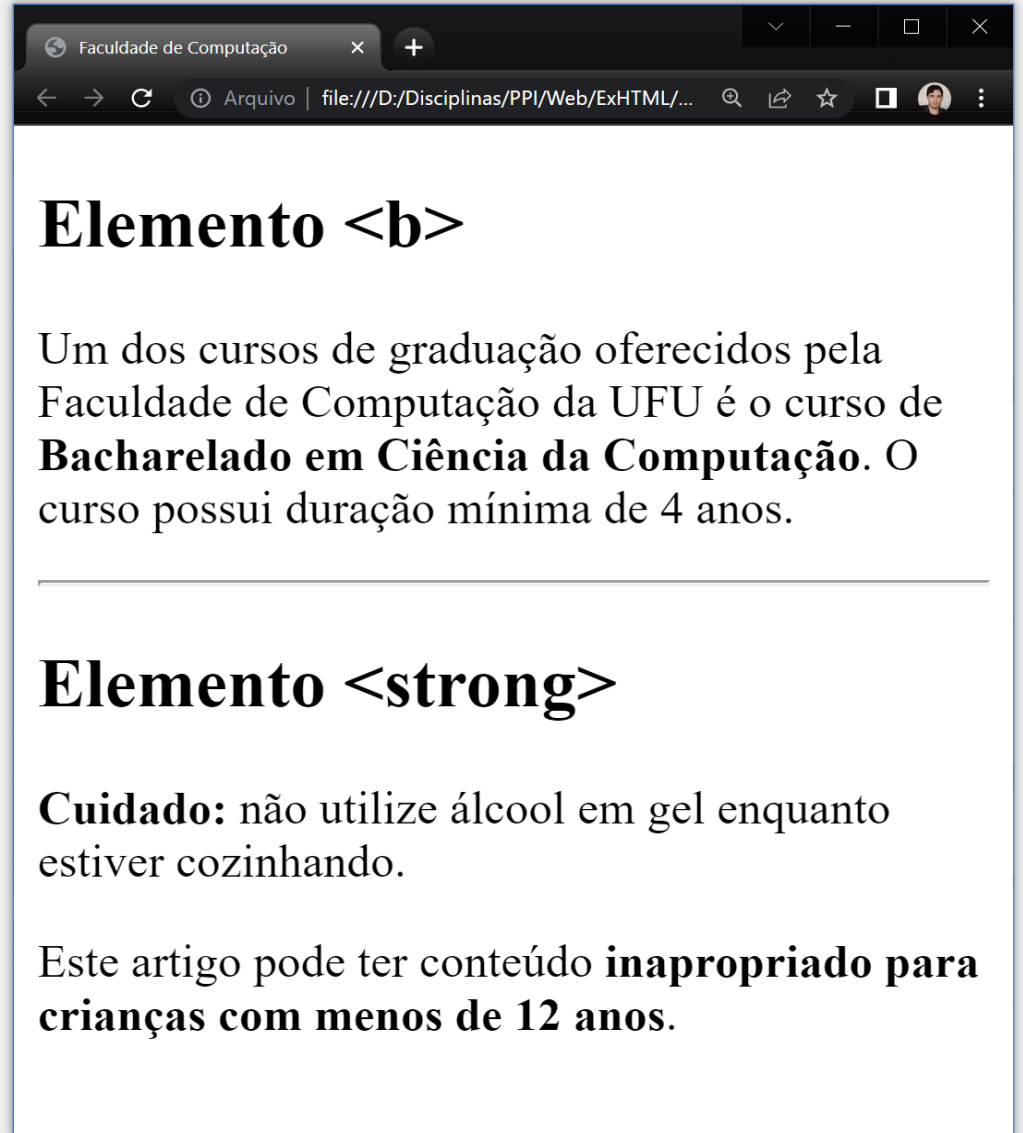
# Exemplo de <b> e <strong>

```
<body>
  <h2>Elemento &lt;b>; </h2>

  <p>Um dos cursos de graduação oferecidos pela
  Faculdade de Computação da UFU é o curso de
  <b>Bacharelado em Ciência da Computação</b>.
  O curso possui duração mínima de 4 anos.
</p>

<hr>
<h2>Elemento &lt;strong>; </h2>

<p><strong>Cuidado:</strong> não utilize
  álcool em gel enquanto estiver cozinhando.</p>
<p>Este artigo pode ter conteúdo
  <strong>inapropriado para crianças
  com menos de 12 anos</strong>.
</p>
</body>
```



**Elemento <b>**

Um dos cursos de graduação oferecidos pela Faculdade de Computação da UFU é o curso de **Bacharelado em Ciência da Computação**. O curso possui duração mínima de 4 anos.

---

**Elemento <strong>**

**Cuidado:** não utilize álcool em gel enquanto estiver cozinhando.

Este artigo pode ter conteúdo **inapropriado para crianças com menos de 12 anos**.

# Elementos `<i>` e `<em>`

## `<i>`

- Historicamente: `<i>` de **italic**
- HTML5: destaque textual para situações como:
  - palavras de outros idiomas
  - termos técnicos
  - designações taxonômicas, dentre outros.

## `<em>`

- Enfatização textual

Por padrão, o conteúdo de `<i>` e `<em>` aparecerão em itálico

# Exemplo de <i> e <em>

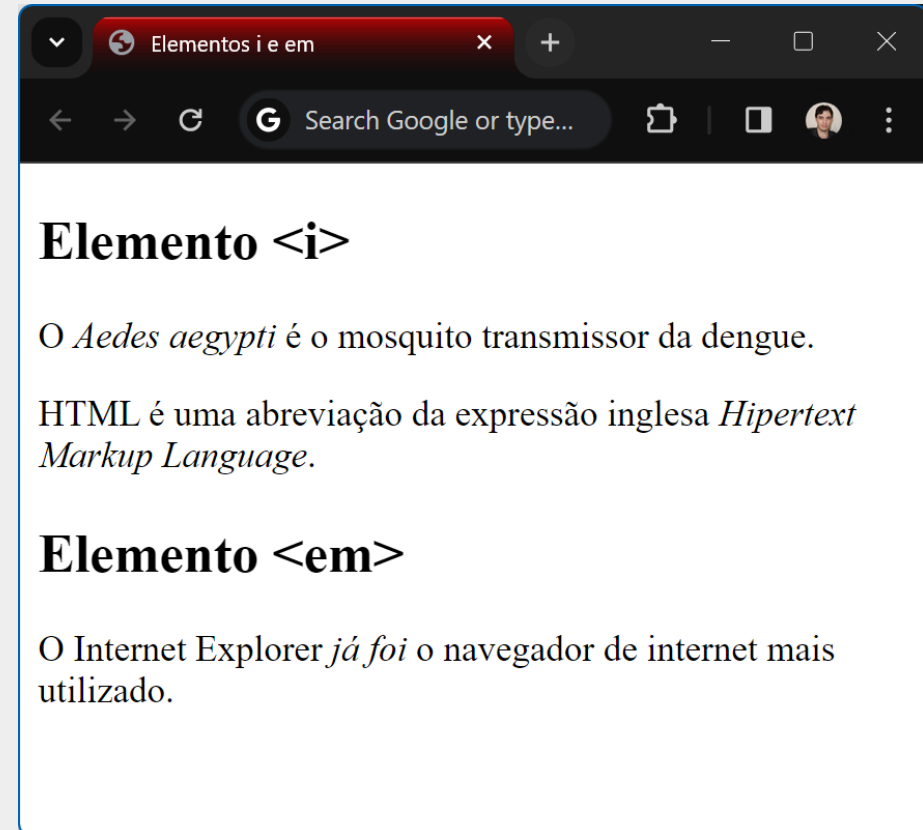
```
<h2>Elemento &lt;i&gt; </h2>

<p>O <i>Aedes aegypti</i> é o mosquito
transmissor da dengue.</p>

<p>HTML é uma abreviação da expressão inglesa
<i>Hipertext Markup Language</i>.
</p>

<h2>Elemento &lt;em&gt; </h2>

<p>O Internet Explorer <em>já foi</em> o
navegador de internet mais utilizado.</p>
```





# Listas

Lista não ordenada:

- Item
- Item
- Item
- Item
- Item
- Item
- Item
- Item

Lista ordenada:

1. Item
2. Item
3. Item
4. Item
5. Item
6. Item
7. Item
8. Item

# Listas

`<ul>`

- **unordered list**
- Cria uma lista não ordenada (com marcadores)

`<ol>`

- **ordered list**
- Cria uma lista ordenada (com numeração)

`<li>`

- **list item**
- Adiciona um novo item de lista

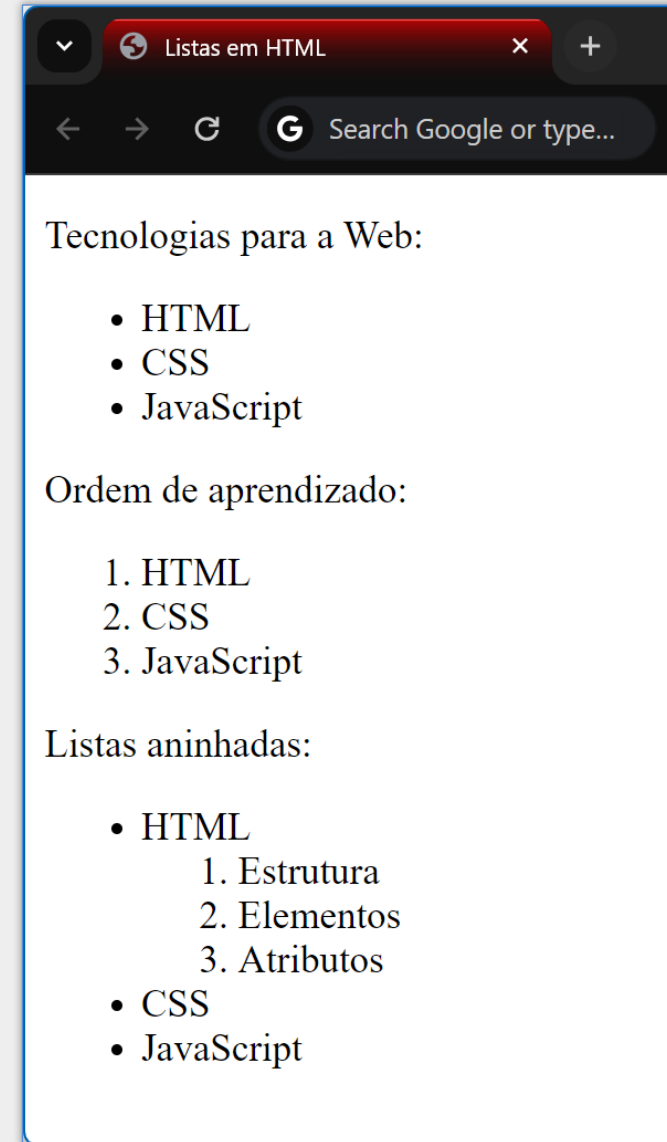
**OBS:** Listas não devem ser inseridas dentro de `<p>` `</p>`

# Exemplo de Listas

```
<p>Tecnologias para a Web:</p>
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>

<p>Ordem de aprendizado:</p>
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ol>
```

```
<p>Listas aninhadas:</p>
<ul>
  <li>HTML
    <ol>
      <li>Estrutura</li>
      <li>Elementos</li>
      <li>Atributos</li>
    </ol>
  </li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>
```



# Imagens



Universidade Federal de Uberlândia – *campus* Santa Mônica

# Inserção de Imagens com <img>

- ``
- Atributo **src**
  - Permite inserir o caminho do arquivo de imagem (obrigatório)
  - Formatos aceitos: JPEG, PNG, GIF, SVG, BMP, dentre outros.
- Atributo **alt**
  - Fornece uma descrição da imagem (obrigatório)
  - Historicamente: texto alternativo
  - Atualmente: acessibilidade (leitura de tela), busca por imagem etc.
- Atributos **width** e **height**
  - Largura e a altura de exibição da imagem, em pixels (número inteiro, sem unidade)
  - Permitem ao navegador reservar espaço adequado
  - Previnem mudanças no layout durante carregamento
- Atributo **title**
  - Permite fornecer informação adicional sobre a imagem
  - O texto aparecerá quando o usuário passar o ponteiro do mouse sobre a imagem



# Exemplo <img>

```
<body>
```

```
<p>A UFU é uma fundação pública,  
integrante da Administração  
Federal Indireta, vinculada ao  
Ministério da Educação (MEC).</p>
```

```

```

```
</body>
```



# Elementos <figure> e <figcaption>

```
<figure>  
    
  <figcaption>Legenda da Figura</figcaption>  
</figure>
```

- Imagem exibida com recuo
- Várias imagens podem compartilhar a mesma legenda
- Figuras não precisam ser necessariamente imagens
  - Ex.: Pode-se criar um gráfico com JavaScript e exibi-lo como figura
  - Ex.: Pode-se exibir um código ou algoritmo como figura

# Exemplo de Figura

```
<p>A Universidade Federal de Uberlândia (UFU) é  
uma fundação pública, integrante da Administração  
Federal Indireta, vinculada ao Ministério da  
Educação (veja o campus Santa Mônica na Figura 1)</p>  
  
<figure>  
    
  <figcaption>Figura 1 - UFU Santa Mônica</figcaption>  
</figure>  
  
<p>A Universidade Federal de Uberlândia (UFU)...</p>
```



A Universidade Federal de Uberlândia (UFU) é uma fundação pública, integrante da Administração Federal Indireta, vinculada ao Ministério da Educação (veja o campus Santa Mônica na Figura 1)



Figura 1 - UFU Santa Mônica

A Universidade Federal de Uberlândia (UFU)...



# Organização dos Arquivos – O que evitar

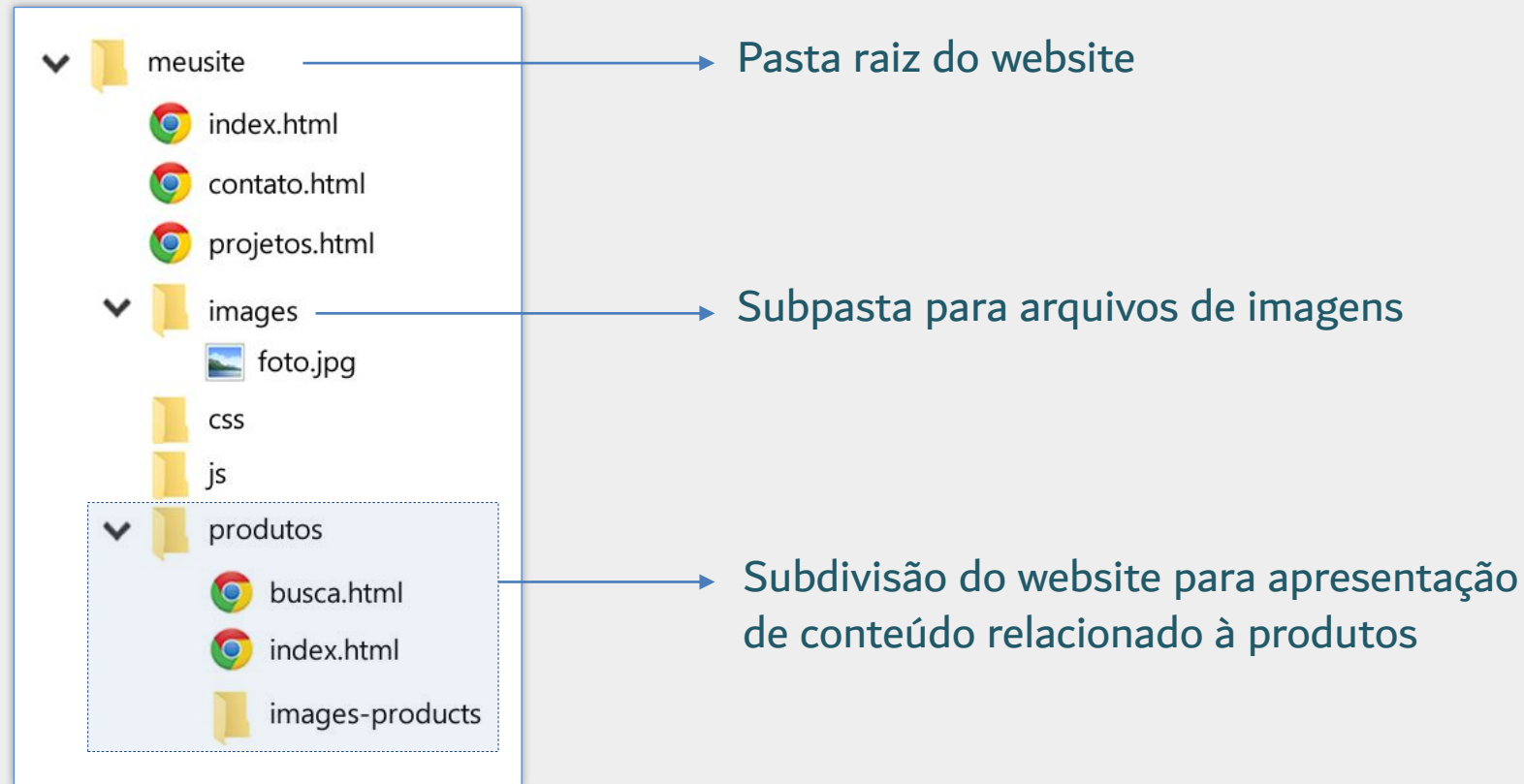
```

```

imagem na mesma pasta do arquivo HTML

- Evite referenciar imagens na mesma pasta do arquivo HTML
- O website pode conter vários arquivos, de vários tipos
  - HTML, imagens, CSS, JavaScript, PHP, etc.

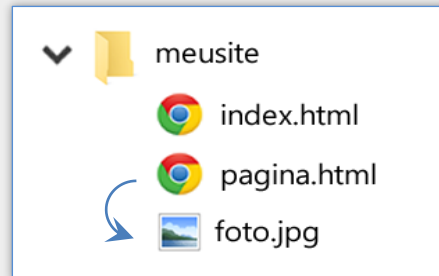
# Organização dos Arquivos - Exemplo



- Recomenda-se organizar o website em múltiplas pastas
- E referenciar os arquivos de maneira relativa
- Geralmente há uma opção no editor para abrir a pasta raiz inteira

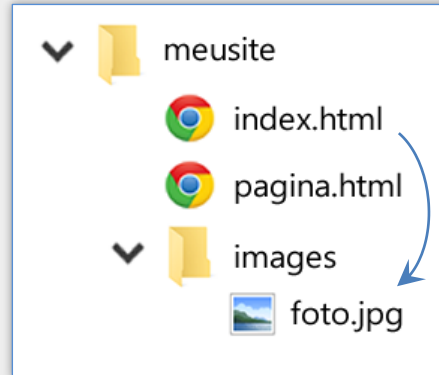
# Caminhos Relativos de Arquivos

Acesso a **foto.jpg** na mesma pasta do HTML



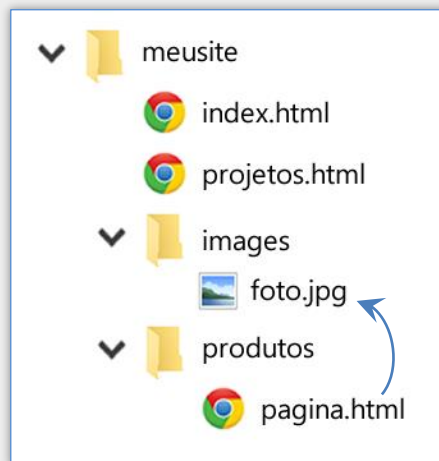
→ ``

Acesso a **foto.jpg** na subpasta **images**



→ ``

Acesso a **foto.jpg** a partir do arquivo **pagina.html** da pasta **produtos**



→ "Ponto-ponto" permite acessar a pasta superior

``

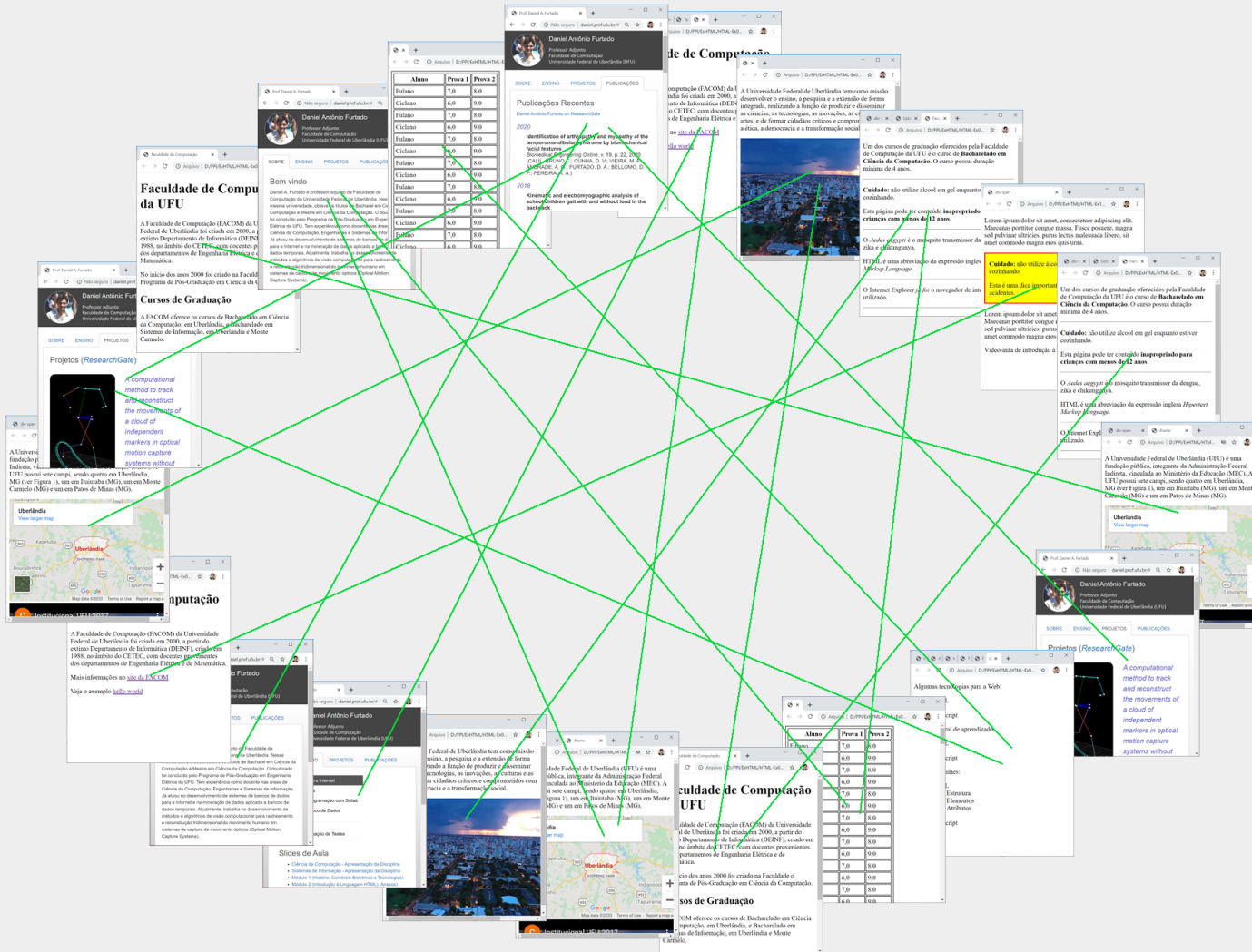
``

Iniciar o caminho com "/" permite acesso direto à pasta raiz do website, independentemente de onde se faz o acesso

# Dicas sobre Caminhos de Arquivos e Imagens

- Jamais utilize caminhos absolutos
  - ✗ `` (Não faça isso)
- Prefira letras minúsculas ou adote um padrão ao escolher os nomes
- Não utilize caracteres especiais
- Utilize imagens com tamanho e resolução adequados
  - ✗ Arquivos de alta resolução diminuídos com `width` e `height` (NÃO faça isso)
  - ✗ Download de arquivo grande desnecessariamente
- Utilize um formato de imagem apropriado
  - PNG: compactação sem perda de informação
  - JPEG: compactação com perda de informação
  - Aspectos a serem considerados: tipo de conteúdo, tamanho, transparência etc.

# Hiperlinks



# Hiperlinks

*Endereço de destino do link*

```
<a href="http://www.ufu.br" > UFU </a>
```

*Nome do link que aparecerá para o usuário*

## ■ Endereço absoluto

- Inclui nome de domínio e protocolo
- Direcionamento para outro website

## ■ Endereço relativo

- Sem nome de domínio
- Direcionamento dentro do próprio site
- Preferível sempre que possível (não vincula o site a um nome de domínio)

# Hiperlinks - Exemplo

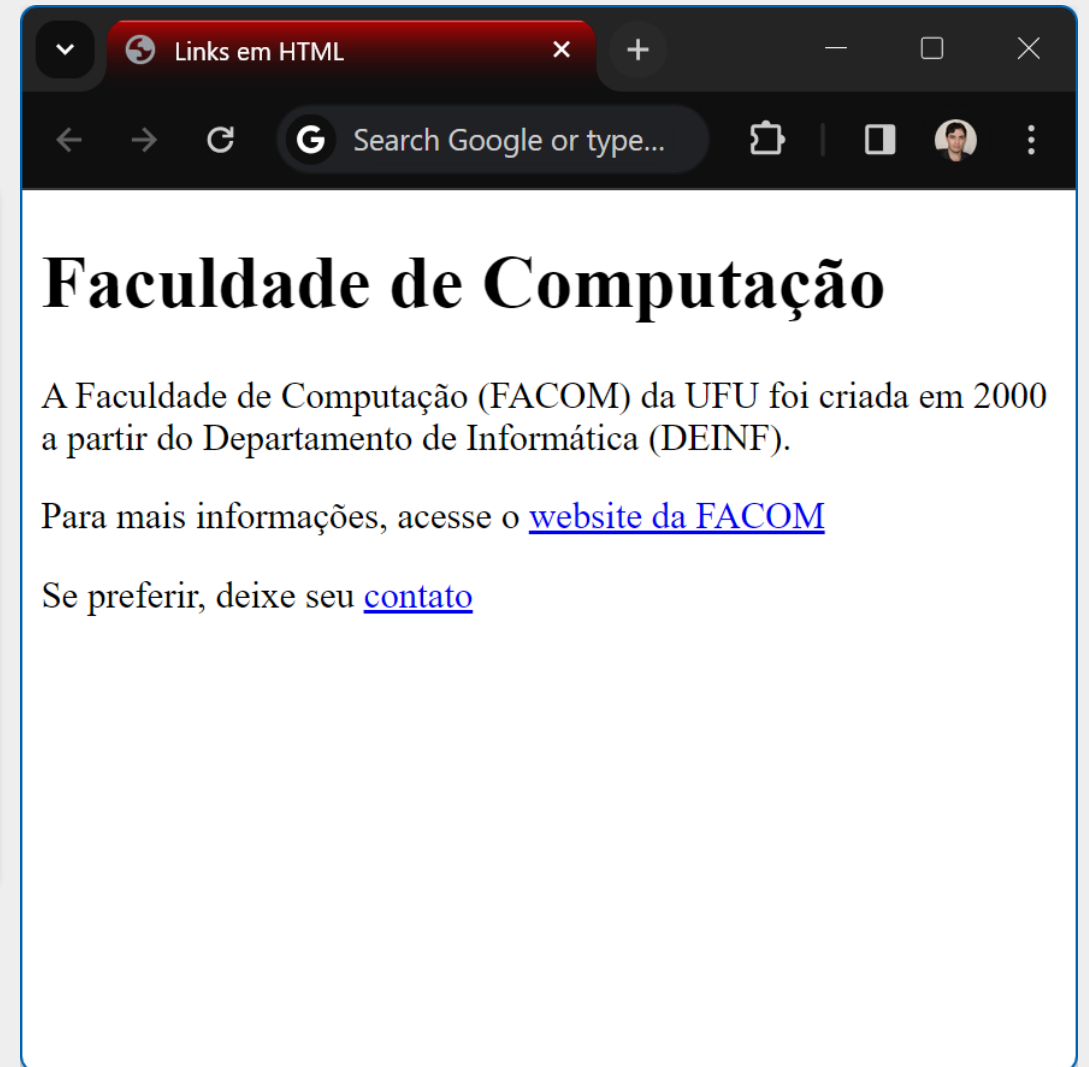
```
<body>

  <h1>Faculdade de Computação</h1>
  <p>A Faculdade de Computação (FACOM) da UFU foi criada em
    2000 a partir do Departamento de Informática (DEINF).</p>

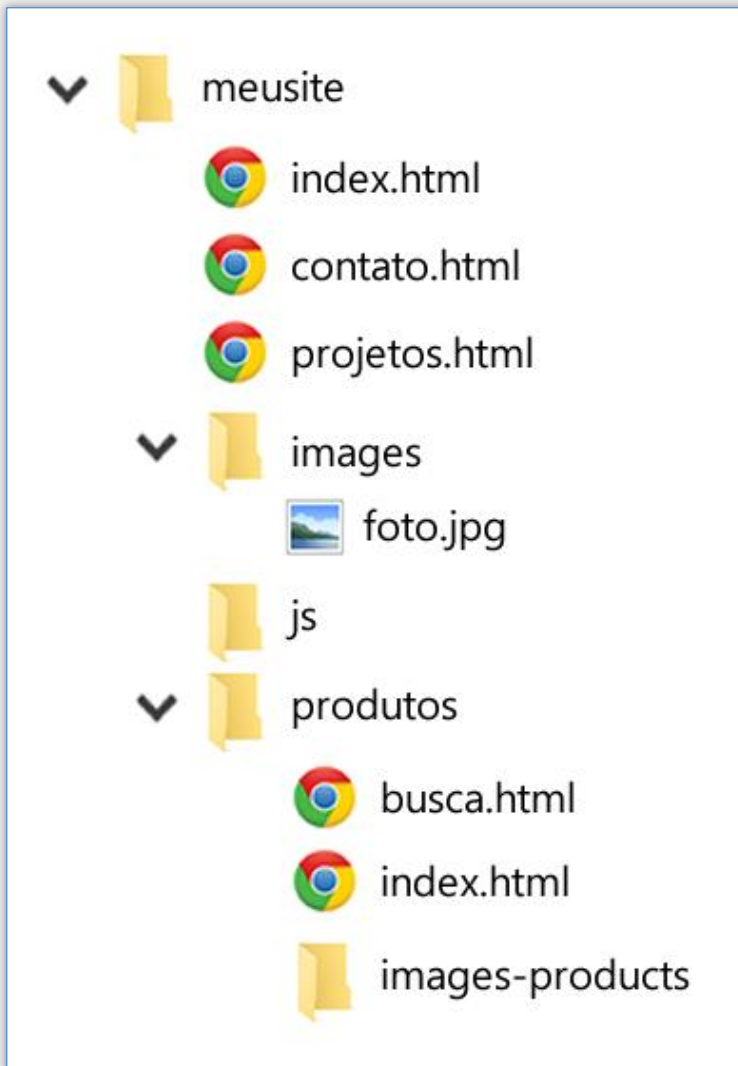
  <p>Para mais informações, acesse o
    <a href="http://www.facom.ufu.br" target="_blank">
      website da FACOM</a></p>

  <p>Se preferir,
    deixe seu <a href="contato.html">contato</a></p>

</body>
```



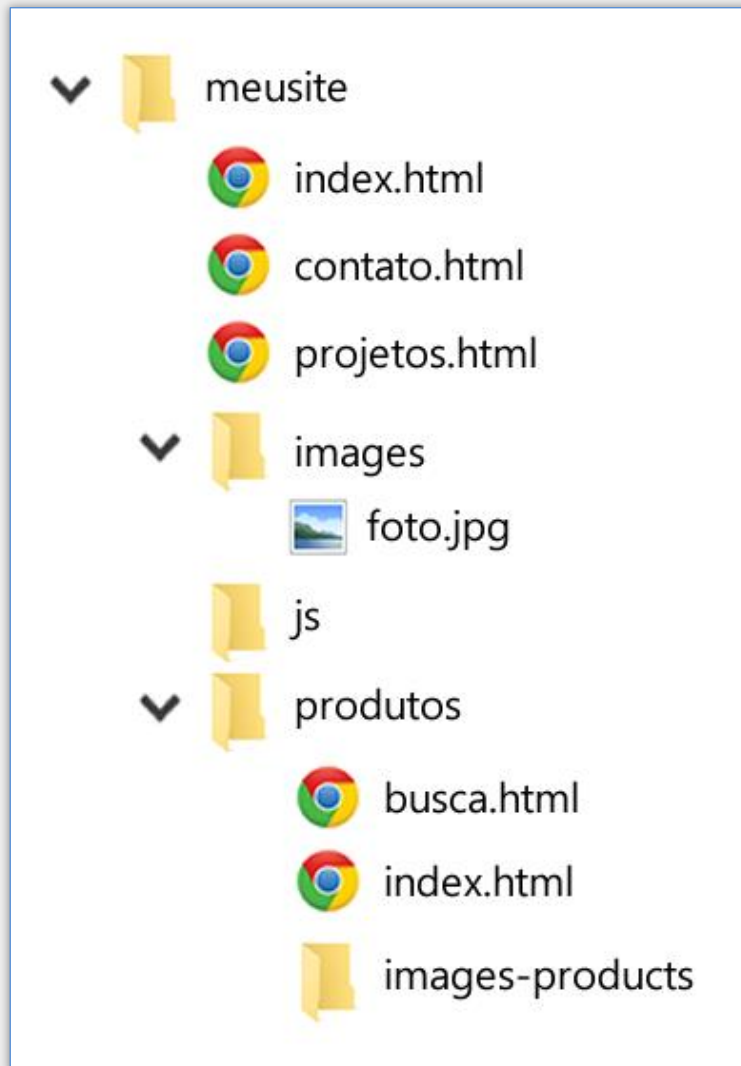
# Estrutura de Pastas e Hiperlinks Relativos



- Link de arq. HTML da pasta raiz para arq. HTML da pasta raiz
  - `<a href="contato.html">Contato</a>`
- Link de arq. HTML da pasta raiz para arq. HTML da pasta produtos
  - `<a href="produtos/busca.html">Buscar</a>`
- Link de arq. HTML da pasta produtos para arq. HTML da pasta produtos
  - `<a href="busca.html">Buscar</a>`
- Link de arq. HTML da pasta produtos para arq. HTML da pasta raiz
  - `<a href=" ../contato.html">Contato</a>` ou
  - `<a href="/contato.html">Contato</a>` (apenas no servidor)



# Estrutura de Pastas e Arquivos – Acesso Online



[www.exemplo.com](http://www.exemplo.com)

(o servidor retornará o arquivo padrão **index.html** da pasta raiz)  
(configurável)

[www.exemplo.com/contato.html](http://www.exemplo.com/contato.html)

(acesso online da página **contato.html**)

[www.exemplo.com/produtos](http://www.exemplo.com/produtos)

(o servidor retornará o arquivo **index.html** da pasta **produtos**)

# Imagem como Hiperlink

- O conteúdo do elemento `a` não precisa ser textual
- Pode-se ter outros elementos, inclusive o `img`
  - Neste caso, a imagem servirá como link

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

# Hiperlink para Fragmento do Documento

## Introdução

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. ([Ver resultados](#))

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

## O Projeto

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

...

...

## Resultados

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

...

...

```
<a href="#resultados">
```

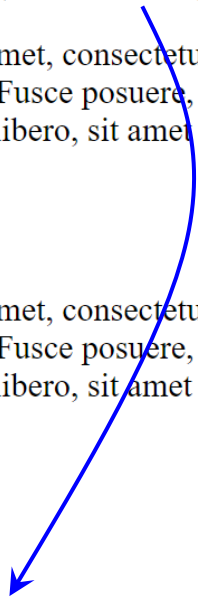
Ver resultados

```
</a>
```

...

```
<h1 id="resultados">Resultados</h1>
```

...



# Outros Hiperlinks

Link para  
enviar e-mail

```
<a href="mailto:exemplo@exemplo.com">  
    Entre em contato por e-mail  
</a>
```

Link para  
número de  
telefone

```
<a href="tel:034-9999-9999">  
    Entre em contato por telefone  
</a>
```

# Tabelas

Matrícula	Total Acertos	Tempo Total	Nota Total	Nro Testes	Iniciais
1	23	00:30:58	3.80	2	LMM
2	22	00:24:08	3.66	2	FFC
3	22	00:26:14	3.60	2	DSL
4	22	00:27:17	3.66	2	CFL
5	22	00:36:04	3.60	2	MRRA
6	21	00:16:13	3.51	2	JM
7	21	00:18:38	3.57	2	LBF
8	21	00:31:13	3.51	2	MPS
9	21	00:33:48	3.46	2	CBG
10	20	00:30:21	3.31	2	MMC

# Tabelas – Elementos Básicos

## <table>

- Criação da tabela

## <tr>

- **table row**
- Define uma **nova linha** na tabela

## <td>

- **table data**
- Define uma **célula padrão** dentro de uma linha

# Tabelas – Elementos Adicionais

## <thead>, <tbody>, <tfoot>

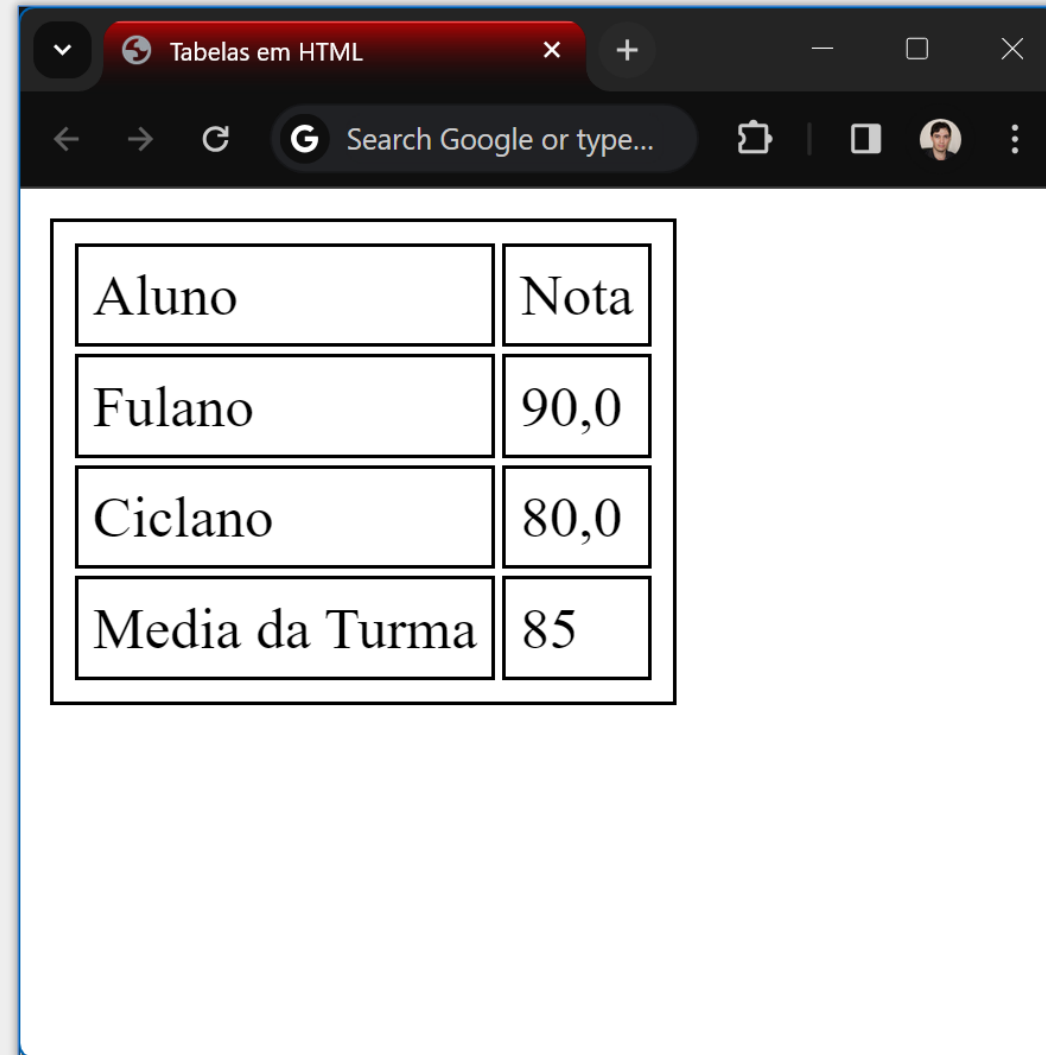
- Definem seções de cabeçalho, corpo e rodapé na tabela, respectivamente
- Comumente utilizados em tabelas longas
- Especialmente úteis quando tabelas longas são enviadas para impressão, pois cabeçalhos e rodapés são repetidos em cada página, facilitando a leitura.

## <th>

- **table head**
- Define uma célula de cabeçalho dentro de uma linha
- Por padrão o navegador exibe o conteúdo em negrito e centralizado
- Comumente utilizando dentro de <thead> e <tfoot>

# Exemplo de Tabela

```
<table>
  <tr>
    <td>Aluno</td>
    <td>Nota</td>
  </tr>
  <tr>
    <td>Fulano</td>
    <td>90,0</td>
  </tr>
  <tr>
    <td>Ciclano</td>
    <td>80,0</td>
  </tr>
  <tr>
    <td>Media da Turma</td>
    <td>85</td>
  </tr>
</table>
```



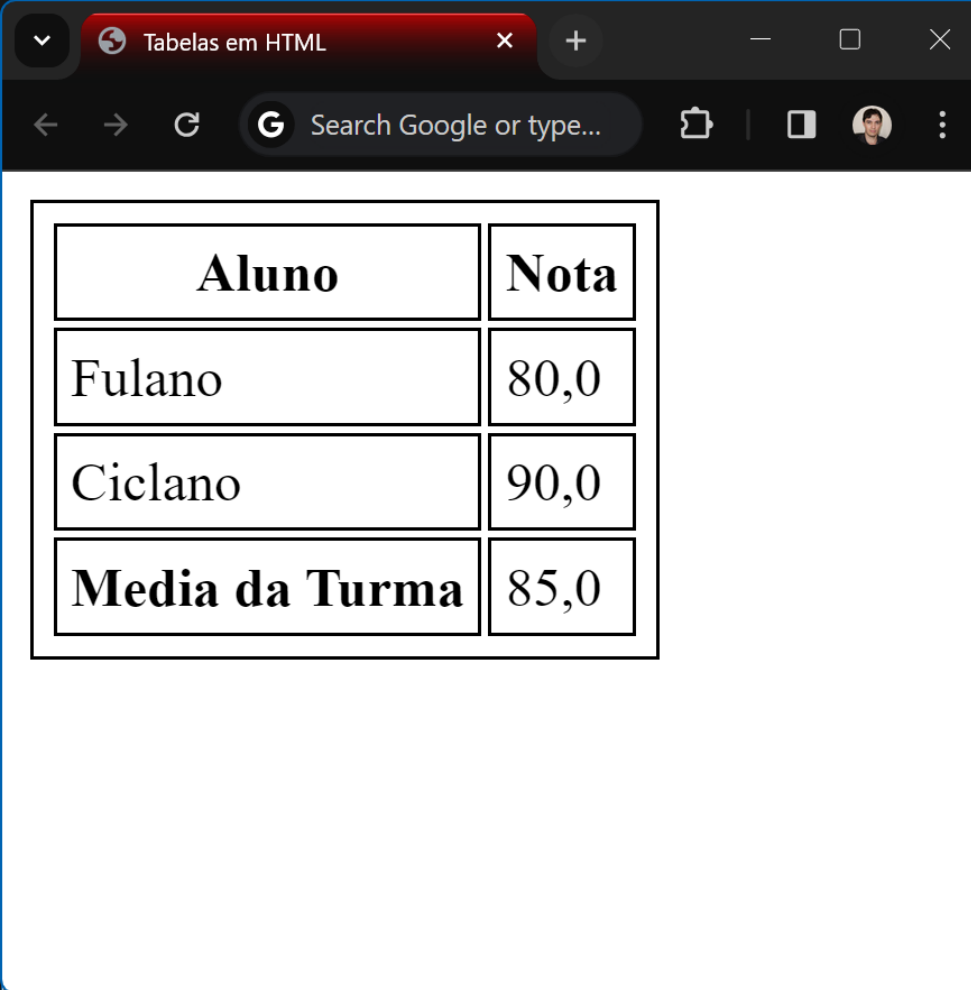
A screenshot of a web browser window titled "Tabelas em HTML". The browser's address bar shows "Search Google or type...". The main content area displays a table with the following data:

Aluno	Nota
Fulano	90,0
Ciclano	80,0
Media da Turma	85



# Exemplo de Tabela com Cabeçalho, Corpo e Rodapé

```
<table>
  <thead <!-- Cabeçalho da tabela -->
    <tr>
      <th>Aluno</th>
      <th>Nota</th>
    </tr>
  </thead>
  <tbody <!-- Corpo da tabela -->
    <tr>
      <td>Fulano</td>
      <td>80,0</td>
    </tr>
    <tr>
      <td>Ciclano</td>
      <td>90,0</td>
    </tr>
  </tbody>
  <tfoot <!-- Rodapé da tabela -->
    <tr>
      <th>Media da Turma</th>
      <td>85,0</td>
    </tr>
  </tfoot>
</table>
```



A screenshot of a web browser window titled "Tabelas em HTML". The browser's address bar shows "Search Google or type...". The main content area displays a table with the following structure:

Aluno	Nota
Fulano	80,0
Ciclano	90,0
Media da Turma	85,0

# Servidor Web

# Colocando o Website Online

- Para colocar o website online normalmente utiliza-se um **nome de domínio** e um **servidor web**
- Nome de domínio
  - Endereço do website na Web
  - Pode-se registrar um nome de domínio próprio
  - Ou utilizar um nome de domínio existente e criar um subdomínio gratuito
- Servidor Web
  - É possível hospedar o website em servidor terceirizado (gratuito ou pago)
  - Também é possível contratar um serviço de VPS (*virtual private server*)
  - Outra possibilidade é montar um servidor dedicado próprio ou contratar um serviço que o ofereça
    - Necessário instalar servidor HTTP como Apache HTTP, NGINX, Microsoft IIS etc.

# Colocando o Website Online

- Exemplos de serviços de hospedagem
  - [www.infinityfree.net](http://www.infinityfree.net)
  - [www.awardspace.com](http://www.awardspace.com)
  - [www.atspace.com](http://www.atspace.com)
  - E vários outros...

# infinityfree - Colocando o Website Online

1. Crie uma conta gratuita em [www.infinityfree.net](http://www.infinityfree.net) e faça login
2. Crie uma nova conta de hospedagem e registre um subdomínio (slide a seguir)
3. Acesse os dados para conexão por FTP
4. Envie os arquivos do website utilizando um software cliente FTP
5. Acesse o website no navegador utilizando o nome de subdomínio criado

# infinityfree – Criação de conta e subdomínio gratuito

1. Depois de logar, acesse **Accounts** → **Create Account**
2. Escolha o plano **gratuito**
3. Digite um nome para o **subdomínio** (por ex. trabalhosppi, fulanoppi etc.)
4. Escolha uma das opções para o **domínio** (por ex. infinityfreeapp.com)
5. Verifique a **disponibilidade** e vá para o próximo passo
6. Na etapa de **informações adicionais**, escolha uma opção para **Email Consent**. Não é necessário alterar os demais campos.
7. Finalize a criação clicando em **Create Account**.

# infinityfree – Acessando dados para conexão por FTP

1. Depois de criar a conta, clique novamente em **Accounts**
2. Clique no link relativo à conta criada
3. No painel à esquerda, clique em **FTP Details**
4. Observe as informações apresentadas (FTP Username, FTP Password e FTP Hostname). Elas serão necessárias para se conectar ao servidor do infinityfree e enviar os arquivos das páginas web.

FTP Details for if0\_37051744

FTP USERNAME	FTP PASSWORD	FTP HOSTNAME
if0_37051744	***** Show/Hide	ftpupload.net
FTP PORT (OPTIONAL)		
21		

# Envio dos Arquivos por FTP com o WinSCP

The screenshot shows the WinSCP Login dialog box. The 'File protocol' is set to 'FTP', 'Encryption' is 'No encryption', 'Host name' is 'ftpload.net', 'Port number' is '21', and 'User name' is 'epiz\_123456'. The 'Password' field is empty. There are 'Save' and 'Advanced...' buttons. At the bottom, there are 'Tools', 'Manage', 'Login', 'Close', and 'Help' buttons. A checkbox 'Show Login dialog on startup and when the last session is closed' is checked.

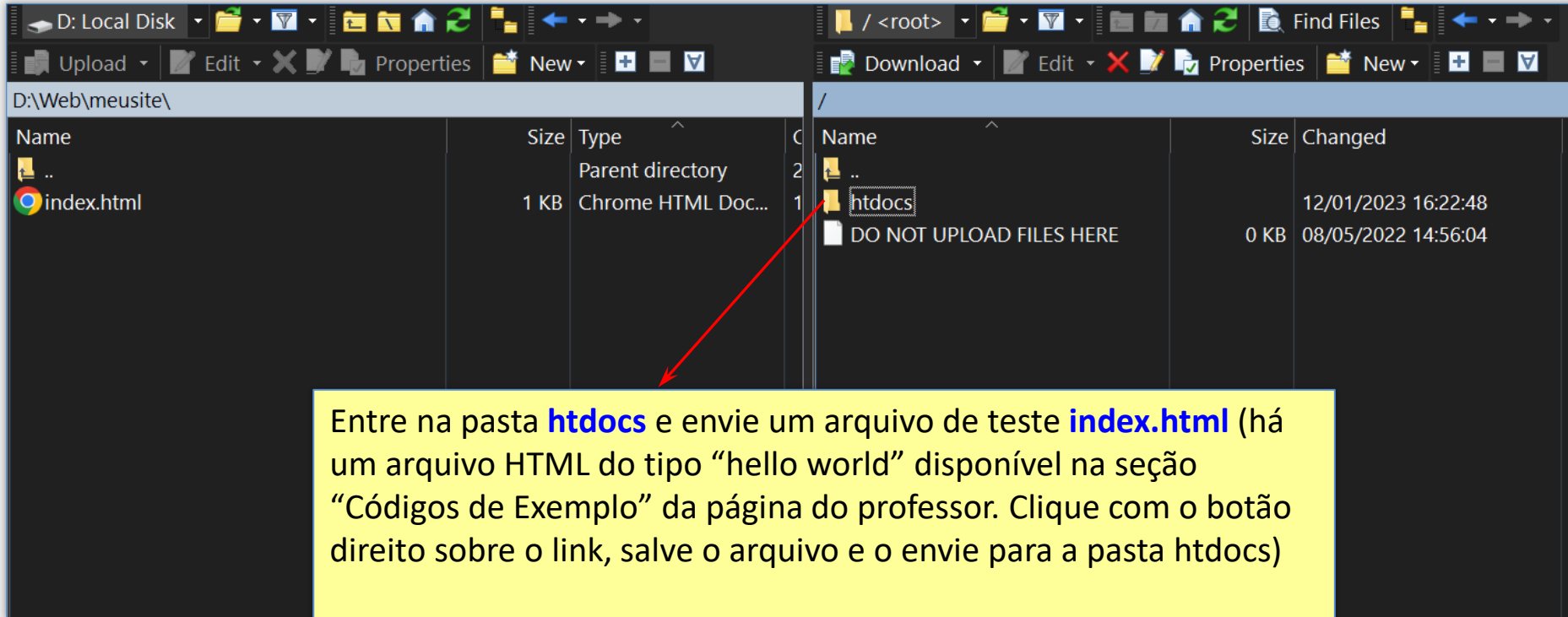
Annotations:

- Altere o protocolo de **SFTP** para **FTP**
- Coloque o FTP USERNAME criado pelo *infinityfree*
- Deixe a porta padrão (21)
- A senha poderá ser informada depois
- Coloque aqui o FTP hostname disponibilizado pelo *infinityfree*.  
Exemplo: *ftpload.net*

Para colocar o website online, precisamos de um programa cliente FTP. Usuários do Windows podem baixar e instalar gratuitamente o **WinSCP**. Para usuários de outros sistemas, há alternativas como o **FileZilla**.



# Envio dos Arquivos com o WinSCP



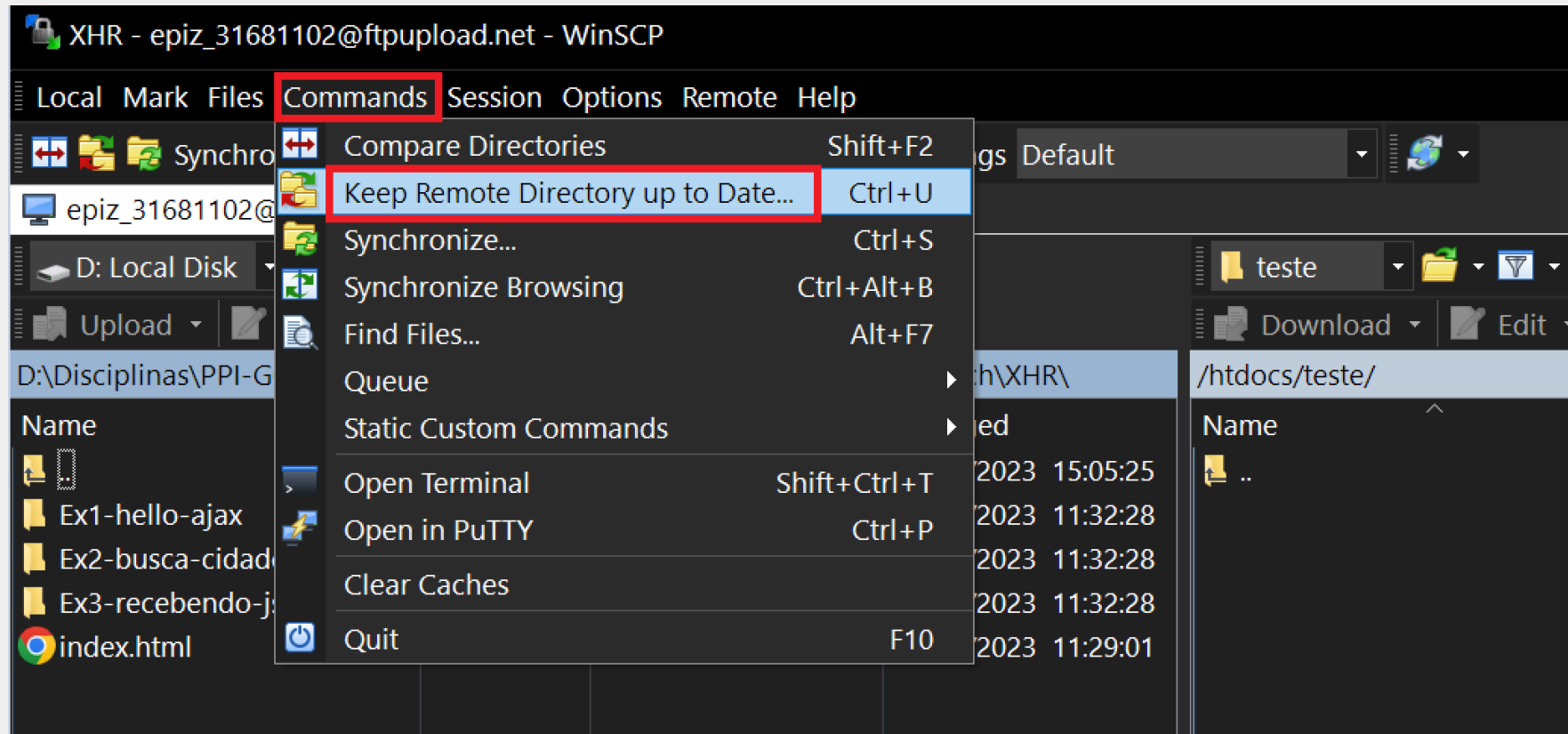
Entre na pasta **htdocs** e envie um arquivo de teste **index.html** (há um arquivo HTML do tipo “hello world” disponível na seção “Códigos de Exemplo” da página do professor. Clique com o botão direito sobre o link, salve o arquivo e o envie para a pasta htdocs)

Em seguida basta acessar o website digitando o endereço de cadastro no navegador (seusite.infinityfreeapp.com)

# Enviando Arquivos Automaticamente

- Durante o desenvolvimento do website é possível abrir a pasta raiz localmente na ferramenta de desenvolvimento (Visual Studio Code) e configurar o WinSCP para enviar os arquivos automaticamente quando eles forem salvos localmente
- Passos:
  1. Depois de se conectar com o WinSCP, escolha a pasta do website no servidor;
  2. Escolha a pasta do website localmente (lado esquerdo da janela). Essa pasta será monitorada pelo WinSCP;
  3. Acesse o menu **Commands** → **Keep Remote Directory up to Date** → **Start**
  4. Minimize a janela do WinSCP. A partir de agora, quando salvar um arquivo localmente no Visual Studio Code, ele será enviado automaticamente para o servidor


# Enviando Arquivos Automaticamente



# Validação da Página HTML

- Exibição adequada no navegador **não é garantia de código correto**
  - O navegador pode ocultar erros e inconsistências
- Documento fora da especificação
  - Apresentação **inconsistente e imprevisível** nos navegadores
- Serviço oferecido pelo W3C para validação de código HTML5
  - [validator.w3.org](http://validator.w3.org)

# Validação da Página HTML



## Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

[Validate by URI](#)   **Validate by File Upload**   [Validate by Direct Input](#)

### Validate by File Upload

Upload a document for validation:

File:

[▶ More Options](#)

**Note:** file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our [information page](#) on the W3C QA Website.

**Atenção:** Ao validar uma página hospedada no **infinityfree**, prefira a validação por **upload de arquivo** ao invés de validar diretamente pelo endereço online, pois o serviço de hospedagem do infinityfree pode alterar o conteúdo HTML que é enviado ao navegador, inserindo erros que não existem de fato no arquivo HTML original.

# Visual Studio Code – Atalhos Essenciais

- Aplicar a formatação e indentação do código
  - **SHIFT + ALT + F**
- Mover um bloco de código para cima ou para baixo
  - Depois de selecionar, utilize **ALT + ↑** ou **ALT + ↓**
- Copiar um bloco de código para cima ou para baixo
  - Depois de selecionar, utilize **ALT + SHIFT + ↑** ou **ALT + SHIFT + ↓**
- Edição de múltiplas linhas simultaneamente
  - Segure a tecla **ALT** e clique nos pontos a serem editados
  - Ou segure **CTRL + ALT** e utilize as setas
- Comentar e descomentar múltiplas linhas
  - Depois de selecionar, utilize **CTRL + ;**
- Abrir a Command Palette e buscar por arquivos e comandos
  - Depois de selecionar, utilize **CTRL + P**

# Visual Studio Code – Extensões

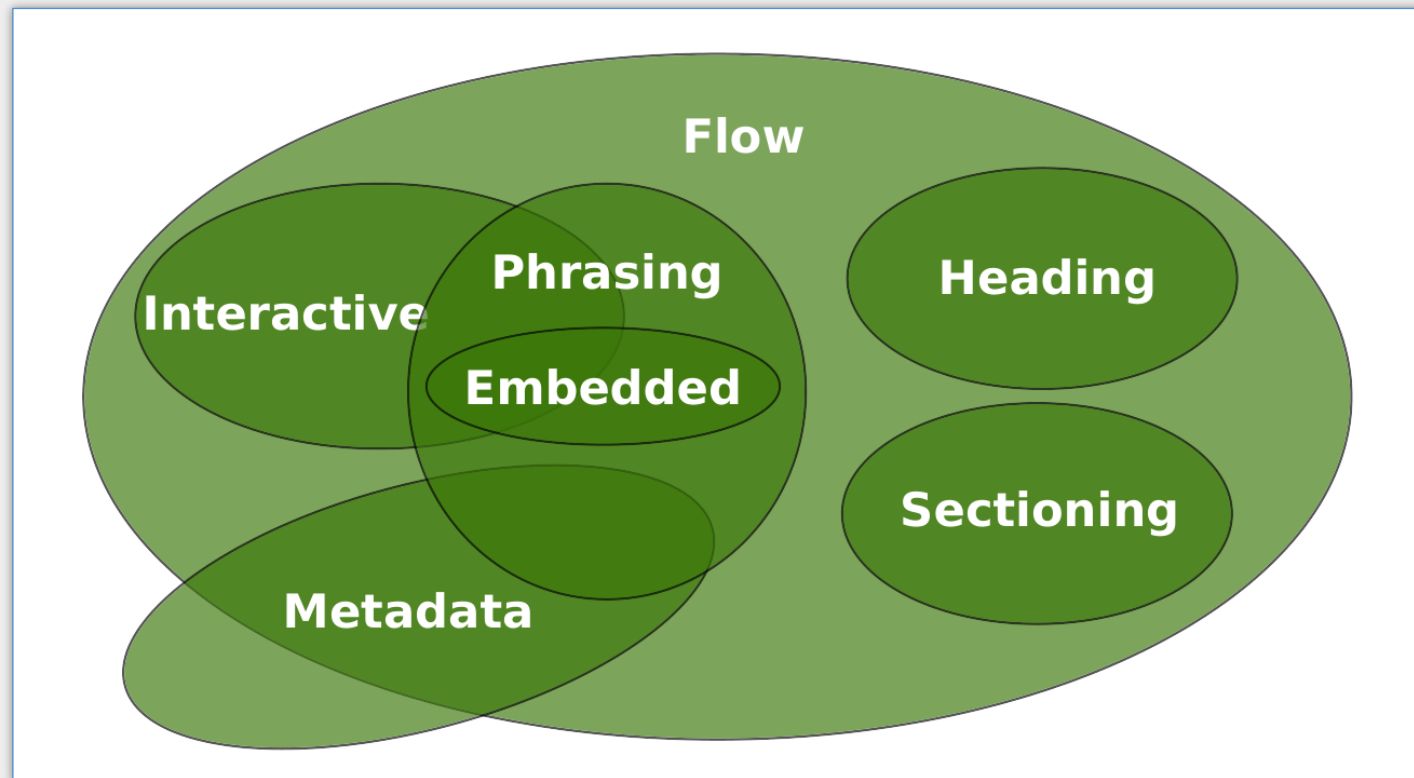
- Visualizar página HTML em tempo real
  - Extensão [Live Preview](#) da Microsoft
- Autocomplete para JavaScript
  - Extensão [IntelliCode](#) da Microsoft
- Renomear tag de fechamento junto com tag de abertura
  - Extensão [Auto Rename Tag](#)
- Autocomplete e **indentação** para código PHP e outros recursos
  - Extensão [PHP Intelephense](#)

## Parte 2

Categorias de elementos, conteúdo embutido, semântica e layout, validação e formulários



# Categorias de Conteúdo – HTML5



Na HTML5 os elementos com características similares são agrupados em categorias, chamadas **categorias de conteúdo**, ilustradas na imagem acima. Um elemento pode pertencer a zero, uma ou várias categorias. Entretanto, há também uma classificação mais simples, criada antes da HTML5, que classifica os elementos em **elementos de bloco** e **elementos de linha**.

# Elementos de Bloco e Elementos de Linha

## ■ Elementos de bloco

- Começam e terminam com uma **quebra de linha**
- Ocupam, por padrão, **toda a largura disponível**
- Exemplos: `<p>`, `<h1>`, `<table>`, `<ol>`, `<ul>`, `<li>`, `<div>`

## ■ Elementos de linha

- Não começam com quebra de linha
- Ocupam **somente a largura necessária para sua exibição**
- Exemplos: `<img>`, `<a>`, `<span>`, `<b>`, `<em>`

# Elementos de Bloco e Elementos de Linha

```
<head>
  <meta charset="UTF-8">
  <title>Elemento de Bloco vs Elemento de Linha</title>
  <style>h1, h2, a, img { background-color: yellow; }</style>
</head>
<body>
  <h1>Faculdade de Computação</h1>
  <h2>Cursos de Graduação</h2>
  <h2>Cursos de Pós-Graduação</h2>

  <p>Informações em
    <a href="http://www.facom.ufu.br">FACOM</a> e
    <a href="http://www.ufu.br">UFU</a>
  </p>

  <!-- Comportamento padrão. Pode ser alterado com CSS -->
</body>
```



`<h1>`, `<h2>` e `<p>` são exemplos de elementos de bloco. Observe que eles ocupam toda a largura disponível e começam com uma quebra de linha. Por outro lado, `<a>` e `<img>` são elementos de linha. Portanto, podem ser exibidos um na frente do outro e ocupam apenas o espaço necessário para apresentação de seu conteúdo.

# Elementos `<div>` e `<span>`

## `<div>`

- **Container genérico de nível de bloco**
- Utilizado para agrupar outros elementos
- Sem semântica (não agrega significado)
- Usar apenas quando não houver elemento semântico mais apropriado (como `<header>`, `<footer>`, `<main>`, `<article>`, `<nav>`, `<aside>` etc.)

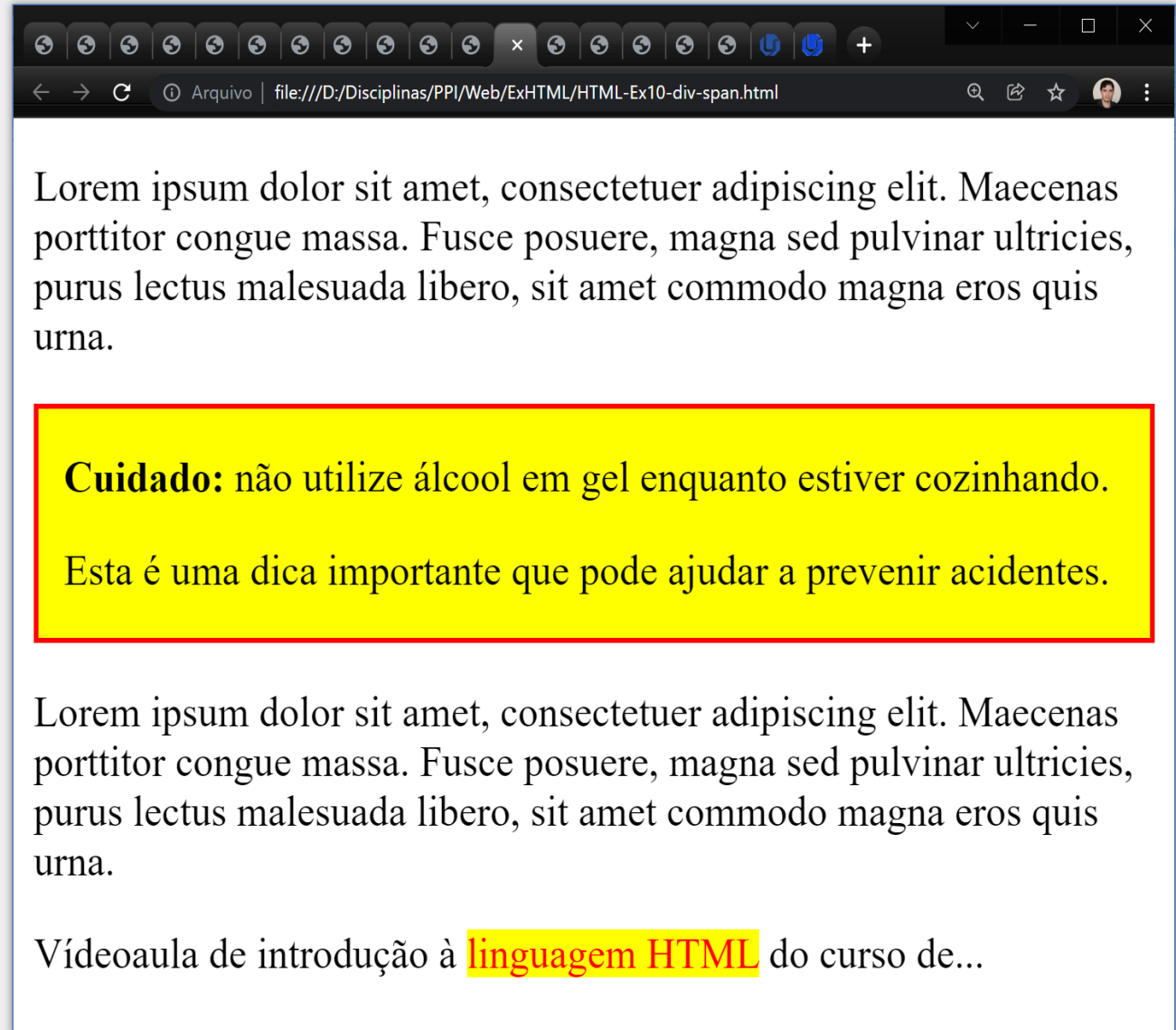
## `<span>`

- **Container genérico de nível de linha**
- Sem semântica (não agrega significado)
- Usar apenas quando não houver elemento semântico mais apropriado

# Elementos <div> e <span>

```
<head>
  <style>
    .warning {
      border: 2px solid red; background-color:
        yellow; padding-left: 10px; }
    .destacado {
      background-color: yellow;
      color: red; }
  </style>
</head>
<body>
  <p>Lorem ipsum dolor sit amet, consectetur elit...</p>
  <div class="warning">
    <p><strong>Cuidado:</strong> não utilize álcool
    em gel enquanto estiver cozinhando.</p>
    <p>Esta é uma dica importante que pode
    ajudar a prevenir acidentes.</p>
  </div>

  <p>Lorem ipsum dolor sit amet, consectetur elit...</p>
  <p>Videoaula de introdução à
  <span class="destacado">linguagem HTML</span>
  do curso de...</p>
</body>
```



Arquivo | file:///D:/Disciplinas/PPI/Web/ExHTML/HTML-Ex10-div-span.html

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

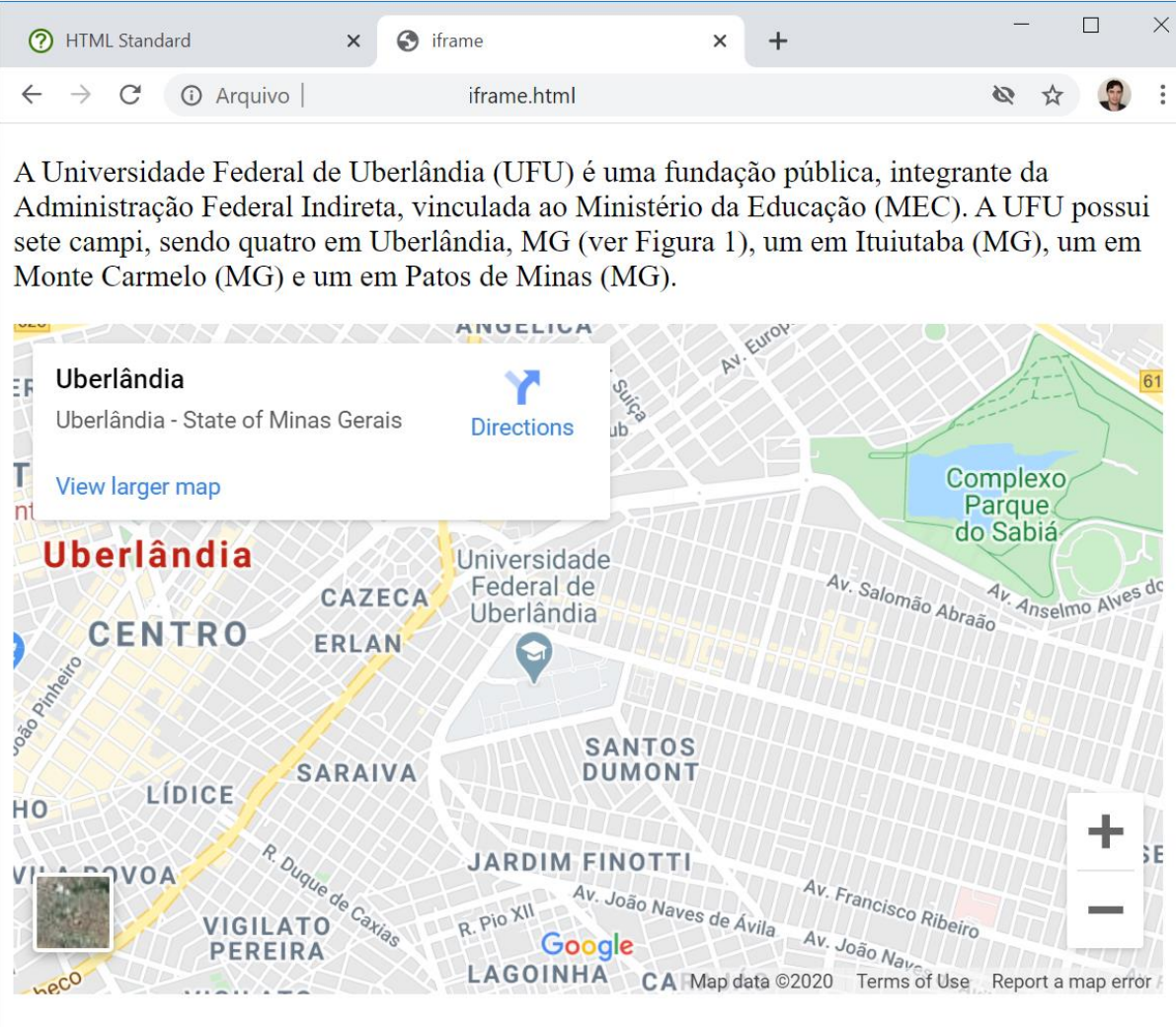
**Cuidado:** não utilize álcool em gel enquanto estiver cozinhando.

Esta é uma dica importante que pode ajudar a prevenir acidentes.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.


Vídeoaula de introdução à **linguagem HTML** do curso de...

# Conteúdo Embutido



The screenshot shows a web browser window with two tabs: "HTML Standard" and "iframe". The address bar displays "Arquivo | iframe.html". The main content area contains a text block and a Google Map. The text block reads: "A Universidade Federal de Uberlândia (UFU) é uma fundação pública, integrante da Administração Federal Indireta, vinculada ao Ministério da Educação (MEC). A UFU possui sete campi, sendo quatro em Uberlândia, MG (ver Figura 1), um em Ituiutaba (MG), um em Monte Carmelo (MG) e um em Patos de Minas (MG).". Below the text is a Google Map of Uberlândia, MG, with a location pin on the "Universidade Federal de Uberlândia". A search box in the top-left corner of the map displays "Uberlândia" and "Uberlândia - State of Minas Gerais", with a "Directions" button and a "View larger map" link. The map shows various neighborhoods like CENTRO, CAZECA, ERLAN, SARAIVA, SANTOS DUMONT, JARDIM FINOTTI, and LAGOINHA. A "Complexo Parque do Sabiá" is also visible. The Google logo and "Map data ©2020" are at the bottom of the map.

A Universidade Federal de Uberlândia (UFU) é uma fundação pública, integrante da Administração Federal Indireta, vinculada ao Ministério da Educação (MEC). A UFU possui sete campi, sendo quatro em Uberlândia, MG (ver Figura 1), um em Ituiutaba (MG), um em Monte Carmelo (MG) e um em Patos de Minas (MG).



# Conteúdo Embutido - <iframe>

- O elemento <iframe> permite embutir o conteúdo de outra página HTML dentro da página atual;
- O conteúdo embutido pode ser, por exemplo, outra página do próprio website, um vídeo do YouTube ou um mapa do Google Maps;
- Atributos comuns no <iframe>
  - **src** indica o endereço da página a ser embutida
  - **width** e **height** largura e altura do 'quadro' de exibição da página embutida
  - **allow** política de permissões relacionada ao conteúdo embutido
  - **allowfullscreen** indica se o conteúdo poderá ser exibido em tela cheia

# Exemplo de Conteúdo Embutido – Código

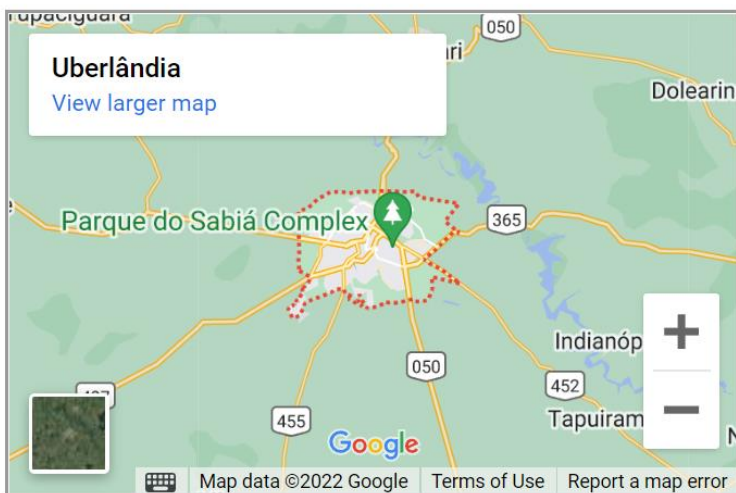
```
<iframe width="380" height="250"  
  src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d241549  
  allowfullscreen>  
</iframe>  
  
<iframe width="380" height="250"  
  src="https://www.youtube.com/embed/IgVUSu_NCPE"  
  allow="accelerometer; autoplay; gyroscope; picture-in-picture"  
  allowfullscreen>  
</iframe>  
  
<iframe width="380" height="250"  
  src="HTML-Ex02-paragrafos.html">  
</iframe>
```

Exemplo: <https://youtu.be/Jtl3kNUsWx4?t=3749>



# Exemplo de Conteúdo Embutido – Resultado

A Universidade Federal de Uberlândia (UFU) é uma fundação pública, integrante da Administração Federal Indireta, vinculada ao Ministério da Educação (MEC). A UFU possui sete campi: quatro em Uberlândia, MG (ver Figura 1), um em Ituiutaba (MG), um em Monte Carmelo (MG) e um em Patos de Minas (MG).



## Faculdade de Computação da UFU

A Faculdade de Computação (FACOM) da Universidade Federal de Uberlândia foi criada em 2000 a partir do Departamento de Informática (DEINF), criado em 1988, no âmbito do CETEC, com docentes provenientes dos departamentos de Eng. Elétrica e Matemática.

No início dos anos 2000, foi criada na Faculdade o

A Universidade Federal de Uberlândia tem como missão desenvolver o ensino, a pesquisa e a extensão de forma integrada, realizando a função de produzir e disseminar as ciências, as inovações, as culturas e as artes, e de formar cidadãos críticos e comprometidos com a ética, a democracia e a transformação social.

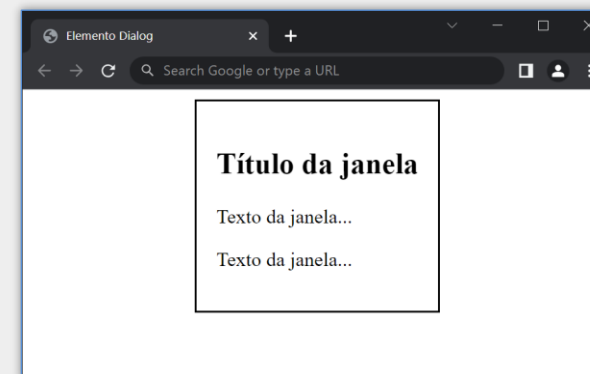
# Exibição de Arquivos de Vídeo

```
<video width="800" height="600" controls>  
  <source src="video1.mp4" type="video/mp4">  
</video>
```

# Elemento Dialog

- Permite criar uma caixa de diálogo que pode ser usada como **janela modal**
- Suportado pelos principais navegadores modernos
- A caixa de diálogo pode ser facilmente aberta e fechada com JavaScript utilizando os métodos `show()`, `showModal()` e `close()`
- O atributo `open` permite criar a janela já aberta, porém com utilidade prática limitada
- Pode ser estilizada com CSS

```
<dialog id="meuModal" open>
  <h2>Título da janela</h2>
  <p>Texto da janela...</p>
  <p>Texto da janela...</p>
</dialog>
```



# Atributos Globais

- São atributos que podem ser utilizados em todos os elementos HTML
- Podem não ter efeito prático em alguns elementos
- Os atributos **id** e **title** apresentados anteriormente são atributos globais

# Exemplos de Atributos Globais

## id

- Cria um nome de identificação único para o elemento
- Não pode conter espaços

## title

- Adiciona informação extra ao elemento
- A informação é exibida ao passar o ponteiro do mouse sobre o elemento

## lang

- Especifica a linguagem/país do conteúdo do elemento

# Exemplos de Atributos Globais

## data-\*

- Atributo de dado customizável
- Permite ao desenvolvedor criar seu próprio atributo como **data-matricula**

## contenteditable

- Torna o conteúdo do elemento editável (até mesmo parágrafos, títulos, etc.)

## class

- Permite referenciar uma classe CSS

## style

- Permite inserir código CSS embutido no elemento

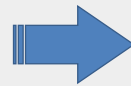
# Entidades HTML

- Há alguns caracteres que não podem ser colocados diretamente no documento HTML, como parte do conteúdo
- Por exemplo, os caracteres `<` e `>` são reservados para compor as tags
- A forma correta de inserir esses caracteres é utilizando **entidades HTML**

# Entidades HTML



```
<html>
<body>
  <p>
    1 < 2
  </p>
</body>
</html>
```



**Error** Bad character `<` after `<`. Probable cause: Unescaped `<`. Try escaping it as `&lt;`.

At line 11, column 9

```
dy>↵ ↵ <p>1 <2</p>↵↵↵</body>
```

Mensagem de **erro** apresentada pelo [validator.w3.org](http://validator.w3.org) ao utilizar o caractere `<` como parte do conteúdo



```
<html>
<body>
  <p>
    1 &lt; 2
  </p>
</body>
</html>
```



**Document checking completed. No errors or warnings to show.**

Used the HTML parser.

Total execution time 3 milliseconds.

Mensagem de **sucesso** apresentada pelo [validator.w3.org](http://validator.w3.org) quando a entidade HTML adequada é utilizada



# Exemplos de Entidades HTML

Caractere	Descrição	Nome da Entidade	Nº da Entidade
<	Menor que	&lt;	&#60;
>	Maior que	&gt;	&#62;
&	'e' comercial	&amp;	&#38;
"	Aspas duplas	&quot;	&#34;
'	Aspas simples	&apos;	&#39;
	<i>non-breaking space</i>	&nbsp;	&#160;

- 80 km/h → `<p> 80&nbsp;km/h </p>`
- Nomes de entidades são sensíveis a maiúsculas e minúsculas

# Elementos Semânticos de Estrutura e Layout



Prof. Daniel A. Furtado



[INÍCIO](#) [ENSINO](#) [PROJETOS](#) [PUBLICAÇÕES](#)

## Programação para Internet

A disciplina tem como objetivo capacitar o aluno para o desenvolvimento de aplicações Web utilizando as tecnologias de base, com foco no desenvolvimento do front-end e na programação direta do back-end, incluindo acesso a banco de dados.

Os objetivos específicos incluem: 1) discutir o funcionamento de sistemas Web e os protocolos envolvidos; 2) discutir o paradigma da programação para a Web e 3) desenvolver interfaces gráficas para a Web; 4) desenvolver websites dinâmicos e interativos através da programação direta do back-end e 5) utilizar conceitos e tecnologias para acesso a banco de dados em sistemas Web.

### **Sistema de Avaliação**

Serão aplicadas três avaliações práticas, um projeto de implementação e vários testes de aula. As avaliações práticas devem ser realizadas em horário de aula, sob supervisão do professor. O projeto de implementação deverá ser apresentado pela equipe no final do semestre letivo, conforme cronograma disponibilizado pelo professor.

## Sistemas de Banco de Dados

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magnam delectus impedit at molestias commodi autem odit maxime necessitatibus nam, accusantium velit natus, minus veritatis. Ex repellendus earum totam similique porro. Magnam delectus impedit at molestias commodi autem odit maxime necessitatibus nam, accusantium velit natus, minus veritatis.

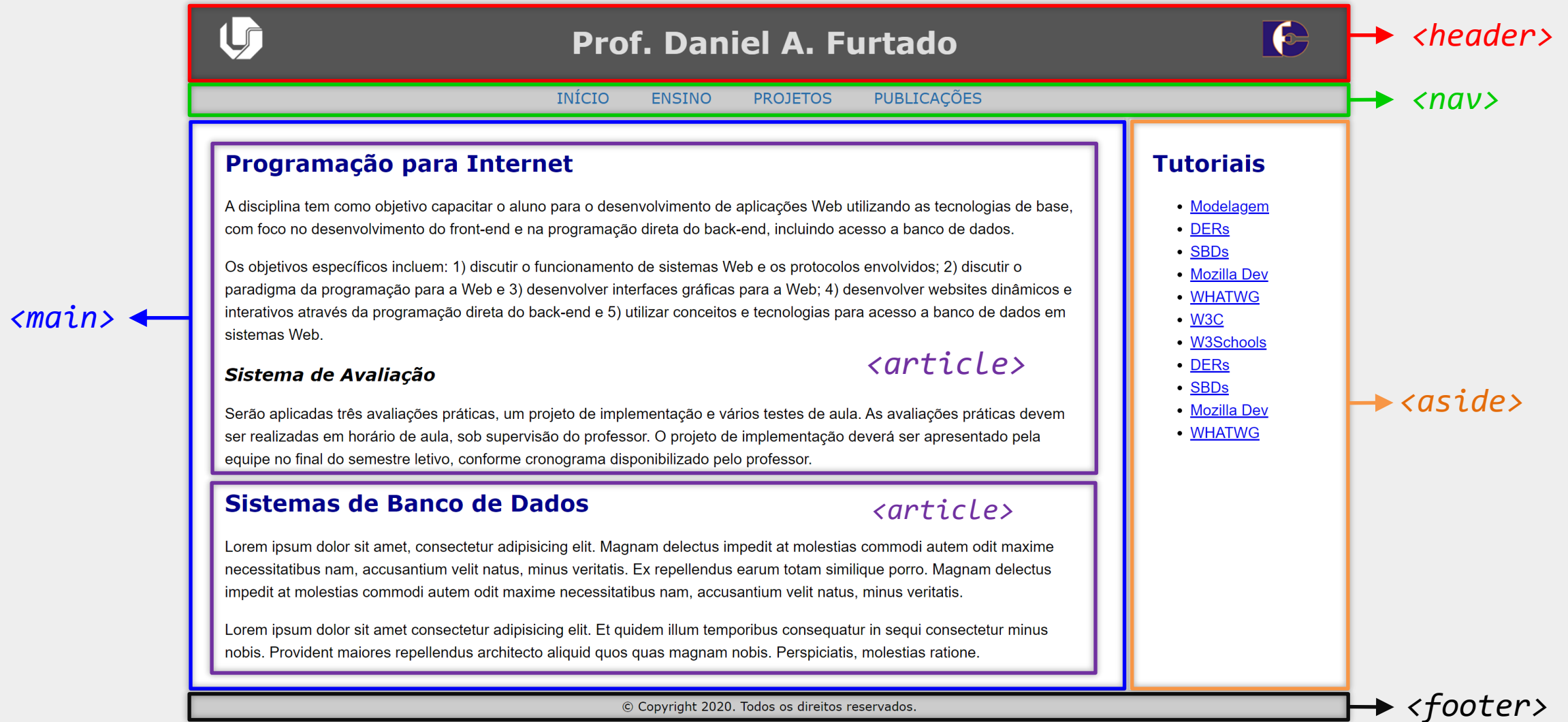
Lorem ipsum dolor sit amet consectetur adipiscing elit. Et quidem illum temporibus consequatur in sequi consectetur minus nobis. Provident maiores repellendus architecto aliquid quos quas magnam nobis. Perspiciatis, molestias ratione.

## Tutoriais

- [Modelagem](#)
- [DERs](#)
- [SBDs](#)
- [Mozilla Dev](#)
- [WHATWG](#)
- [W3C](#)
- [W3Schools](#)
- [DERs](#)
- [SBDs](#)
- [Mozilla Dev](#)
- [WHATWG](#)

© Copyright 2020. Todos os direitos reservados.

# Elementos Semânticos de Estrutura e Layout



<https://youtu.be/JtI3kNUsWx4?t=4437>

# Elementos Semânticos de Estrutura e Layout - Exemplo

```
<body>
  <!-- Cabeçalho principal, compartilhado por todas as páginas do website -->
  <header>
    
    <h1>Prof. Daniel A. Furtado</h1>
  </header>
  <!-- Menu de navegação principal, compartilhado pelas páginas do website -->
  <nav>
    <ul>
      <li><a href="#">ENSINO</a></li>
      <li><a href="#">PROJETOS</a></li>
    </ul>
  </nav>
  <!-- Conteúdo principal da página -->
  <main>
    <h2>Programação para Internet</h2>
    <p>A disciplina tem como objetivo capacitar o aluno...</p>
  </main>
  <!-- Conteúdo indiretamente relacionado ao conteúdo principal -->
  <aside>
    <p>Conteúdo da barra lateral...</p>
  </aside>
  <!-- Rodapé principal, compartilhado por todas as páginas do website -->
  <footer>
    <p>© Copyright 2024. Todos os direitos reservados.</p>
  </footer>
</body>
```

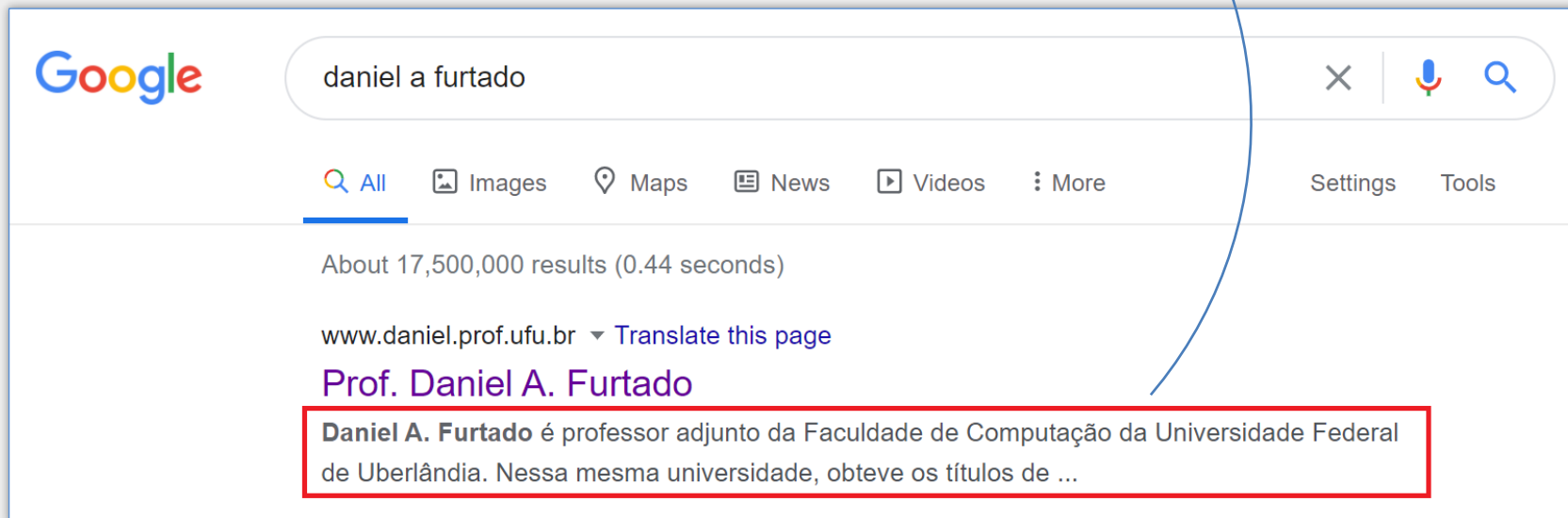
# Elemento <section>

- Elemento de seccionamento de conteúdo
- Pode ser usado, por exemplo, para organizar um artigo longo em subseções
- Recomenda-se inserir um título dentro da seção (com <h2>, <h3>, etc.)

```
...  
<article>  
  <section>  
    ...  
  </section>  
  
  <section>  
    ...  
  </section>  
</article>  
...
```

# Descrição da Página com o Elemento <meta>

```
<meta name="description" content="Daniel A. Furtado é professor...">
```



The screenshot shows a Google search for "daniel a furtado". The search bar contains the text "daniel a furtado". Below the search bar, there are navigation options: "All", "Images", "Maps", "News", "Videos", and "More". To the right, there are "Settings" and "Tools" links. The search results show "About 17,500,000 results (0.44 seconds)". The first result is from "www.daniel.prof.ufu.br" with a "Translate this page" link. The title of the result is "Prof. Daniel A. Furtado". The description, which is highlighted with a red box, reads: "Daniel A. Furtado é professor adjunto da Faculdade de Computação da Universidade Federal de Uberlândia. Nessa mesma universidade, obteve os títulos de ...". A blue arrow points from the description text in the search result to the corresponding content attribute in the meta tag code shown above.

# Outros tipos de Metadados

```
<meta property="og:image" content="URL-da-imagem">  
<meta property="og:title" content="título">  
<meta property="og:description" content="descrição">
```

Exemplo de informações de metadados utilizando o [Open Graph](#) do Facebook. Quando um link para a página for compartilhado na rede social, ele vai aparecer juntamente com um [título](#), uma [descrição](#) e uma [imagem](#), conforme foram informados nessas meta tags.

# Formulários



# Formulário Simples em HTML

Produto:

# Formulário Simples em HTML

Indica o **script no servidor** que receberá e processará os dados do formulário

Os principais métodos para envio de formulário são:  
**GET**: dados enviados pela própria URL (*GET request*)  
**POST**: dados enviados no corpo da requisição (*POST request*)

```
<form action="buscaProduto.php" method="get">
```

O **label** é vinculado ao campo pelos atributos **for/id**

Recomenda-se agrupar campo e rótulo em um container genérico (`div`) ou colocá-los dentro de um parágrafo (`p`)

```
<div>
```

```
<label for="produto">Produto:</label>
```

Clique no rótulo foca o campo  
Leitores de tela leem rótulo

```
<input type="text" id="produto" name="produto">
```

```
</div>
```

Envia o formulário

O dado é submetido com este nome de identificação. Permite resgatar o dado posteriormente no servidor. Não precisa ser igual ao **id**.

```
<button>Buscar</button>
```

```
</form>
```

Produto:

Buscar

# Formulário Simples em HTML

```
<form action="cadastra.php" method="post">
  <div>
    <label for="produto">Produto:</label>
    <input type="text" id="produto" name="prodNome">
  </div>
  <div>
    <label for="descricao">Descrição:</label>
    <input type="text" id="descricao" name="prodDesc">
  </div>
  <div>
    <label for="marca">Marca:</label>
    <input type="text" id="marca" name="prodMarca">
  </div>

  <button>Enviar</button>
</form>
```



Produto:

Descrição:

Marca:

# Outra Forma de Vincular os Rótulos aos Campos

- Outra maneira de associar corretamente os rótulos aos campos é inserindo o elemento `<input>` como conteúdo do elemento `<label>`
- Neste caso não é necessário utilizar o atributo `for` juntamente com o `id`
- Porém o atributo `name` continua sendo fundamental

```
<form action="cadastra.php" method="post">  
  <p><label>Produto: <input type="text" name="prodNome"></label></p>  
  <p><label>Descrição: <input type="text" name="prodDesc"></label></p>  
  <p><label>Marca: <input type="text" name="prodMarca"></label></p>  
  
  <button>Enviar</button>  
</form>
```

# Não Utilize `<br>` para Separar os Campos

- `<br>`'s não devem ser utilizados para separar campos de formulário, conforme detalhado pelo WHATWG na especificação da HTML (<https://html.spec.whatwg.org/multipage/text-level-semantics.html#the-br-element>);
- Ao invés de quebrar a linha com o `<br>`, utilize um container de bloco (div) ou um elemento `<p>` conforme apresentado anteriormente

Não faça isso!

```
<form action="cadastra.php" method="post">  
  <label>Produto: <input type="text" name="prodNome"></label>  
  <br>  
  <label>Descrição: <input type="text" name="prodDesc"></label>  
  <br>  
  <label>Marca: <input type="text" name="prodMarca"></label>  
</form>
```

# Campo Textual Simples vs Campo para Senha

```
<input type="text">
```

Fulano da Silva

Campo textual simples

```
<input type="password">
```

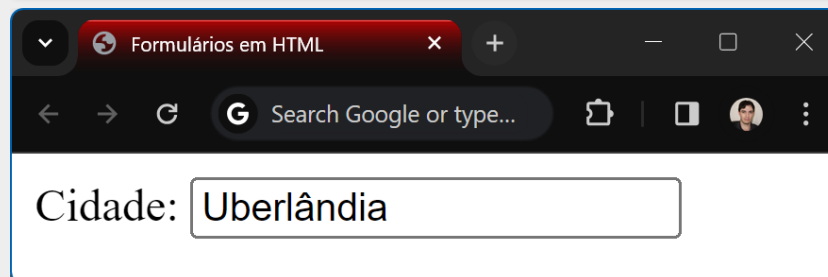
.....

Campo para senhas

# Atributo **value** em Campos Textuais

- O atributo **value** em campos textuais pode ser utilizado para fornecer um valor inicial, já preenchido, para o campo;
- O usuário poderá manter o valor sugerido ou alterá-lo;
- Não confundir com o atributo **placeholder**

```
<div>  
  <label for="cidade">Cidade:</label>  
  <input type="text" id="cidade" name="cidade" value="Uberlândia">  
</div>
```



# Métodos de Envio de Formulário

## GET

- Os dados do formulário são codificados pelo navegador na forma de uma string e adicionados no final da URL para envio ao servidor
- Ex.: `cadastra.php?produto=monitor&descricao=fullhd`
- Geralmente utilizado em formulários pequenos, sem dados sensíveis (senhas, etc.)
  - Ex.: formulário com palavras de busca

## POST

- Os dados são enviados no corpo da requisição HTTP (não aparece na URL)
- Permite upload de arquivos
- Mais adequado para formulários longos e/ou com informações sensíveis
- Mais adequado quando os dados serão adicionados a uma base de dados



# Observações Sobre Botões

## `<button>` Enviar `</button>`

- Um elemento `button` sem o atributo `type` cria um botão do tipo `submit`
- Dentro de um `<form>`, enviará o formulário
- Equivalente à `<button type="submit" ...>`
- Um elemento `button` não precisa ter apenas texto como conteúdo
  - Pode-se ter, por exemplo, um `<img>`

## `<button type="button">` Nome do botão `</button>`

- Botão para uso geral
- Não envia um formulário automaticamente. Pode ser útil quando o formulário precisa ser enviado de outras formas (utilizando Ajax, por ex.) ou quando é necessário ter um botão dentro do formulário para desempenhar outra função.

# Observações Sobre Botões

`<input type="submit" value="Enviar">`

- Botão do tipo `submit` alternativo ao `<button>`
- Elemento sem conteúdo (não permite o uso de um `<img>` como conteúdo)
- O nome do botão é definido pelo atributo `value`

`<input type="reset" value="Limpar">`

- Botão do tipo `reset`
- Dentro de um `<form>`, retorna os campos para os valores iniciais

# Observações Sobre Botões e Links

- Botão está associado a uma ação
- Link está associado a um direcionamento
- Não utilize botões como links
- Não utilize outros elementos (<div>, <span> etc.) como botões
  - Se necessário, estilize o <button> com CSS

# <form> - Atributo **enctype**

- Indica a forma de codificação dos dados do formulário ao serem enviados
- Deve ser usado apenas para **method="post"**
- Pode ser necessário quando o formulário contém arquivos (**<input type="file">**)
- Valores possíveis:
  - "application/x-www-form-urlencoded"    valor padrão
  - "multipart/form-data"    necessário quando há arquivos no form.
  - "text/plain"    dados são enviados como texto simples

<https://youtu.be/Rnfw924oYtE?t=1065>

# <form> - Atributo **autocomplete**

- Indica se os elementos `<input>` podem ser autocompletados
- `<form autocomplete="on">`
  - O navegador pode autocompletar os campos
- `<form autocomplete="off">`
  - O navegador não pode autocompletar os campos
  - Não aparecerão valores anteriores nos campos
- **autocomplete** também pode ser utilizado em campos individualmente

# Campo radio - `<input type="radio">`

Escolha o Estado Civil

Solteiro  
 Casado  
 Divorciado

- Permite a seleção de um **único item dentre vários itens**
- Utiliza-se um elemento `<input type="radio">` para cada campo e um elemento `<label>` para nomear o respectivo campo
- Utilize o atributo `value` para indicar o valor a ser enviado caso o item seja marcado
- O atributo `checked` pré-seleciona o item
- Campos do mesmo grupo devem compartilhar o mesmo valor do atributo `name`
- Recomenda-se agrupar campo e rótulo em um container
- Campos do mesmo grupo são geralmente agrupados com o elemento `<fieldset>`

# Campo radio - Exemplo

Escolha o Estado Civil

Solteiro  
 Casado  
 Divorciado

```
<fieldset>
  <legend>Escolha o Estado Civil</legend>
  <div>
    <input type="radio" id="solteiro" name="estadocivil" value="solteiro">
    <label for="solteiro">Solteiro</label>
  </div>
  <div>
    <input type="radio" id="casado" name="estadocivil" value="casado">
    <label for="casado">Casado</label>
  </div>
  <div>
    <input type="radio" id="divorciado" name="estadocivil" value="divorciado">
    <label for="divorciado">Divorciado</label>
  </div>
</fieldset>
```

Possuem o mesmo valor para o atributo **name**.  
Os rótulos são inseridos com o elemento **<label>**

# Campo checkbox - `<input type="checkbox">`

Esportes de Interesse

- Natação
- Ciclismo
- Futebol
- Hipismo

- Permite a seleção de **múltiplos itens**
- O atributo **value** deve conter o valor do item
- O atributo **checked** pré-seleciona o item



# Campo checkbox - Exemplo

```
<fieldset>
  <legend>Esportes de Interesse</legend>
  <div>
    <input type="checkbox" id="natação" name="esportes[]" value="natação" checked>
    <label for="natação">Natação</label>
  </div>
  <div>
    <input type="checkbox" id="ciclismo" name="esportes[]" value="ciclismo">
    <label for="ciclismo">Ciclismo</label>
  </div>
  <div>
    <input type="checkbox" id="futebol" name="esportes[]" value="futebol">
    <label for="futebol">Futebol</label>
  </div>
</fieldset>
```

Observe que neste exemplo os três campos possuem o mesmo valor para o atributo `name`: `"esportes[]"`. Os colchetes no final do nome permitem que os itens selecionados pelo usuário sejam resgatados posteriormente no servidor na forma de um *array* de valores (utilizando PHP, por exemplo).

# Campos para Data e Hora

`<input type="date" ...>`

Data de Nascimento: 07/10/2020

outubro de 2020							↑	↓
D	S	T	Q	Q	S	S		
27	28	29	30	1	2	3		
4	5	6	7	8	9	10		
11	12	13	14	15	16	17		
18	19	20	21	22	23	24		
25	26	27	28	29	30	31		
1	2	3	4	5	6	7		

Hoje

`<input type="time" ...>`

Hora: 13:40

13	37
14	38
15	39
16	40
17	41
18	42
19	43

`<input type="datetime-local" ...>`

09/10/2020 09:50

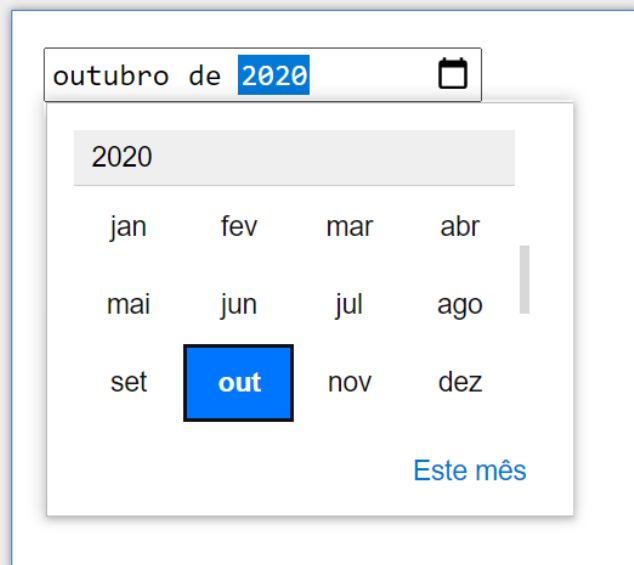
outubro de 2020							↑	↓		
D	S	T	Q	Q	S	S				
27	28	29	30	1	2	3			09	48
4	5	6	7	8	9	10			10	49
11	12	13	14	15	16	17			11	50
18	19	20	21	22	23	24			12	51
25	26	27	28	29	30	31			13	52
1	2	3	4	5	6	7			14	53
									15	54

Hoje

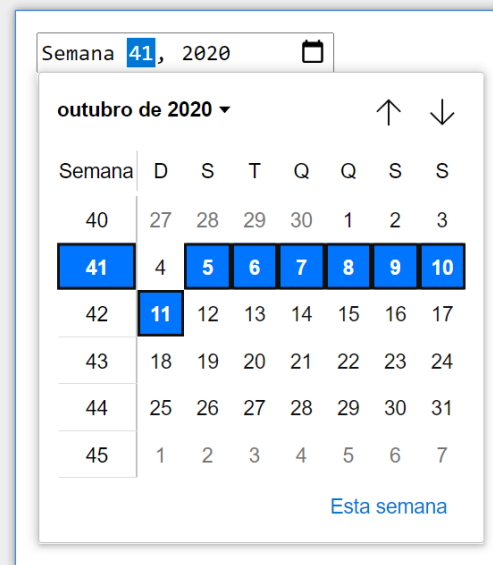
Exemplos de renderização no Google Chrome

# Campos para Mês e Semana

`<input type="month" ...>`



`<input type="week" ...>`



Exemplos de renderização no Google Chrome

# Observações sobre `<input type="date">`

- O formato da data apresentado no navegador depende das configurações locais de idioma e região do usuário
- Mas o valor submetido (propriedade `value`) está sempre no formato `YYYY-MM-DD`
  - Ex.: 2024-01-15
  - Padrão internacional ISO 8601

# Observações sobre `<input type="datetime-local">`

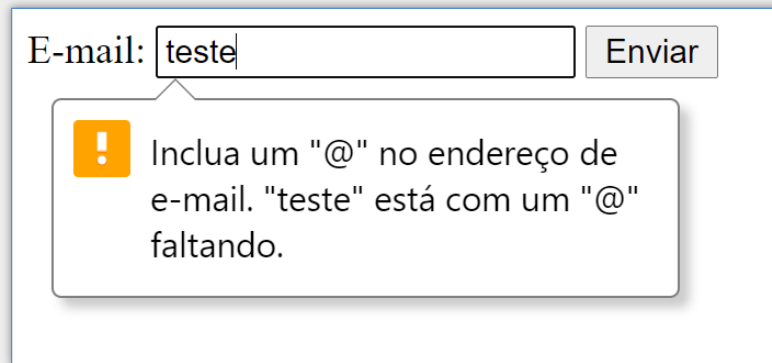
- O formato apresentado no navegador depende das configurações locais de idioma e região do usuário
- Mas o valor submetido (propriedade `value`) está sempre no formato `yyyy-MM-ddThh:mm`
  - Ex.: 2024-01-15T16:40
  - Padrão internacional ISO 8601

# Observações sobre `<input type="datetime-local">`

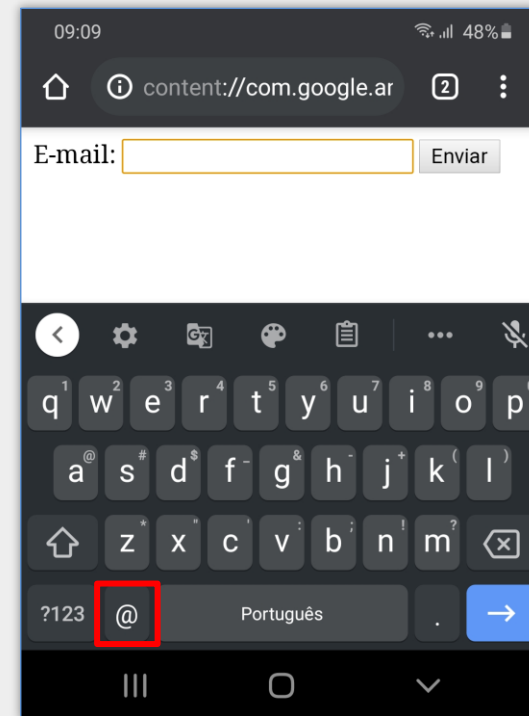
- Desde 2021 é suportado pelos principais navegadores
- Quando não suportado é apresentado como campo textual simples
- Pode-se utilizar os atributos `min` e `max` para restringir a data/hora
  - Formato: `YYYY-MM-DDThh:mm`
- Não utilize `<input type="datetime">` (obsoleto)

# Campo para E-mail

```
<input type="email">
```



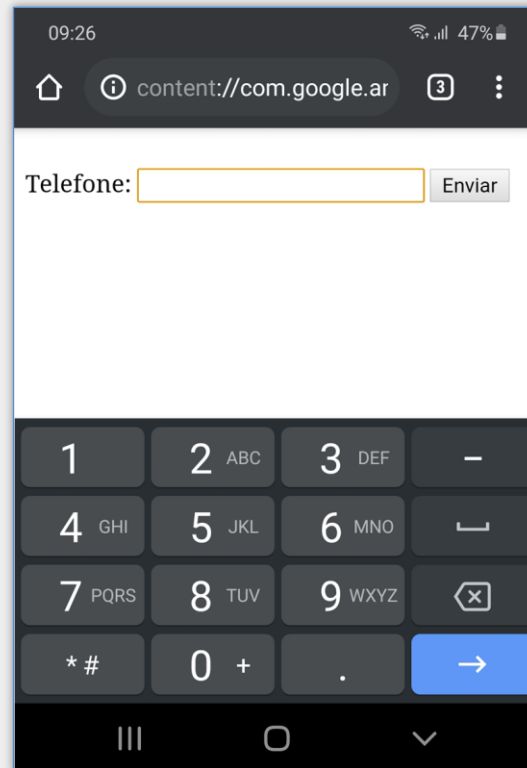
Navegador exige caracter @



Teclado com @ no smartphone

# Campo para Número de Telefone

```
<input type="tel">
```



Teclado numérico no smartphone

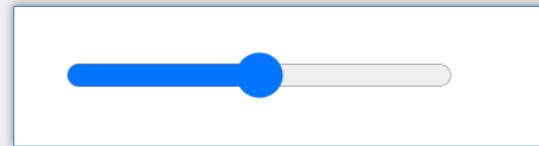


# Campos number e range

```
<input type="number">
```

Filhos:

```
<input type="range">
```

A range slider control consisting of a horizontal track with a blue circular handle positioned at approximately one-third of the way across the track.

- Campos para valores numéricos
- Permite o incremento dos números pelos botões / slider
- Responde às setas cima/baixo do teclado
- Os atributos **min**, **max** e **step** definem o valor **mínimo**, **máximo** e **incremento**
- O intervalo padrão do campo **range** é 0-100 com incremento (**step**) de 1
- **step** não precisa ser inteiro (nesse caso os campos aceitarão não inteiros)

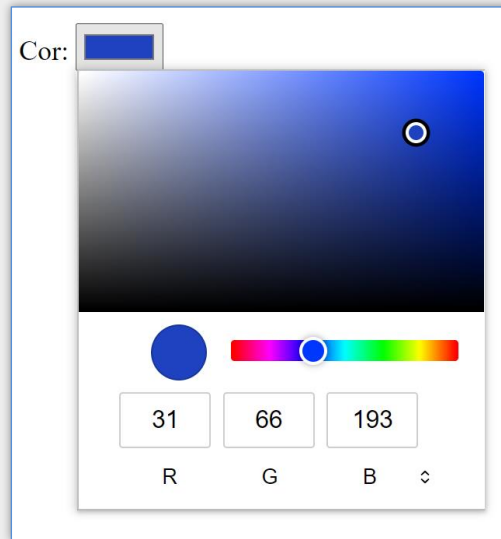
# Campo Oculto hidden

```
<input type="hidden">
```

- Não aparece para o usuário
- Comumente utilizado para envio de dados de controle
- Utilize os atributos **name** e **value** para definir nome e valor

# Campo para Seleção de Cor

```
<input type="color">
```



O valor do atributo **value** será o código RGB em hexadecimal (ex.: #55af00)

# Campo select

Estado: Amazonas

- Amazonas
- Bahia
- Minas Gerais
- Rio de Janeiro
- São paulo
- Outro

```
<label for="estado">Estado: </label>  
<select id="estado" name="estado">  
  <option value="AM" selected>Amazonas</option>  
  <option value="BA">Bahia</option>  
  ...  
</select>
```

- Permite seleção de um ou vários itens de uma lista suspensa
- Cada opção é inserida com o elemento `<option>`
- O conteúdo de `<option>` é apresentado, mas é o seu `value` que é enviado
- O atributo `selected` dentro de `<option>` pré-seleciona a opção
- O atributo `multiple` possibilita a seleção de vários itens
- Se necessário, utilize um `<label>` para rotular o campo

# Campo select Definido como Obrigatório

Estado:

```
<p><label>Estado:  
  <select name="estado" required>  
    <option value="">Selecione</option>  
    <option value="AM">Amazonas</option>  
    <option value="BA">Bahia</option>  
  </select>  
</label></p>
```

Quando um `<select>` de seleção simples é definido como obrigatório (`required`) e o campo não possui uma opção pré-selecionada (`selected`), então deve-se utilizar um elemento `<option>` com `value=""` para atuar como **placeholder** e instruir o usuário a escolher alguma opção.

# Campo textarea

Deixe seu comentário:

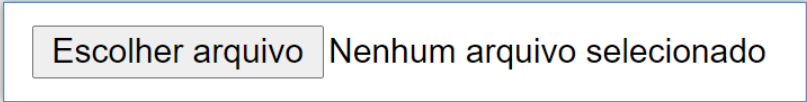
campo textual de  
múltiplas linhas

```
<textarea name="" rows="" cols="">  
    texto inicial do campo  
</textarea>
```

- Campo para textos extensos, com **múltiplas linhas**
- O tamanho inicial pode ser definido com os atributos **rows** e **cols**
- Diferente dos demais, **seu conteúdo** é enviado (**não suporta o atrib. value**)
- O atributo **maxlength** limita a quantidade de caracteres

# Campo para Arquivo

```
<input type="file" ... >
```



Escolher arquivo Nenhum arquivo selecionado

- Deve-se utilizar `enctype="multipart/form-data"` no elemento `<form>`
- O atributo `multiple` permite a seleção de vários arquivos
- O atributo `accept` permite restringir o tipo de arquivo permitido
  - `<input type="file" accept="image/*">`
  - `<input type="file" accept=".png, .jpg, .jpeg">`
- O atributo `capture` permite capturar imagens/sons da câmera/microfone, mas possui baixa compatibilidade dos navegadores
  - Valores: `user` ou `environment`

# Outros Atributos Booleanos

## autofocus

- O campo recebe o foco (cursor piscando) assim que a página é carregada

## readonly

- Torna o campo somente leitura, impossibilitando a alteração do seu valor
- O campo é **submetido normalmente**

## disabled

- Desabilita o campo. O usuário não poderá interagir ou editar
- Campos desabilitados **não são submetidos** com o formulário

## required

- Indica que o campo é de preenchimento obrigatório
- O navegador não envia o formulário sem que o campo seja preenchido



# Outros Atributos para Campos Textuais

## pattern

- Permite indicar expressão regular para filtrar os valores permitidos
- Ex.: `<input type="text" pattern="\d{3}\.\d{3}\.\d{3}-\d{2}">`
  - Aceita um texto no formato xxx.xxx.xxx-xx, onde x são números
  - Não é uma máscara, mas uma restrição
- Ex.: `<input type="text" pattern="[A-Za-z]{2}">`
  - Aceita duas letras com os caracteres alfabéticos A-Z e a-z

# Outros Atributos para Campos Textuais

## minlength e maxlength

- Permitem definir o número mínimo e o número máximo de caracteres permitido para o campo, respectivamente

## size

- Indica a largura de apresentação de um campo `<input>` e a altura de um `<select>`

## placeholder

- Permite a inserção de uma dica de preenchimento
- A dica é apagada quando o campo é preenchido
- Ex: `<input type="text" placeholder="Informe o nome completo">`



Informe o nome completo

# Observação Sobre Validação de Formulários

A validação de formulários utilizando atributos da própria HTML não substitui os scripts de validação, os quais garantem que os dados inseridos estejam no formato adequado. É muito fácil para o usuário fazer ajustes no HTML para ignorar a validação ou removê-la completamente. Se a programação no servidor não validar os dados, falhas diversas podem ocorrer devido a dados em formatos inadequados (por exemplo, dados muito grandes, dados do tipo incorreto etc.).

Mensagem adaptada de [developer.mozilla.org](http://developer.mozilla.org)

# Referências

- <https://developer.cdn.mozilla.net/en-US/docs/Web/HTML>
- <https://html.spec.whatwg.org/multipage/>
- <https://www.w3.org/TR/html52/>
- <https://www.w3schools.com/html/>
- **HTML and CSS: Design and Build Websites**, Jon Duckett.