

## 0.1 Aula 01

Quando se trata de navegadores é importante destacar que, um navegador sempre irá identificar o HTML (expressão inglesa para Hypertext Markup Language) como a única linguagem de exibição de conteúdo. O HTML surgiu como uma linguagem de marcação desenvolvida por Tim Berners-Lee (físico Britânico) com o objetivo de servir como ponte de comunicação e disseminação das pesquisas entre ele e seu grupo de colegas <sup>1</sup>. Isto também só foi possível graças ao trabalho de Tim e Robert Cailliau no desenvolvimento do que viria a ser o protocolo HTTP (expressão inglesa Hypertext Transfer Protocol), principalmente na utilização do esboço da "internet pública" na época, que mais tarde viria ser chamada apenas de "internet".

Se requisitar-mos qualquer texto sem nenhum tipo de formatação, o navegador irá entender, porém, poderá não exibir caracteres acentuados corretamente, assim como, não exibirá quebras de linhas. Para que possamos exibir as informações desejadas com a formatação, é necessário que cada trecho de texto tenha uma **marcação** indicando qual é o significado dele. A seguir segue um exemplo de um trecho de texto sendo formatado em HTML:

```
1
2 <!DOCTYPE html>
3 <html>
4   <head>
5     <title>Pink Floyd</title>
6     <meta charset="utf-8"/>
7   </head>
8   <body>
9     <h1>Brain Damage</h1>
10    <p>The lunatic is on the grass</p>
11  </body>
12 </html>
```

Se você criar um arquivo **index.html** e copiar/colar o trecho de código acima e logo após executá-lo em qualquer navegador, você perceberá como o resultado será exibido de maneira agradável e legível. Para isso, utilizamos algumas marcações HTML. Essas marcações são chamadas de **tags**, e elas basicamente dão **significado** ao texto contido entre sua abertura e fechamento. Vale ressaltar que, o HTML foi criado com o intuito de servir de apoio para trabalhos científicos, logo, as informações marcadas não apresentam nenhum atrativo estético e, nessa deficiência do HTML, reside um dos maiores desafios do programador front-end.

---

<sup>1</sup>Fonte: <https://pt.wikipedia.org/wiki/HTML>

## 0.2 Sintaxe do HTML

Como dito anteriormente, o HTML é um conjunto de **tags** que servem para marcação do conteúdo de uma página no navegador. No código exibido anteriormente, vimos que as tags são os elementos a mais que escrevemos usando a sintaxe `<tag>`. O HTML possui uma vasta coleção de tags e cada uma possui uma funcionalidade específica. Você pode acessar o seguinte endereço e ter uma noção das tags disponíveis, assim como os navegadores que dão suporte atualmente: <https://www.w3schools.com/html/default.asp>.

Muitas tags possuem conteúdo, como o texto do título "Brain Damage". O conteúdo começa com a abertura da tag `<h1>` e acaba com o fechamento da mesma em `</h1>`. É importante ressaltar que, quando uma tag não possui texto em seu corpo, é normal abreviar-mos os fechamento da mesma, como por exemplo, a tag para exibir uma imagem. Geralmente utilizamos ``. Algumas tags também podem receber **atributos** dentro de sua definição. São parâmetros usando a sintaxe de "nome=valor"(você pode notar o atributo `src` utilizado na tag `<img>`).

## 0.3 Estrutura de um documento HTML

Um documento HTML válido precisa seguir obrigatoriamente a estrutura composta pelas tags `<html>`, `<head>` e `<body>`:

```
1 |  
2 | <html>  
3 |   <head>  
4 |   </head>  
5 |   <body>  
6 |   </body>  
7 | </html>
```

A tag `<html>` é a raiz do script. `<head>` e `<body>` são tags consideradas "irmãs" dentro dessa hierarquia, onde são filhas da tag `<html>`. A tag `<head>` contém informações sobre o documento que serão apenas utilizadas internamente pelo navegador. Porém, é necessário a declaração da tag `<title>` (título do documento) dentro da mesma. 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, chamado de **encoding** ou **charset**. Pode-se configurar o tipo de codificação que iremos utilizar, sendo mais comum hoje em dia a utilização do "UTF-8" ou *Unicode*. Já a tag `<body>` contém o corpo do documento, que é exibido pelo navegador. É o con-

teúdo que enxergamos quando visitamos qualquer website. É necessário que o `<body>` tenha ao menos um elemento filho definido em seu escopo, ou seja, uma ou mais tags HTML. Como por exemplo no trecho:

```
1
2 <body>
3   <h1>Brain Damage</h1>
4   <p>The lunatic is on the grass</p>
5 </body>
```

Uma instrução importante também é a primeira definição do documento: `<!DOCTYPE html>`. Esta não é uma tag HTML, mas uma instrução especial que indica para o navegador qual a *versão do HTML* deve ser utilizada para renderizar a página. Atualmente o HTML está em sua versão 5.2.

## 0.4 Tags HTML

Como dito anteriormente, o HTML é composto por diversas tags, cada uma com sua função e significado. o HTML 5, então, adicionou novas tags, que serão vistas no decorrer das aulas. Citarei aqui algumas tags básicas importantes. É importante sempre ter em mente procurar pela documentação para se ter um nível de detalhamento maior acerca dos atributos que cada tag possa fornecer.

### 0.4.1 Títulos

Quando queremos indicar que um texto é um título em nossa página, utilizamos as tags de heading em sua marcação. As tags de heading são tags de conteúdo que seguem uma ordem de importância, sendo `<h1>` o título principal, e `<h6>` o título de menor importância. A ordem de importância, além de influenciar no tamanho padrão de exibição do texto, tem impacto nas ferramentas que processam HTML (como Google, por exemplo).

```
1
2 <!DOCTYPE html>
3 <html>
4   <head>
5     <title>Titulos</title>
6     <meta charset="utf-8">
7   </head>
8   <body>
9     <h1>Titulo 1</h1>
10    <h2>Titulo 2</h2>
```

```

11     <h3>Titulo 3</h3>
12     <h4>Titulo 4</h4>
13     <h5>Titulo 5</h5>
14     <h6>Titulo 6</h6>
15 </body>
16 </html>

```

## 0.4.2 Parágrafos

Como o próprio nome diz, essa tag tem como objetivo dar sentido a um texto escrito para ser processado em um navegador. A melhor forma de representar um texto é utilizando a tag `<p>`. Por exemplo:

```

1
2 <body>
3   <p>The lunatic is on the grass</p>
4   <p>Remembering games and daisy chains and laughs</p>
5 </body>

```

A inclusão da tag `<p>` adiciona automaticamente a quebra de linha entre os textos. Se você não queira utilizar `<p>`, pode-se utilizar a tag `<br>` para a quebra de linha de um texto. Entretanto, é recomendável que se evite este tipo de escrita, pois não é nada legível e pode dar uma tremenda dor de cabeça em estruturas mais complexas.

## 0.4.3 Marcações de Ênfase

Quando queremos dar uma ênfase diferente a um trecho de texto, podemos utilizar as marcações de ênfase. Podemos deixar um texto "mais forte" com a tag `<strong>` ou deixar o texto com uma "ênfase acentuada" com a tag `<em>`. Também há a tag `<small>`, que diminui o tamanho do texto. Por padrão, os navegadores renderizarão o texto dentro da tag `<strong>` em negrito e o texto dentro da tag `<em>` em itálico. Existem ainda as tags `<b>` e `<i>`, que atingem o mesmo resultado visualmente, mas as tags `<strong>` e `<em>` são mais indicadas por definirem nossa intenção de significado ao conteúdo, mais do que uma simples indicação visual.

## 0.5 Imagens

Como dito anteriormente, a tag `<img>` define uma imagem em uma página HTML e necessita de dois atributos preenchidos: `src` e `alt`. O primeiro atri-

buto é obrigatório e aponta para o local da imagem e o segundo, um texto alternativo para a imagem caso essa não possa ser carregada ou visualizada. Leitores do Google, por exemplo, utilizam robôs que lêem o conteúdo deste campo. É important destacar que o campo *src* aceita tanto um caminho para uma pasta local no computador, quanto uma url de um website externo para renderizar a imagem desejada. O HTML5 introduziu duas novas tags para a imagem: `<figure>` e `<figcaption>`. A primeira tag define uma imagem com a conhecida tag `<img>`. Além disso, permite adicionar uma legenda para imagem através da tag `<figcaption>`. Por exemplo:

```
1 |
2 | <figure>
3 |   
4 |   <figcaption>Caminho da imagem :P</figcaption>
5 | </figure>
```

## 0.6 Listas HTML

O HTML possui algumas tags definidas para que seja possível utilizar listagens em nossas páginas. Existem três tipos de listas suportadas atualmente:

### 0.6.1 Listas de Definição

Estas listas, geralmente, são utilizadas para demarcar um glossário, quando listamos termos e seus significados:

```
1 |
2 | <dl>
3 |   <dt>Aula 1</dt>
4 |   <dd>HTML</dd>
5 |   <dt>Aula 2</dt>
6 |   <dd>CSS</dd>
7 | </dl>
```

### 0.6.2 Listas Não-Ordenadas

A lista ordenada não segue uma ordem ou padrão comum. O desenvolver é livre para escolher a melhor maneira de representar os dados.

```
1 |
2 | <ul>
3 |   <li>Joao</li>
```

```
4 <li>Maria</li>
5 <li>Jose</li>
6 </ul>
```

### 0.6.3 Listas Ordenadas

Como o próprio nome já diz, são listas que seguem uma ordem. Por padrão, a ordem seguida é a numérica, mas nada impede de definir um tipo diferente utilizando a propriedade "type". Entretanto, esta propriedade será removida da próxima versão do HTML. Para modificar o tipo da lista ordenada será necessário utilizar estilização do CSS. Porém, só iremos ver isso mais para frente. O exemplo a seguir mostra uma lista ordenada com o tipo definido pelo alfabeto maiúsculo. Note também, que é possível definir listas aninhadas como no exemplo abaixo, onde definio uma lista ordenada, e internamente uma lista não ordenada.

```
1
2 <ol type="A">
3   <li>Acucar</li>
4   <li>Agua</li>
5   <li>Cafe</li>
6   <ul>
7     <li>Descafeinado</li>
8     <li>Expresso</li>
9   </ul>
10 </ol>
```

## 0.7 Exercício

Agora vamos aplicar o conhecimento acima na construção de um exercício simples. Este exercício tem objetivo a criação do conteúdo da parte "Sobre" de um website. Como ponto de partida, disponibilizei um wireframe que pode ser utilizado com base. A ideia é aplicar os conceitos sem se preocupar com a formatação.