

# **Desenvolvimento de Interfaces**

## **Aula 4 HTML – Introdução**

## Agenda

- **Conceito de Internet e Web**
- **Tecnologias client-side**
- **Exibindo informações na Web**
- **HTML**



## Introdução

### Como vimos anteriormente....

A **Internet** é uma gigantesca rede mundial que conecta milhões de computadores, permitindo que eles troquem informações. Pense nela como um sistema de estradas que ligam cidades ao redor do mundo, possibilitando a troca de mensagens, arquivos e dados em tempo real.

A **Web (World Wide Web)** é um dos serviços mais populares da internet. Ela é um **conjunto de páginas e sites** que você acessa usando um navegador (como Google Chrome, Firefox ou Safari). A Web foi criada para **compartilhar informações de forma fácil e rápida**, e hoje é o principal meio de comunicação, entretenimento e negócios online.

## O que é um Website e uma Aplicação Web?

### Website

Um **website** (ou site) é um conjunto de páginas interligadas acessíveis pela internet. Ele é usado principalmente para **exibir informações** como textos, imagens e vídeos. Um website pode ser um blog, um site institucional ou um portfólio online.

- **Como funciona:** As páginas de um site são armazenadas em um **servidor** e podem ser acessadas por meio de um navegador (como Chrome, Firefox ou Edge) usando um endereço único chamado **URL** (por exemplo, [www.exemplo.com](http://www.exemplo.com)).
- **Exemplos:** Blogs, sites de notícias, páginas de empresas e portfólios pessoais.

## O que é um Website e uma Aplicação Web?

### Aplicação Web

Uma **aplicação web** é um tipo de website mais **dinâmico e interativo**, projetado para permitir que os usuários **realizem tarefas específicas**. Ao contrário de um **site estático**, ela responde às ações do usuário, como enviar formulários, fazer compras online, gerenciar contas ou trabalhar com dados em tempo real.

**Como funciona:** Além de exibir informações, uma aplicação web **processa dados** e geralmente está conectada a um **banco de dados** para armazenar e recuperar informações.

**Exemplos:** Redes sociais (Facebook), serviços de e-mail (Gmail), plataformas de e-commerce (Amazon), bancos online e ferramentas de produtividade (Google Docs).

## O que é um Website e uma Aplicação Web?

### Principais Diferenças e Características:

#### Website:

- **Estático:** Focado em **exibir informações** de forma simples e direta.
- **Menos interativo:** O usuário basicamente lê ou assiste ao conteúdo, com pouca interação além de navegar entre as páginas.

#### Aplicação Web:

- **Dinâmica:** Permite **interações complexas** e **processamento de dados**.

## O que é um Website e uma Aplicação Web?

### Principais Diferenças e Características:

#### Aplicação Web:

- **Altamente interativa:** O usuário pode enviar informações, receber respostas instantâneas e interagir em tempo real.
- **Atualização em Tempo Real:** Dados podem ser atualizados na tela sem precisar recarregar a página (como em redes sociais ou chats online).
- **Integração com Bancos de Dados:** Armazena informações para personalizar a experiência do usuário, como em lojas virtuais ou sistemas de login.

## Exibindo informações na Web

A única linguagem que o navegador consegue interpretar para a exibição de conteúdo é o HTML. Para iniciar a exploração do HTML, vamos imaginar o seguinte caso: o navegador realizou uma requisição e recebeu como corpo da resposta o seguinte conteúdo:

**Uniceub Centro Universitário**

**Bem-vindo ao Uniceub.**

**Confira nossas cursos.**

**Receba informações sobre nossos cursos por e-mail.**

**Estude com os melhores professores.**

**Estude sem sair de casa.**

Para conhecer o comportamento dos navegadores quanto ao conteúdo descrito antes, vamos reproduzir esse conteúdo em um arquivo de texto comum, que pode ser criado com qualquer editor de texto puro.

Salve o arquivo como index.html e abra-o a partir do navegador à sua escolha.



## Tecnologias client-side



**Tecnologias client-side são tecnologias utilizadas no desenvolvimento web do lado cliente (parte da aplicação que roda no navegador),**

## Tecnologias client-side



**Tecnologias client-side são responsáveis por tudo o que você vê e interage em uma página web, como textos, imagens, botões, animações e formulários.**

### **Por que são importantes?**

**Essas tecnologias garantem que as páginas sejam rápidas, interativas e atraentes, pois o navegador carrega e executa o código diretamente no dispositivo do usuário, sem precisar ficar pedindo informações ao servidor o tempo todo.**

## Associação com Desenvolvimento Web



O **desenvolvimento web** é o processo de **criação de sites e aplicativos** que você acessa na Web.

É como construir lojas, casas e prédios ao longo das estradas da internet.

## Associação com Desenvolvimento Web



**Existem duas partes principais no desenvolvimento web:**

- **Client-Side (Front-End)** – Tudo o que você vê e com o que interage na tela.
- **Server-Side (Back-End)** – O que acontece nos bastidores, como processamento de dados e armazenamento de informações.

## Tecnologias client-side

As tecnologias cliente-side se dividem em:

- **Marcação:** trata-se da linguagem de marcação HTML, responsável por definir o que representa cada conteúdo em uma página e criar as ligações (links) para outros conteúdos;
- **Apresentação:** refere-se à tecnologia CSS, que é uma linguagem de estilo utilizada para definir a aparência de documentos escritos em HTML. Seu principal benefício é prover a separação entre o formato visual e o conteúdo de um documento;
- **Comportamento:** trata-se do JavaScript, utilizado para trazer mais dinamismo às páginas web através de recursos como controle de eventos, manipulação de elementos DOM e seus conteúdos, entre outras funcionalidades.

## Tecnologias client-side

As **tecnologias client-side** são executadas no navegador do usuário, tornando as páginas **rápidas, interativas e visualmente atraentes**.

**HTML** fornece a estrutura, **CSS** dá estilo e identidade visual, e **JavaScript** adiciona interatividade ou seja comportamento.

Elas são essenciais para criar **experiências dinâmicas e responsivas** na web moderna.

Essas tecnologias são a base para o desenvolvimento web e são indispensáveis para quem deseja criar sites e aplicações interativas.

## HTML

```
<html>  
  
  <head>  
  </head>  
  
  <body>  
    <header>  
    </header>  
  
    <section>  
  
    </section>  
  </body>  
</html>
```

**HTML significa HyperText Markup Language, ou Linguagem de Marcação de Hipertexto.**

**É uma linguagem de marcação padrão para a Web que é usada para definir a estrutura de um documento web.**

**O HTML é usado para definir elementos como cabeçalho, corpo e rodapé, bem como elementos de conteúdo, como imagens, vídeos e textos.**



## HTML

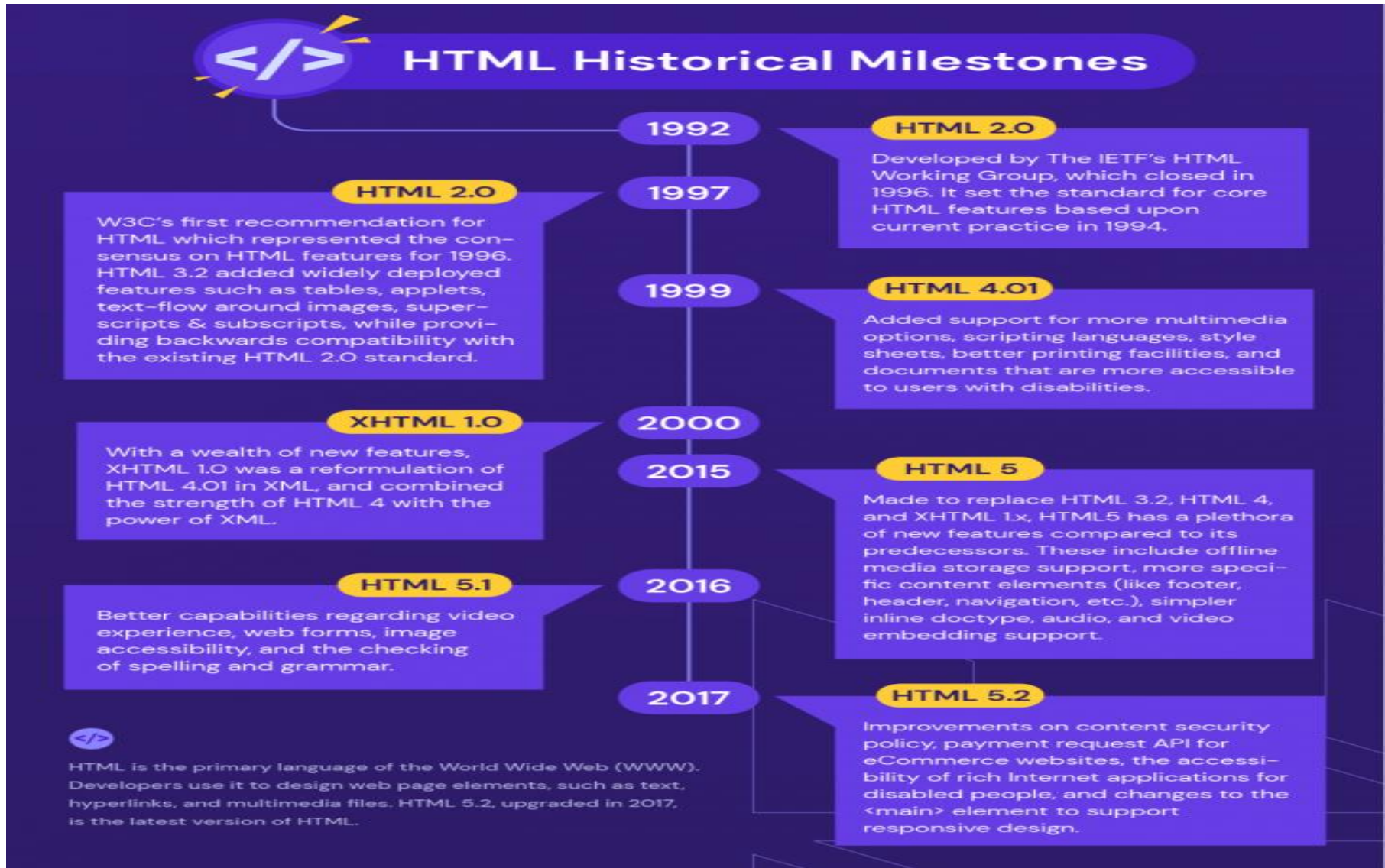


**A importância do HTML é que ele permite que os desenvolvedores criem páginas da web que sejam estruturadas e legíveis por navegadores.**

**O HTML também é usado para adicionar links a outras páginas da web, o que permite que os usuários naveguem pela Web.**



## HTML

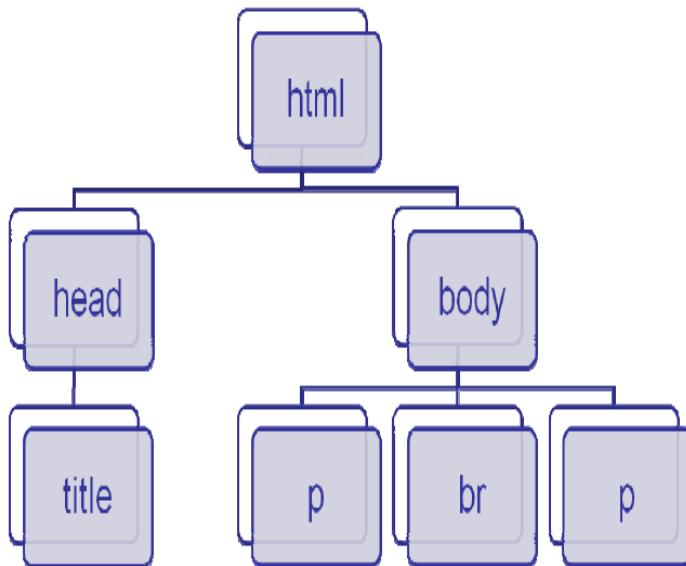


## HTML

### Documento HTML

Relação de elementos HTML organizados em árvore na qual alguns elementos são filhos de outros.

### Árvore de elementos



### Estrutura Geral doc HTML

```
<!DOCTYPE html>
<html>

  <head>
    <title>Título do Documento</title>
  </head>

  <body>
    <h1>Olá Mundo</h1>
  </body>

</html>
```

## HTML 5 - Estrutura geral



Um documento HTML é composto por **tags** que organizam e estruturam o conteúdo de uma página web.

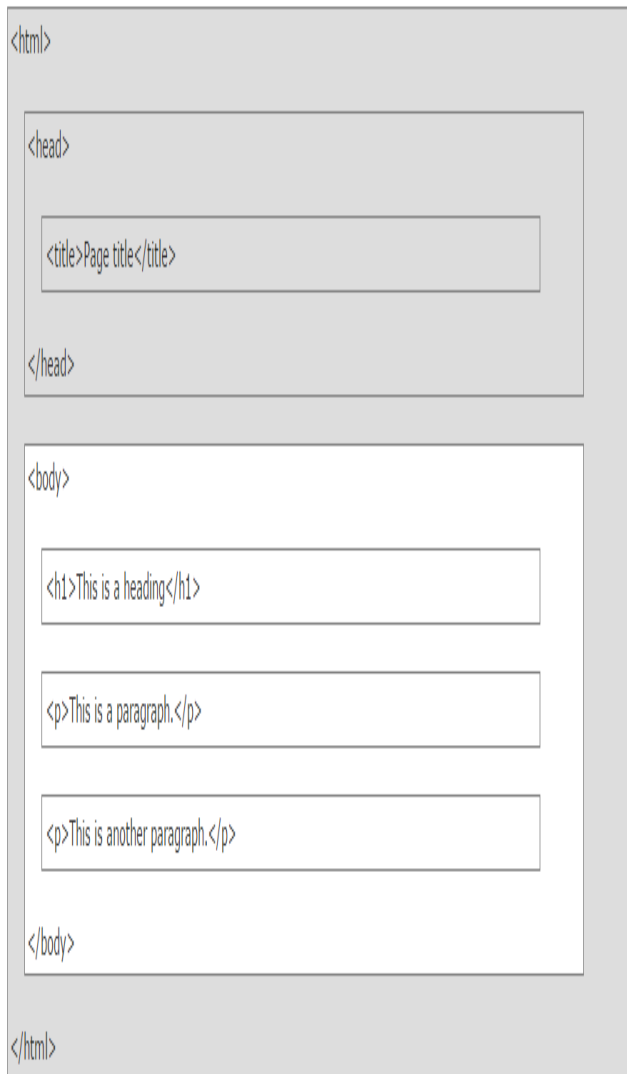
Essas tags informam ao navegador como exibir o conteúdo na tela. Sua estrutura básica se divide em:

**Instrução** `<!DOCTYPE html>`:

Declara que o documento é um HTML5, ajudando o navegador a renderizar a página corretamente. Não é uma tag HTML, mas deve estar no topo do arquivo.



## HTML 5 - Estrutura geral



**Elemento <html>:** A tag raiz que envolve todo o conteúdo da página. Indica o início e o fim do documento HTML. Pode incluir o atributo lang para definir o idioma (ex: <html lang="pt-br">).

**Elemento <head>:** Contém metadados sobre a página, como título (<title>), codificação de caracteres (<meta charset="UTF-8">), links para CSS e scripts. Essas informações não aparecem diretamente na tela.

**Elemento <body>:** Inclui todo o conteúdo visível da página, como textos, imagens, links, vídeos e botões. É o que o usuário vê e interage ao acessar o site.

## Tags HTML - esquema

- **Toda tag recebe um nome, uma identificação;**
- **O nome da tag deve estar escrito entre os sinais `<` e `>`. Por exemplo: `<p>;`**
- **Uma tem uma abertura e, na maioria das vezes, um fechamento, usando-se a barra `/` antes do nome. Por exemplo: `</p>;`**
- **Tags também podem apresentar atributos, que expandem suas características ou propriedades.**
- **Exemplo `<p lang="pt-br"> Texto </p>`**

## HTML - Visão geral

### Elemento HTML

**Geralmente contém três componentes:**

**Tag inicial, conteúdo e Tag final**

**Sintaxe de um elemento HTML:**

**<nometag> conteúdo </nometag>**

**Exemplos:**

<b>Tipo de elemento</b>	<b>sintaxe</b>
Conteúdo HTML	<html> ... </html>
Cabeçalho do documento	<head> ... </head>
Título do documento	<title> ... </title>
Corpo do documento (área visível do doc)	<body> ... </body>
Título de seção nível 1	<h1> ... </h1>
Parágrafo	<p> ... </p>

## HTML - Visão geral

### Elementos HTML

Tipo de elemento	sintaxe
Cria um link para outra página ou recurso	<code>&lt;a href="http://"&gt;&lt;/a&gt;</code>
Exibe uma imagem	<code>&lt;img src="" alt=""&gt;</code>
Cria listas não ordenadas (ul) e ordenadas (ol)	<code>&lt;ul&gt;</code> , <code>&lt;ol&gt;</code> , <code>&lt;li&gt;</code>
Define uma seção genérica da página	<code>&lt;div&gt;...&lt;/div&gt;</code>
Cria tabelas e suas partes	<code>&lt;ul&gt;</code> , <code>&lt;ol&gt;</code> , <code>&lt;li&gt;</code>
Define scripts JavaScript para a página	<code>&lt;script&gt; ... &lt;/script&gt;</code>
Define estilos CSS para a página	<code>&lt;style&gt; ... &lt;/style&gt;</code>
Define relações entre o documento	<code>&lt;link rel="stylesheet"</code>

## HTML - Visão geral

**Exemplos de utilização de elementos HTML**



## HTML 5 - Estrutura geral

### Exemplo

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <title>Uniceub Centro Universitário</title>
</head>

<body>
  <h1>Primeira Página</h1>
</body>

</html>
```

## Sintaxe do HTML

- O HTML é um conjunto de **tags** responsáveis pela marcação do conteúdo de uma página no navegador.
- No código que vimos antes, as tags são os elementos a mais que escrevemos usando a sintaxe `<nomedatag>`.
- Diversas tags são disponibilizadas pela linguagem HTML e cada uma possui uma funcionalidade específica.
- Exemplo o uso da tag `<h1>`.

```
<h1>Uniceub Centro Universitário</h1>
```

### OBS:.

- Uma tag é definida com caracteres `<` e `>`, e seu nome (h1 no caso).
- Muitas tags possuem conteúdo, como o texto do título ("Uniceub Centro Universitário").
- Nesse caso, para determinar onde o conteúdo acaba, usamos uma *tag de fechamento* com a barra antes do nome: `</h1>`.

## Sintaxe do HTML

**Algumas tags podem receber atributos dentro de sua definição.**

**São parâmetros usando a sintaxe de nome=valor. Para definir uma imagem, por exemplo, usamos a tag `<img>` e, para indicar qual imagem carregar, usamos o atributo `src`:**

**Exemplo ``**

**Repare que a tag `img` não possui conteúdo por si só. Nesses casos, não é necessário usar uma tag de fechamento como antes no `h1`.**

## HTML - Visão geral

### ❑ **Marcação (*markup*)**

- Realizada através das tags

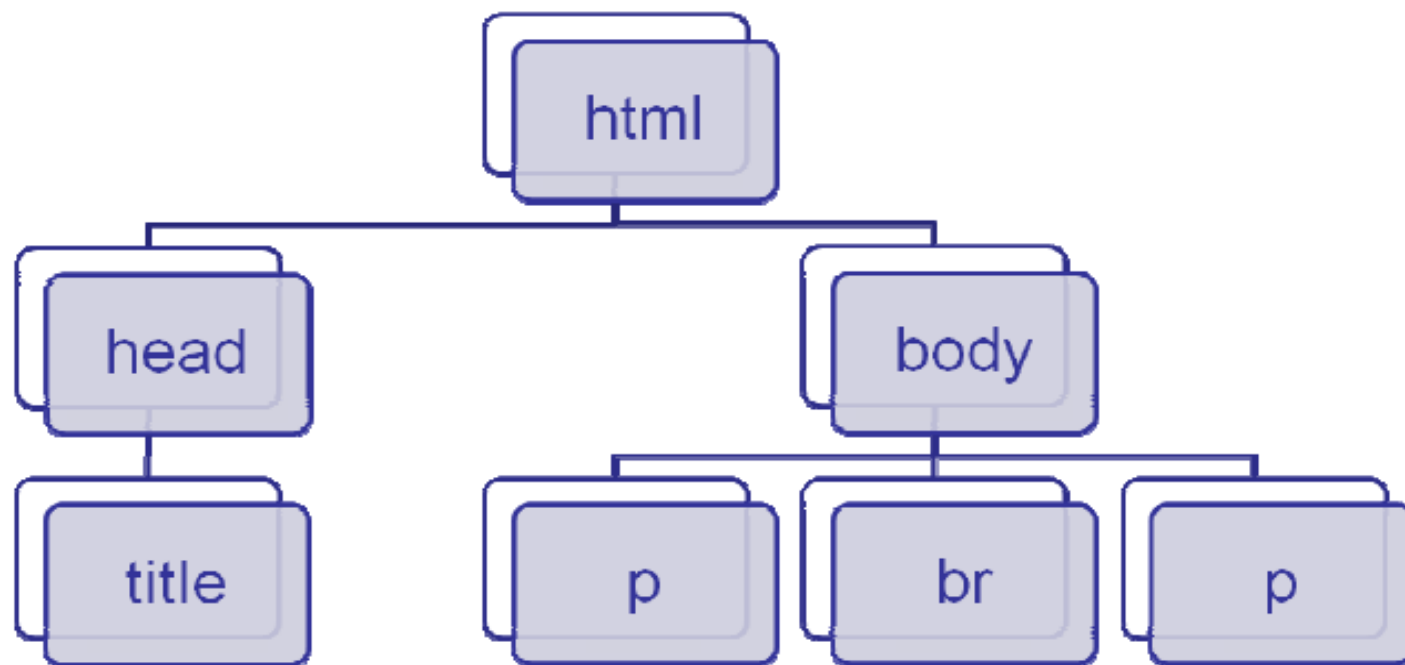
### ❑ **Tag HTML**

- Delimitadas por “<” e “>”
- Usadas para descrever um elemento
- Exemplo de tags HTML:
  - <html>
  - </html>
  - <head>
  - <p>
  - Etc...

## HTML - Visão geral

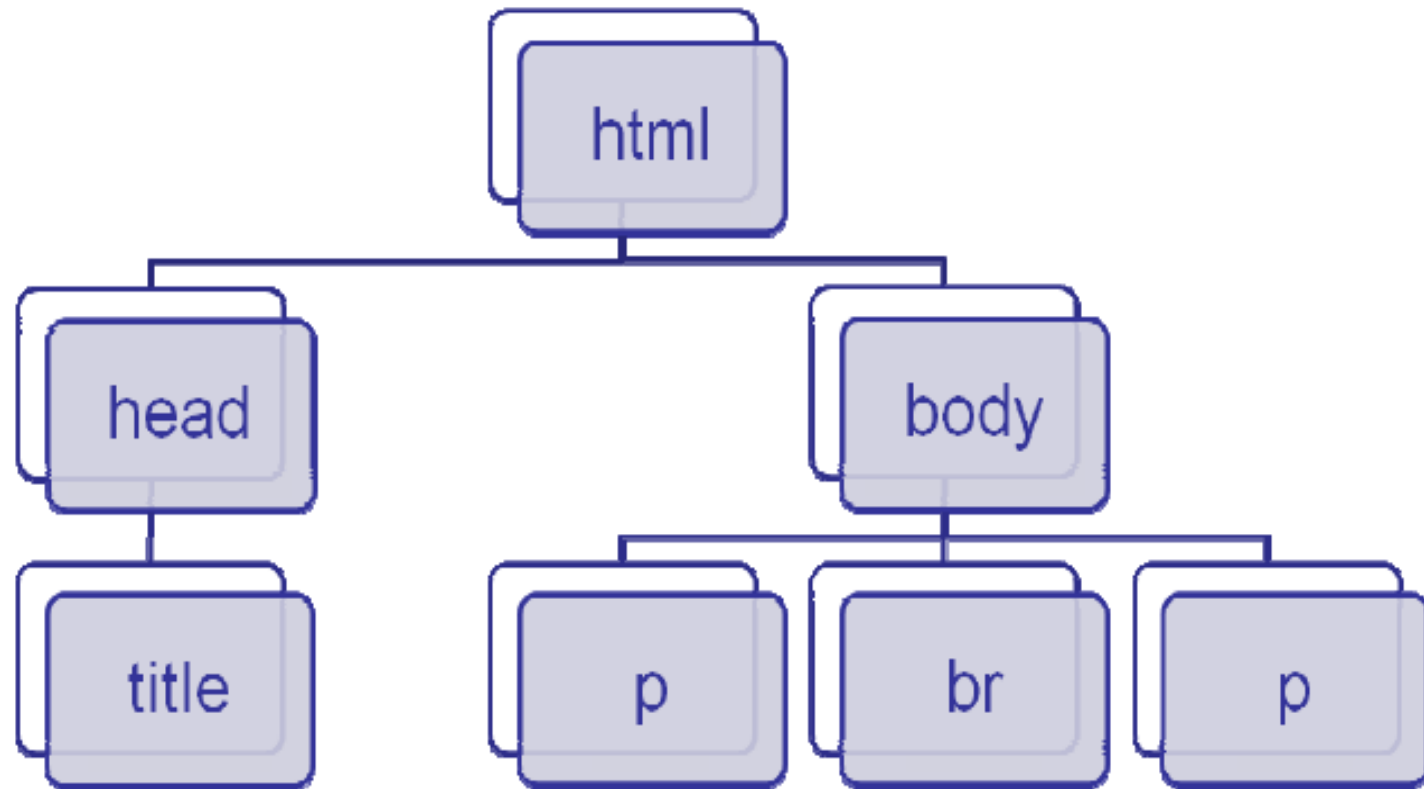
### □ Documento HTML

- Relação de elementos HTML organizados em árvore na qual alguns elementos são filhos de outros.



## HTML - Exemplo

### □ Árvore de elementos



## HTML - Visão geral

### ■ Elemento HTML

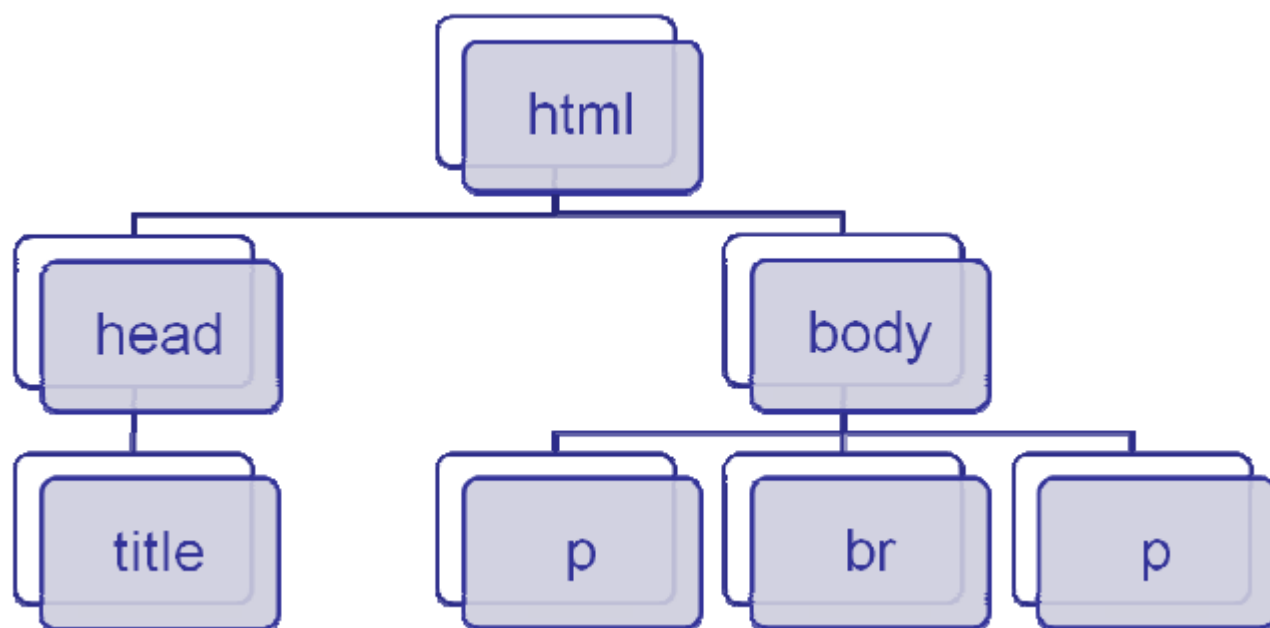
- Geralmente contém três componentes:
  - Tag inicial, conteúdo e Tag final
- Sintaxe de um elemento HTML:
  - `<nometag> conteúdo </nometag>`
- Exemplos:

Tipo de elemento	sintaxe
Conteúdo HTML	<code>&lt;html&gt; ... &lt;/html&gt;</code>
Cabeçalho do documento	<code>&lt;head&gt; ... &lt;/head&gt;</code>
Título do documento	<code>&lt;title&gt; ... &lt;/title&gt;</code>
Corpo do documento (área visível do doc)	<code>&lt;body&gt; ... &lt;/body&gt;</code>
Título de seção nível 1	<code>&lt;h1&gt; ... &lt;/h1&gt;</code>
Parágrafo	<code>&lt;p&gt; ... &lt;/p&gt;</code>

## HTML - Visão geral

### □ Elemento Raiz

- O elemento raiz desta grande árvore é sempre o “elemento HTML” ( `<html>` )





## HTML - Visão geral

### □ Extensão “.html”

- Um arquivo com documento HTML devem ter a extensão “.html”.

### □ Delimitadores no documento HTML

- Equivale a um delimitador qualquer combinação de:
  - Espaços
  - Tabulação
  - Mudança de linha

## Exemplo 1

**Abra o editor de sua preferência, digite o texto abaixo e Salve o arquivo com a extensão HTML.**

**Exemplo: index.html**

```
<!DOCTYPE HTML>
<html>
<head>
    <title>Título que aparecerá na barra de cima
    (de título) do navegador
    </title>
</head>
<body>
    Aqui vai o conteúdo da página.
</body>
</html>
```

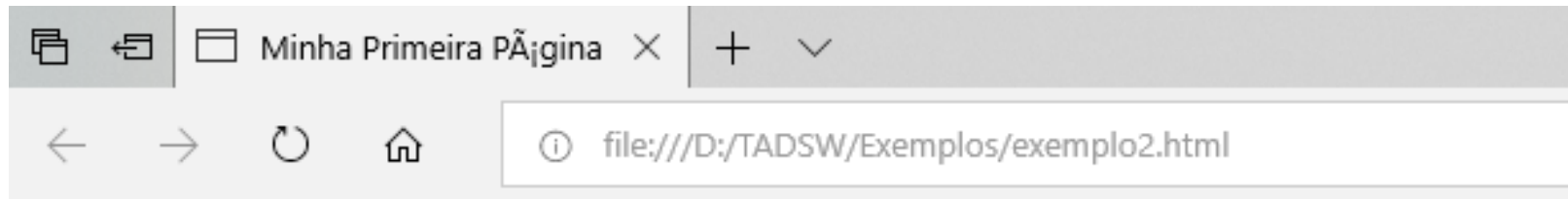
## HTML 5 – Exemplo 2

### Código html

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Minha Primeira Página</title>
</head>
<body>
  <p> Olá Mundo</p>
  <p> Olá Turma</p>
</body>
</html>
```

## HTML 5 - Exemplo

### Apresentação no browser



Olá Mundo

Olá Turma

## **Estrutura geral de um documento HTML**

## HTML 5 - Estrutura geral

- Um documento HTML possui a seguinte estrutura geral:
  - Instrução DOCTYPE (*identif. do tipo de documento*)
  - Elemento html (*descrição do documento html*)
  - Elemento head (*cabeçalho do documento*)
  - Elemento body (*corpo do documento*)

## HTML 5 - Estrutura geral

- **Instrução DOCTYPE**
- O DOCTYPE **não é uma tag HTML**, mas uma instrução especial. Ela indica para o navegador qual versão do HTML deve ser utilizada para renderizar a página.
- Utilizaremos `<!DOCTYPE html>`, que indica para o navegador a utilização da versão mais recente do HTML - a versão 5, **HTML 5**.
- Há muitos comandos complicados nessa parte de DOCTYPE que eram usados em versões anteriores do HTML e do XHTML.
- O recomendado é sempre usar a última versão do HTML, usando a declaração de DOCTYPE simples:

## HTML 5 - Estrutura geral

### Exemplo

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <title>Uniceub Centro Universitário</title>
</head>

<body>
  <h1>Primeira Página</h1>
</body>

</html>
```



## Características das tags

- Toda tag recebe um nome, uma identificação;
- O nome da tag deve estar escrito entre os sinais < e >.  
Exemplo <p>;
- Uma tem uma abertura e, na maioria das vezes, um fechamento, usando-se a barra / antes do nome.  
</p>;
- Tags também podem apresentar atributos, que expandem suas características ou propriedades. Exemplo  
<p lang="pt-br"> Texto </p>

## Tags completas - esquema

- Toda tag recebe um nome, uma identificação;
- O nome da tag deve estar escrito entre os sinais < e >.  
Exemplo <p>;
- Uma tem uma abertura e, na maioria das vezes, um fechamento, usando-se a barra / antes do nome.  
</p>;
- Tags também podem apresentar atributos, que expandem suas características ou propriedades. Exemplo  
<p lang="pt-br"> Texto </p>

## HTML 5 - Estrutura geral

### A tag <html>

- Um documento HTML válido precisa seguir obrigatoriamente a estrutura composta pelas tags <html>, <head> e <body> e a instrução <!DOCTYPE>. Vejamos cada uma delas:
- Na estrutura do nosso documento, antes de tudo, inserimos uma tag <html>.
- Dentro dessa tag, é necessário declarar outras duas tags: <head> e <body>. Essas duas tags são "irmãs", pois estão no mesmo nível hierárquico em relação à sua tag "pai", que é <html>.

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

## HTML 5 - Estrutura geral

### A tag `<head>`

A tag `<head>` contém informações sobre nosso documento que são de interesse somente do navegador, e não dos visitantes do nosso site.

São informações que não serão exibidas na área do documento no navegador.

A especificação obriga a presença da tag de conteúdo `<title>` dentro do nosso `<head>`, permitindo especificar o título do nosso documento, que normalmente será exibido na barra de título da janela do navegador ou na aba do documento.

## HTML 5 - Estrutura geral

### Charset - UTF-8

Outra configuração muito utilizada, principalmente em documentos cujo conteúdo é escrito em um idioma como o português, que tem caracteres como acentos e cedilha, é a configuração da codificação de caracteres, chamada de **encoding** ou **charset**.

Podemos configurar qual codificação queremos utilizar em nosso documento por meio da configuração de charset na tag `<meta>`. Um dos valores mais comuns usados hoje em dia é o UTF-8, também chamado de Unicode. Há outras possibilidades, como o latin1, muito usado antigamente.

O **UTF-8** é a recomendação atual para encoding na Web por ser amplamente suportada em navegadores e editores de código, além de ser compatível com praticamente todos os idiomas do mundo.

## HTML 5 - Estrutura geral

### tag <body>

A tag <body> contém o corpo do nosso documento, que é exibido pelo navegador em sua janela.

É necessário que o <body> tenha ao menos um elemento "filho", ou seja, uma ou mais tags HTML dentro dele.

Nesse exemplo, usamos a tag <h1>, que indica um título.

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <title>Uniceub Centro Universitário</title>
</head>

<body>
  <h1>Primeira Página</h1>
</body>

</html>
```

## HTML 5 - Estrutura geral

**Somente o elemento <body> é apresentado pelo browser.**

```
<html>
```

```
<head>
```

```
<title>Page title</title>
```

```
</head>
```

```
<body>
```

```
<h1>This is a heading</h1>
```

```
<p>This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
</body>
```

```
</html>
```

## HTML 5 - A ESTRUTURA DOS ARQUIVOS DE UM PROJETO

**Existem várias formas de organizar as pastas de um projeto, uma das recomendações é que exista uma pasta do projeto e dentro dela pastas individualizadas de imagens, arquivos JScript etc...**

**Ex**

**ProjWeb**

**imagem**

**css**

**scripts**



## Adicionando atributos...

- ✓ **As tags HTML, às vezes precisam de informações extras que poderão variar de acordo com a formatação desejada em cada seção da página. A essas informações damos o nome de propriedades.**
- ✓ **Cada tag poderá ter um conjunto de propriedades que lhe proporcionará melhor caracterização. Essas propriedades devem ser declaradas junto com a inicialização da tag e seus valores estarão acondicionados entre aspas.**
- ✓ **Veja o exemplo:**

```
<BODY bgcolor="#FF0000"> ...  
</BODY>
```

## Alterando a cor de fundo da página...

- ✓ Acrescentando alguns atributos ao exemplo anterior. Altere a tag <BODY> com os atributos abaixo e verifique como ela se comportará.

```
<BODY bgcolor="#FFFF00" text="#0000FF">
```

- No caso descrito acima, alteramos a cor de fundo da página – propriedade bgcolor – para ser visualizada com a cor: #FF0000 e que a cor dos textos existentes no corpo do documento terão a cor #0000FF.
- ✓ As cores utilizada na nossa página estão em código hexadecimal. Cada cor tem o seu próprio código e o padrão de cores RGB. Veja no anexo a cor e o respectivo código hexadecimal.
- ✓ As cores mais comuns podem ser escritas com o seu nome equivalente em inglês. Veja o exemplo:

```
<BODY bgcolor="yellow" text="blue">
```

## Outras propriedades para a tag Body...

```
<BODY link="#333333" vlink="#009966"  
      alink="#FF0000">
```

```
<BODY leftmargin="0" topmargin="0" marginwidth="0"  
      marginheight="0">
```

- ✓ As propriedades **link**, **vlink** e **alink** dizem quais cores os links assumirão quando eles estiverem no estado inicial, quando for um link que já foi visitado e quando ele estiver ativo, respectivamente.
- ✓ As propriedades **leftmargin**, **topmargin**, **marginwidth** e **marginheight** representam as margens esquerda, superior, direita e inferior a serem utilizadas na página. As margens deve ser especificadas em pixels.
- ✓ Pode-se utilizar todas essas propriedades juntas ou somente aquelas necessárias.

## Exercício – Página simples com cor

- ✓ Digite o código abaixo e salve-o como `exercicio2_2_1.html`.

```
<HTML>
  <HEAD>
    <TITLE>Documento com fundo azul!!</TITLE>
  </HEAD>
  <BODY bgcolor="#0000FF" text="#00FF00">
    Página com fundo e texto coloridos!
  </BODY>
</HTML>
```

- ✓ Agora faça as seguintes alterações:
  - Altere a cor de fundo
  - Altere a cor do texto

**Obrigado**