

HTML básico

BCC481 Programação Web

Vander Luis de Souza Freitas

- **H**yper **T**ext **M**arkup **L**anguage (HTML);
- Descreve a estrutura de uma página Web por uma série de elementos, descritos por tags, como “heading”, “paragraph”, etc.
- Os Browsers utilizam tais tags para renderizar o conteúdo da página.

Exemplo de estrutura de uma página:

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    ... conteúdo da página ...
  </body>
</html>
```

Exemplo de estrutura de uma página:

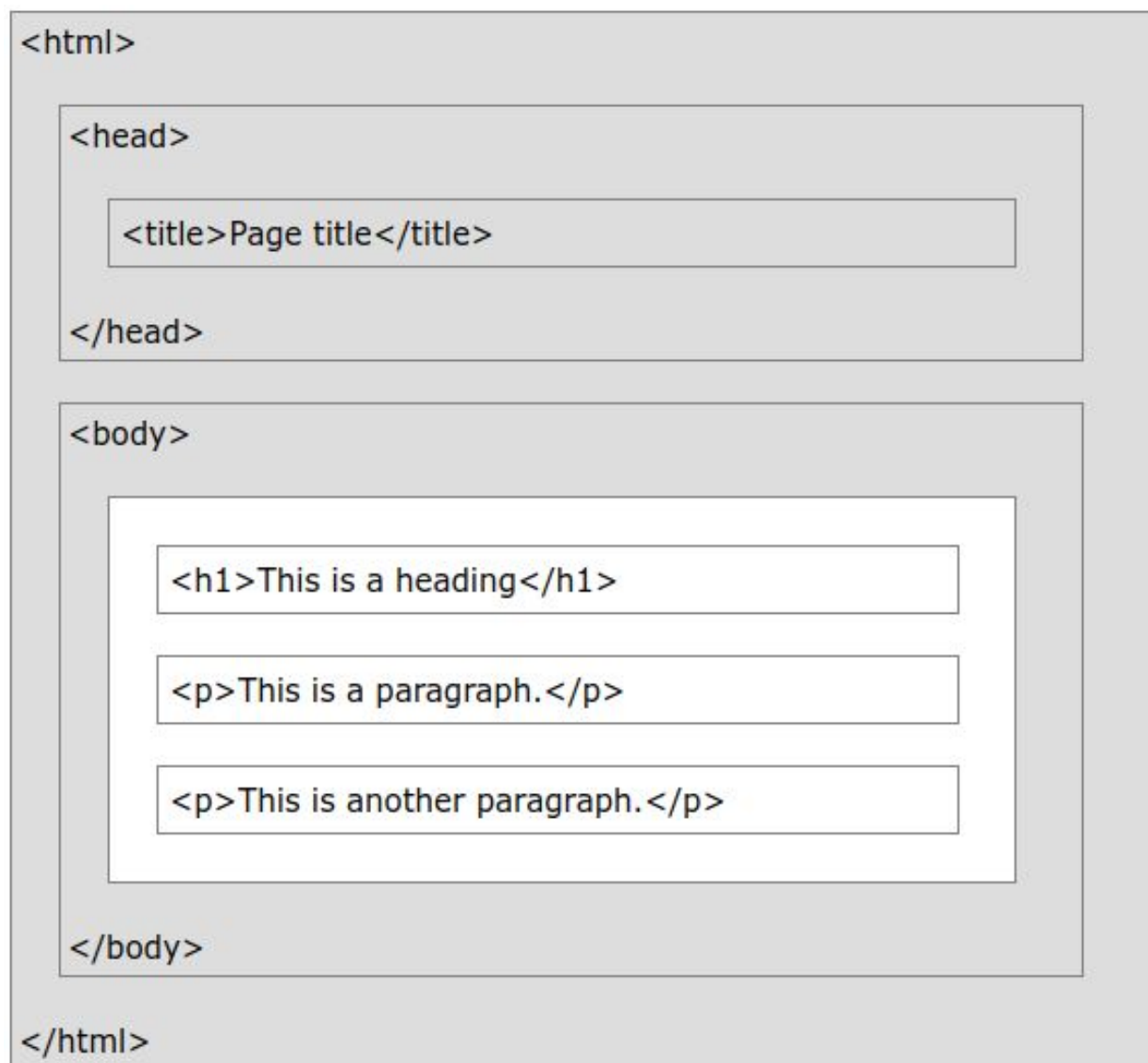
```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    ... conteúdo da página ...
  </body>
</html>
```

Metadados que não aparecem na viewport do browser

Aparece como nome da página, na aba do browser

Conteúdo que é renderizado na viewport do browser

Exemplo de estrutura de uma página:



`<!DOCTYPE html>` define que trata-se de um documento HTML5

`<html>` é a raiz da página

`<head>` contém informações sobre o documento (metadados):

`<title>`, `<style>`, `<meta>`, `<link>`, `<script>` e `<base>`

`<title>` especifica o título

`<body>` contém o conteúdo visível da página

`<h1>` define um tópico grande

`<p>` define um parágrafo

As tags geralmente vem em pares do tipo `<p>` and `</p>`

HTML

Versões do HTML, indicadas ao browser por meio da tag `<!DOCTYPE>`

Version	Year
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

HTML

HTML 5

```
<!DOCTYPE html>
```

HTML 4.01 Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```

HTML 4.01 Transitional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

HTML 4.01 Frameset

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
"http://www.w3.org/TR/html4/frameset.dtd">
```


`<p>`

HTML is `awesome!!!`

``

`</p>`

- Um elemento geralmente tem tags de início e término (`<p>` e `</p>`)
- Conteúdo: coisas entre as tags de início e fim
- Um elemento pode ser fechado em si mesmo (`img`)
- Um elemento pode ter atributos (`src="puppy.png"`)
- Elementos podem conter outros elementos (`p` contém `em` e `img`)
- HTML não é case sensitive, ou seja, `<P>` é o mesmo que `<p>`

Algumas tags:

Títulos h1, h2, ... h6	<code><h1>Título</h1></code>
Parágrafo	<code><p>Olha aqui um parágrafo!</p></code>
Quebra de linha	Não há linha que não possa <code>
</code> ser quebrada
Imagem	<code></code>
Link	<code>click here!</code>
Texto em negrito	<code>Be BOLD</code>
Ênfase (itálico)	Palavras <code>like these</code> são bons exemplos

Exemplo:



Exemplo:

Código HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    ...
  </body>
</html>
```

Texto da página

BCC481: Programação Web

News:

Horários de atendimento
disponibilizados
Bora estudar!

UFOP

Exemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    <h1>BCC481: Programação Web</h1>
    <strong>News:</strong><br/>
    Horários de atendimento disponibilizados<br/>
    Bora estudar!<br/>
    <br/>
    <a href="http://ufop.br"> UFOP </a>
  </body>
</html>
```

Exemplo:



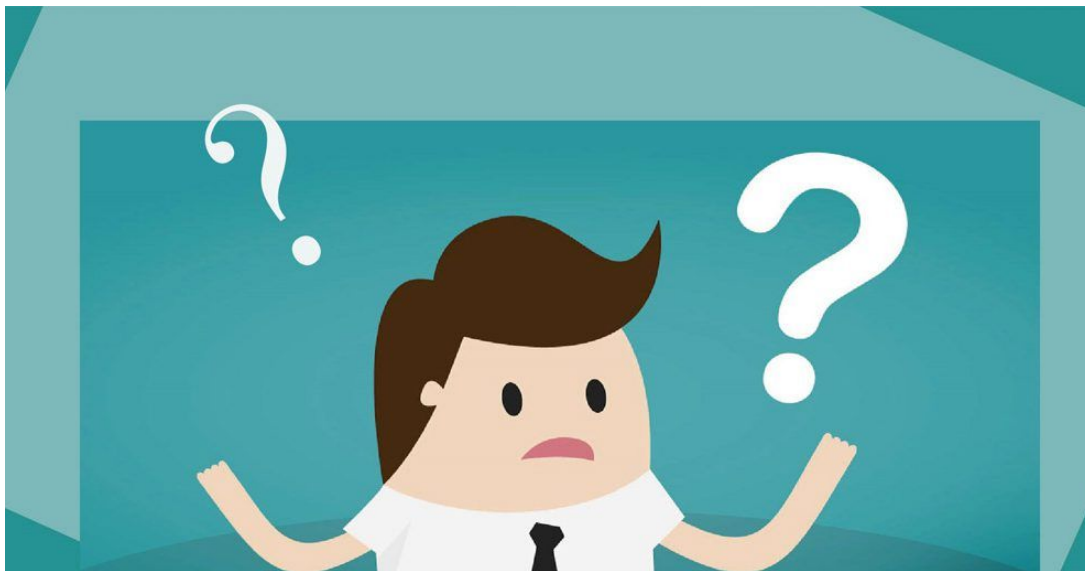
```
<!DOCTYPE html>
<html>
  <head>
    <title>BCC481</title>
  </head>
  <body>
    <h1>BCC481: Programação Web</h1>
    <strong>News:</strong><br/>
    Horários de atendimento disponibilizados<br/>
    Bora estudar!<br/>
    <a href="http://ufop.br"> UFOP </a>
  </body>
</html>
```

Estranhamente, o texto dentro de `<h1>` ficou em uma linha reservada, enquanto o texto dentro de `` não.

```
<h1>BCC481: Programação Web</h1>
```

```
<strong>News:</strong><br/>
```

```
Horários de atendimento disponibilizados<br/>
```



Why, God?!

Elementos
Block vs Inline.
Trataremos
disso em breve

Headings: entre `<h1>` e `<h6>`, sendo `<h1>` o heading mais “importante” e `<h6>` o menos “importante”.

Sou o h1

Sou o h2

Sou o h3

Sou o h4

Sou o h5

Sou o h6

Parágrafos: `<p>`Eu sou um parágrafo`</p>`

Links: `` Link to google``

Imagens: ``

Botões: `<button>`Click me`</button>`

Click me

Listas:

```
<ul>
```

```
  <li>Coffee</li>
```

```
  <li>Tea</li>
```

```
  <li>Milk</li>
```

```
</ul>
```

- Coffee

- Tea

- Milk

```
<ol type="1">
```

```
  <li>Coffee</li>
```

```
  <li>Tea</li>
```

```
  <li>Milk</li>
```

```
</ol>
```

1. Coffee

2. Tea

3. Milk

- type="1": Números
- type="A": Letras maiúsculas
- type="a": Letras minúsculas
- type="I": Números romanos maiúsculos
- type="i": Números romanos minúsculos

Horizontal rules:

```
<h1>This is heading 1</h1>
<p>This is some text.</p>
<hr />
<h2>This is heading 2</h2>
<p>This is some other text.</p>
<hr />
```

`<hr />` define uma quebra temática na página, mostrando separação de conteúdo.

This is heading 1

This is some text.

This is heading 2

This is some other text.

define uma quebra temática na página, mostrando separação de conteúdo.

Elemento head:

Define título do documento, charset, styles, scripts e outros metadados.

```
<head>  
  <title>My First HTML</title>  
  <meta charset="UTF-8">  
</head>
```

Formatação de texto:

`` - Bold text `` `
`
`` - Important text ```
`
`<i>` - Italic text `</i>``
`
``- Emphasized text ```
`
`<mark>`- Marked text `</mark>``
`
`<small>`- Small text `</small>``
`
``- Deleted text ```
`
`<ins>`- Inserted text `</ins>``
`
X`_{`Subscript text `}``
`
Y`^{`Superscript text `}``
`

- **Bold text**
- **Important text**
- *Italic text*
- *Emphasized text*
- **Marked text**
- Small text
~~- Deleted text~~
- Inserted text
XSubscript text
YSuperscript text

Comentários:

```
<!-- Write your comments here -->
```

```
<!-- Do not display this image at the moment  
  
-->
```

Imagens:

```

```

Recomendação: utilizar style (CSS) ao invés dos atributos width e height.

Dica: sempre especifique as dimensões da imagem, ou o browser pode começar a piscar (flicker) enquanto a imagem é carregada

Tag <picture>: Especifica qual imagem é exibida, a depender do recurso disponível (dispositivo, resolução)

```
<picture>
  <source media="(min-width: 650px)" srcset="img_food.jpg">
  <source media="(min-width: 465px)" srcset="img_car.jpg">
  
</picture>
```

Nota: o browser sempre escolhe o primeiro elemento <source> cujos requisitos são atendidos e ignora o restante.

Recomendação: Sempre especifique um como último filho do elemento <picture>, pois ele será usado por todos os browsers que não suportarem o elemento <picture> ou as tags <source> definidas

Iframe: Insere uma página web dentro de outra página web

```
<iframe src="demo_iframe.htm"
style="height:200px;width:300px;border:none;"></iframe>
```

ou com bordas estilizadas:

```
<iframe src="demo_iframe.htm" style="border:2px solid red;"></iframe>
```


Link Bookmark: Utilizados para pular para partes específicas de uma mesma página. Quando o link é clicado, a página rola até a parte desejada.

Inicialmente, dê um id para a porção desejada da página e depois crie um link para esse id, como segue:

```
<h2 id="C4">Chapter 4</h2>  
<a href="#C4">Jump to Chapter 4</a>
```

repare no símbolo # dentro do link.

É possível também criar um link para uma parte específica de outra página:

```
<a href="html_demo.html#C4">Jump to Chapter 4</a>
```

HTML5

Os novos elementos mais interessantes são:

- Elementos semânticos: `<header>`, `<footer>`, `<article>` e `<section>`.
- Atributos de elementos de formulário: `number`, `date`, `time`, `calendar` e `range`.
- Novos elementos gráficos: `<svg>` e `<canvas>`.
- Novos elementos multimídia: `<audio>` e `<video>`.

Novas APIs:

- HTML Geolocation
- HTML Drag and Drop
- HTML Local Storage
- HTML Application Cache
- HTML Web Workers
- HTML SSE

HTML5

HTML5Shiv é utilizado para garantir compatibilidade para browsers IE mais antigos que o IE9.

Recomendação: Utilize sempre o **HTML5Shiv** quando utilizar os novos elementos do HTML5, como `<article>`, `<section>` Basta inserir o seguinte código no `<head>` após todos os stylesheets:

`<head>`

`<!--[if lt IE 9]>`

`<script`

`src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>`

`<![endif]-->`

`</head>`

HTML5

Lista completa dos novos elementos do HTML5:

https://www.w3schools.com/html/html5_new_elements.asp

Semântica introduzida nos novos elementos de HTML5:

https://www.w3schools.com/html/html5_semantic_elements.asp

De acordo com o W3C, uma semântica Web: “Permite dados serem compartilhados e reutilizados entre aplicações, companhias e comunidades”.

HTML5

Guia de estilo HTML5 e convenções de código:

https://www.w3schools.com/html/html5_syntax.asp

- Use o doctype correto
- Letras minúsculas
- Sempre fechar os elementos
- Feche elementos vazios. Exemplo: `
`
- Valores de atributos sempre entre aspas
- Sempre adicione o atributo alt nas imagens.
- Não use espaços entre os símbolos de igual (=)
- Código com linhas curtas (máximo 80 caracteres)
- Não use linhas em branco a toa
- Use dois espaços para indentação, ao invés de TAB
- Não omita os elementos `<html>`, `<head>` e `<body>`
- Insira o método viewport no elemento `<meta>`
- Use um espaço antes de abrir `{` no CSS.

HTML5

Pode-se criar também novos elementos, como o <myHero>, por exemplo:

```
<!DOCTYPE html>
<html>
<head>
  <script>document.createElement("myHero")</script>
  <style>
    myHero {
      display: block;
      background-color: #dddddd;
      padding: 50px;
      font-size: 30px;
    }
  </style>
</head>
<body>
  <h1>A Heading</h1>
  <myHero>My Hero Element</myHero>
</body>
</html>
```

Viewport: Elemento incluído no HTML5 que diz como uma página deve ser dimensionada, levando em conta o dispositivo. Trata-se da área visível pelo usuário. É declarada dentro da tag `<head>`

```
<meta name="viewport" content="width=device-width,  
initial-scale=1.0">
```

Nota: Sempre inclua viewport em todos os websites

Layout: Uma página web é organizada em várias colunas e linhas. O HTML dispõe de diversos elementos semânticos para definição de layouts, como:

`<header>`- Defines a header for a document or a section

`<nav>` - Defines a container for navigation links

`<section>`- Defines a section in a document

`<article>`- Defines an independent self-contained article

`<aside>`- Defines content aside from the content (like a sidebar)

`<footer>`- Defines a footer for a document or a section

`<details>`- Defines additional details

`<summary>`- Defines a heading for the `<details>` element



HTML Screen Readers (Web semântica)

Screen reader é um software que lê o conteúdo de um código HTML, convertendo o texto e permitindo que o usuário ouça o conteúdo. Por isso é importante utilizar sempre as tags corretas para cada parte do documento, dando a eles significado e não apenas aparência.