**FUNDAÇÃO ESCOLA TÉCNICA LIBERATO SALZANO VIEIRA DA CUNHA**

**curso técnico de informática para internet**

**júlio césar rodrigues de oliveira**

**nº 3 turma: 12331**

**introdução a front end**

**Disciplina: Tópicos Especiais**

**Professor: Raul Gerhardt**

**Novo Hamburgo, 29 de outubro de 2022**

**júlio césar rodrigues de oliveira**

**introdução a front end**

Trabalho de Tópicos Especiais apresentado ao Curso de Técnico de Informática Para Internet da Fundação Escola Técnica Liberato Salzano Vieira da Cunha.

Professor: Raul Gerhardt

**Novo Hamburgo, 29 de outubro de 2022**

Sumário

[1 introdução 4](#_Toc118719790)

[2 html 5](#_Toc118719791)

[2.1 História do HTML 5](#_Toc118719792)

[2.2 Estrutura do HTML 6](#_Toc118719793)

[2.2.1 Tags estruturais 6](#_Toc118719794)

[2.2.2 Tags textuais 7](#_Toc118719795)

[2.2.3 Tags de links 8](#_Toc118719796)

[2.2.4 Tags de mídias 9](#_Toc118719797)

[2.2.5 Outras tags comuns 10](#_Toc118719798)

[3 CSS 11](#_Toc118719799)

[3.1 História do CSS 11](#_Toc118719800)

[3.2 Seletores 11](#_Toc118719801)

[3.2.1 Seletores de elementos 12](#_Toc118719802)

[3.2.2 Seletores de classes 12](#_Toc118719803)

[3.2.3 Seletores de Ids 12](#_Toc118719804)

[3.2.4 Seletor universal 13](#_Toc118719805)

[3.2.5 Seletor de atributos 13](#_Toc118719806)

[3.2.6 Combinadores equivalentes adjacentes 14](#_Toc118719807)

[3.2.7 Combinadores equivalentes gerais 15](#_Toc118719808)

[3.2.8 Combinadores filhos 15](#_Toc118719809)

[3.2.9 Combinadores descendentes 15](#_Toc118719810)

# introdução

*Front-end* é a interface gráfica, a parte visível de uma aplicação, onde os usuários interagem com o sistema, ou seja, a “parte da frente” e está presente na grande maioria das aplicações, sejam elas aplicativos *mobiles*[[1]](#endnote-1), *websites*[[2]](#endnote-2), jogos, sistema do caixa eletrônico, e entre diversos outros tipos.

Este trabalho tem como objetivo introduzir os leitores à programação *front-end* e suas principais linguagens e ferramentas, a fim de proporcioná-los conhecimentos o suficiente para desenvolver e realizar seus próprios projetos e *webpages*[[3]](#endnote-3).

Será abordado conceitos básicos, com o auxílio de exemplos e imagens, acerca das linguagens HTML, CSS, JavaScript e sobre a biblioteca do JavaScript, jQuery. Além disso, será apresentado o conceito de programação orientada a objetos e a linguagem de programação TypeScript, a qual utiliza desta estrutura.

# html

O HTML (*HyperText Markup* Language) é uma linguagem de marcação de Hipertexto usado para criar *webpages*, sites e acessar documentos na web. É no HTML que se define o conjunto de especificações (símbolos) que determinam a formatação do texto o qual o *browser*[[4]](#endnote-4)interpretará. HTML é a linguagem de programação mais utilizada para construir páginas na internet.

## História do HTML

Criado no CERN (*European Council for Nuclear* *Research*) na suíça em 1991, por Tim Berners-Lee, foi inicialmente projetado para interligar instituições de pesquisa próximas e facilitar o compartilhamento de documentos. Em 1992, foi liberada a biblioteca de desenvolvimento WWW (*World Wide Web*), uma rede de alcance mundial.

Em 1995 surgiu a primeira versão após a criação do HTML puro, a versão 2.0. Foi apresentada no primeiro evento da *World Wide Web Conference,* um evento mundial sobre a web. O objetivo da versão 2.0 era formalizar todas as alterações e modificações ocorridas anteriormente, desde a criação do HTML.

A versão 3.2 foi lançada em 1997. Essa versão corrigiu diversos problemas de compatibilidade existentes na versão 2.0, além de trazer novas funcionalidades ao HTML. Ela foi lançada pelo grupo W3C, uma comunidade internacional que trabalham juntos para desenvolver padrões e boas práticas para a WWW. O HTML 3.2 implementou características como tabelas, applets e texto flutuante ao redor de imagens.

Em 1999, a versão 4.0.1 foi lançada trazendo a compatibilidade com as suas versões anteriores, utilizando as implementações de *scrict*, *transitional* e *framesets*. *Strict* proibia utilização de elementos obsoletos, *transitional* permite a utilização de elementos obsoletos e *framesets* é a implementação direcionada para site que usam *frames*[[5]](#endnote-5). A versão 4.0.1 também acrescentou suporte para opções de multimídias e folhas de estilos e trouxe melhorias nas práticas de programação.

Pouco tempo após incrementar a versão 4.0.1, no ano de 2000, foi lançada a primeira versão do XHTML. Ela combinava as vantagens do HTML com as do XML, trazendo uma sintaxe mais rigorosa e menos ambígua, tornando o HTML mais simples de ser processado.

Em 2014 surgiu a versão do HTML 5. Ela trouxe suporte de áudio e vídeo em alto nível que, o que antes era necessário usos de *plug-ins*[[6]](#endnote-6) para ser possível. Introduziu novas *tags*[[7]](#endnote-7) como <header>, <footer>, <section>, <article> e outros, gerando uma forma semântica de se trabalhar com a linguagem. A nova versão também facilitou a reutilização de diversas informações e funcionalidades, diminuindo a necessidades de *plug-ins*, deixando os *sites* mais leves aumentando o desempenho dos sites.

## Estrutura do HTML

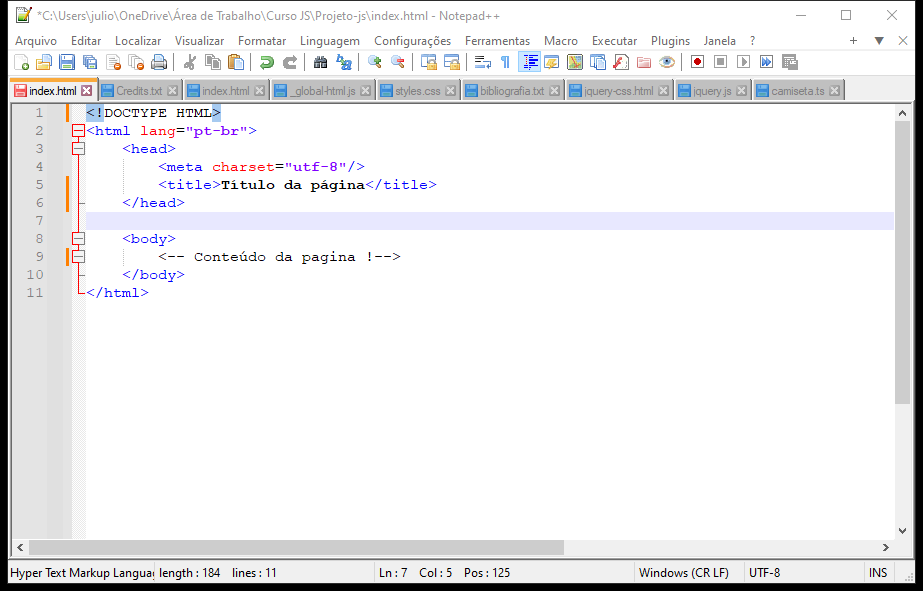
O documento HTML possui uma estrutura base, na qual sempre é utilizada ao iniciar a construção do código HTML. Sempre se inicia com a linha “<!DOCTYPE HTML>”, depois vem a tag[[8]](#endnote-8) <html> e dentro dela as tags <head> e <body>. O doctype é uma instrução aos navegadores e outros aplicativos que os indicam que o documento a seguir é HTML. Ressaltando que este item é obrigatório e deve ser sempre a primeira linha do código.

Já o <html> é a tag raiz do código, que define todo o documento HTML.

Em seguida, a tag <head> contém informações do documento que não são apresentadas visualmente ao usuário. São dados de uso e controle do documento, como vinculação com outros arquivos, aplicação de lógica de programação de scripts e metadados.

Na tag <body> contém tudo que se deseja que o usuário tenha acesso, sejam eles textos, mídias (imagens, áudios e vídeos), links, formulários etc.

Figura 1 - Estrutura base do HTML.



Fonte: o autor

### Tags estruturais

As tags estruturais possuem uma grande importância na questão semântica da sua página e ajudam a estruturar seu código para uma melhor visualização por parte do navegador e do usuário. Além disso, o uso correto dessas tags são de grande importância para a otimização dos processos de SEO (*Search Engine Optimization*). Uma alta classificação do SEO fará seu site aparecer mais alto nas buscas de sites de procura, como o Google, o que ocasionará em mais acessos ao site.

A tag <header> define um cabeçalho. Contém elementos que representam um grupo de suporte introdutório ou navegacional. Podem apresentar também outros elementos como um logo, seções de cabeçalho, formulário de pesquisa, entre outros.

Na tag <main> estará o conteúdo principal da página. Deve incluir o conteúdo diretamente relacionado com o tópico central da página ou com a funcionalidade principal da aplicação.

A tag <footer> define o rodapé da página. O rodapé é o conteúdo final de uma página ou de uma seção da página Podem conter informações de contato, informações de um autor ao final de um artigo, entre outros.

A tag <section> separam a página por seções. Usadas principalmente para estruturar e organizar o documento agrupando de forma lógica o conteúdo quando outros tags não se encaixam. Tem como objetivo substituir o uso da tag <div> na estruturação do documento.

Nas tags <article> definem-se um artigo da página. Utilizadas principalmente por blogs, sites de notícias e páginas de conteúdo. Um artigo deve fazer sentido por si só, sendo capaz de ser compreendido e distribuído fora da página.

A tag <aside> representa uma seção de uma página cujo conteúdo é indiretamente relacionado ao conteúdo do seu entorno. Frequentemente utilizado nas *sidebars[[9]](#endnote-9)* de um documento.

A tag <nav> é usada para estruturar o conteúdo de navegação. Bastante utilizado em conjunto com listas e na criação de menus.

A tag <div> é um container genérico para conteúdo de fluxo, que de certa forma não representa nada semanticamente. Era utilizado anteriormente a versão HTML 5 no lugar das tags acima. Geralmente utilizado para organizar melhor o conteúdo de subseções.

### Tags textuais

As tags <h1> até <h6> são tags de títulos e possuem valor hierárquico maior para menor, respectivamente. Elas possuem valor semântico, logo o bom uso delas afetam também a classificação no SEO.

Para os parágrafos, utilizamos a tag <p>. Também é possível fazer o uso da tag <pre>, a diferença desta é que o texto será pré-formatado, ou seja, será interpretado pelo navegador exatamente como foi escrito no arquivo HTML. Diferente da tag <p>, que retira espaços múltiplos e quebra de linhas, mostrando todo o texto de forma sequencial.

**Figura 2 – Exemplo diferença de <p> e <pre>.**

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Fonte: o autor

A tag <span> é geralmente inserida em textos que deseja dar uma formatação ou interação especiais. Podendo até mesmo ser inserido no meio de um parágrafo.

As tags <b> e <i> transformam os textos, respectivamente, em negrito e itálico. Já as tag <br/> insere uma quebra de linha e <hr/> uma linha horizontal, usada para separar o conteúdo.

### Tags de links

Com a tag <script> é possível incluir códigos de scripts ao seu HTML. Pode ser incluída em qualquer parte do documento, no entanto, recomenda-se por questões de boas práticas que sejam inseridos dentro da tag <head>, fornecendo o caminho do arquivo do script e evitar escrever um script direto no arquivo HTML.

A tag <link> é usada para estabelecer relacionamentos do documento atual com outros arquivos e recursos externos. Bastante usadas para fazer conexões com arquivos CSS ou adicionar favicon, uma imagem pequena que fica no canto esquerdo da aba do navegador.

Já a tag <a> cria um hyperlink, que redireciona para uma outra página ou seção da página atual. Normalmente utilizadas em textos ou imagens, esse link está visível e acessável ao usuário dentro do escopo <body>, diferente da tag anterior <link>, a qual fica dentro do escopo <head> e é inacessível ao usuário.

### Tags de mídias

A tag <img> serve para incluir imagens na sua página HTML. Dentro do atributo “src” é indicado o caminho que encontra-se a imagem, É muito importante informar o atributo “alt”. O texto inserido neste atributo serve tanto para substituir a imagem, caso ela não carregue, quanto para descrever a imagem aos leitores de telas para deficientes visuais. Possui também grande valor para a avaliação do SEO.

A tag <vídeo> e <áudio> servem para indicar a inserção de um vídeo ou audio. Devemos colocar alguns atributos dentro da tag para poder ativar o controle, controls sendo a principal delas. Dentro da tag <vídeo> ou <áudio>, incluímos a tag <source> com os atributos src e type, onde informaremos o local do arquivo e o tipo deste. Por questões de compatibilidade com o navegador, recomenda-se utilizar mais de um tipo de source. No final, deve-se escrever uma mensagem, que será exibida somente caso o navegador não suporte os formatos dos arquivos.

**Figura 3 – Tags <vídeo> e <áudio>**

Texto, Carta

Descrição gerada automaticamente

Fonte: homehost.com.br

### Outras tags comuns

A tag <style> é onde define-se os estilos de formatação da página. Nela é possível incluir códigos CSS (Cascading Style Sheets) dentro do arquivo HTML, sem necessitar de um arquivo CSS externo.

A tag <ol> cria uma lista ordenada. Listas ordenadas têm seus itens mostrados com uma contagem que os precede. Não precisam ser necessariamente números, podem ser letras, algoritmos romanos e até mesmo símbolos.

A tag <ul> cria uma lista não-ordenada. Diferente da lista ordenada, a lista não ordenada não possui contagem.

A tag <li> insere um nas listas ordenadas e não ordenadas. Esta tag é inserida dentro das tags <ol> e <ul>.

A tag <form> cria um formulário. Formulários são usados para receber e processar dados inseridos pelo usuário.

A tag <input> insere de campos dentro formulário. Podem ser do tipo texto, número, email, data, senha, botão e entre outros tipos.

A tag <label> define um rótulo para o campo correspondente do formulário.

# CSS

O CSS (*Cascading Style Sheet,* ou folha de estilos em cascatas) tem como finalidade estilizar elementos de uma linguagem de marcação como o HTML. O CSS é responsável pela formatação do estilo e layout da página. Nele é possível alterar diversas propriedades visais, tais como a cor do fundo e da fonte, espaçamentos de margens e *paddings* (margem interna), tipo e tamanho da fonte, bordas e entre várias outras propriedades. Até mesmo editar efeitos como *hover* e *focus,* que são, respectivamente, quando o mouse ou o cursor estiver sobre o elemento selecionado.

Entre as principais vantagens do uso do CSS estão: possibilidade do uso de um arquivo de estilo CSS externo, diminuindo a quantidade do código HTML facilitando a compreensão do mesmo. Reutilização de código, formatando mais de um elemento com seletores e classes. Diminuição do risco de erros. Facilidade nas integrações de estilos pré-formatados.

## História do CSS

Em outubro de 1994 a criação de uma linguagem de estilo foi proposta por Hakon Lie, a fim de facilitar a programação de sites. Originalmente o HTML não foi projetado para ter tags que ajudariam a formatar a página, e mesmo tendo novas tags introduzidas na versão HTML 3.2 com este propósito, o código ficava muito complexo. Outra complicação era a necessidade de ter que reescrever o código de HTML de versões anteriores a 3.2 para a nova.

Em 1995 a proposta foi apresentada à W3C, que tiveram grande interesse no projeto e resolveu criar uma equipe para tal. Liderados por Håkon e Bert Bos, o projeto rendeu resultados logo no ano seguinte, lançando a versão CSS1 no ano de 1996.

Dois anos depois, no dia 12 de maio de 1998, foi lançada a recomendação do CSS de nível 2 (CSS2) e em 2012 saiu o CSS3, trazendo diversas novidades com ela.

## Seletores

Um seletor CSS é a primeira parte de uma regra CSS. Eles servem para informar ao navegador quais elementos HTML devem ser selecionados para que os valores de propriedade CSS dentro da regra sejam aplicados a eles. Há uma grande variedade de seletores CSS disponíveis, permitindo uma precisão refinada ao selecionar os elementos a serem estilizados.

### Seletores de elementos

Este tipo de seletor tem a finalidade de selecionar elementos HTML e aplicar estilos a todos os elementos presentes daquele tipo dentro do documento.

|  |
| --- |
| article {  font-family: “verdana”, “arial”;  font-size: 18px;  } |

No exemplo acima, o CSS está selecionando todos os elementos que possuem a tag <article> e estiliza o tipo e tamanho da fonte.

### Seletores de classes

Seletores de classes, como o nome diz, selecionam elementos cujo possuem determinada classe. Bastante utilizados para reutilizar estilos que se repetem, podendo aplicá-los a determinados elementos sem sobrescrever o estilo da tag. Para selecionar uma classe no CSS, deve escrever “.” (ponto) e o nome da classe.

|  |
| --- |
| .fundo-preto {  background: “black”;  color: “white”;  } |

Neste exemplo, todos os elementos que possuírem a classe “fundo-preto”, terão o seu *background* (fundo) alterado para preto e sua fonte para branco.

### Seletores de Ids

Seletores de ids tem o papel de selecionar elementos que possuem um id específico. Um id não pode estar presente em mais de um elemento. É utilizado o símbolo “#” (cerquilha) seguido pelo nome do id para selecioná-lo.

|  |
| --- |
| #nav-bar {  background: #5783eb;  margin: 0px 20px;  } |

Aqui, neste exemplo, o elemento que possui a id “nav-bar” está sendo estilizado para ter uma margem de zero pixels[[10]](#endnote-10) nas margens verticais e 25 pixels nas margens horizontais e tendo seu fundo alterado para uma cor selecionada através do valor hexadecimal equivalente.

### Seletor universal

O seletor universal aplica estilo para elementos de qualquer tipo, através do símbolo “\*” (asterisco). A partir do CSS3, o seletor universal pode ser combinado com *namespaces*. Ele é opcional nos seletores simples logo, \*.fundo-preto e . fundo-preto são equivalentes.

|  |
| --- |
| \* {  font-family: “helvética”, “times”;  } |

O exemplo anterior altera a fonte de todos os elementos para a fonte helvética ou, caso o navegador não a suporte, para a fonte times.

### Seletor de atributos

Este seletor seleciona elementos que possuem um determinado atributo com o valor especificado. Os exemplos abaixo foram retirados do site developer.mozilla.org.

|  |
| --- |
| /\* <a> elementos com um atributo de título \*/  a[title] {  color: purple;  }  /\* <a> elementos combinando com um href "https://example.org" \*/  a[href="https://example.org"] {  color: green;  } |
| /\* <a> elementos com um href contendo "example" \*/  a[href\*="example"] {  font-size: 2em;  }  /\* <a> elementos com href terminando em ".org" \*/  a[href$=".org"] {  font-style: italic;  }  /\* Links internos, começando com "#" \*/  a[href^="#"] {  background-color: gold;  }  /\* Todas divs onde Chinese está em vermelho (red), simplificado para (zh-CN) ou tradicional (zh-TW). [attr|=value] representa um elemento com um atributo de nome attr o qual o valor pode ser exatamente value ou pode começar com value imediatamente seguido por hífen -\*/  div[lang|="zh"] { color: red; } |

### Combinadores equivalentes adjacentes

O combinador equivalente adjacente, representado pelo símbolo “+” (mais), separa dois seletores e seleciona o segundo elemento somente se ele segue imediatamente o primeiro elemento e se ambos forem filhos do mesmo pai (elemento).

|  |
| --- |
| img + span {  font-size: 16px;  font-style: italic;  } |

O exemplo acima seleciona todos os elementos “span” que vem logo depois de um elemento “img”. Este exemplo poderia servir para formatar legendas da imagem.

### Combinadores equivalentes gerais

O combinador equivalente geral, ou combinador geral de irmãos, é representado pelo símbolo “~” e funciona de forma semelhante ao combinador equivalente adjacente, com a diferença que ele seleciona o segundo elemento se ele o primeiro elemento o antecede, indiferente de ser imediatamente ou não. Ambos devem ser filhos do mesmo pai.

|  |
| --- |
| h3 ~ p {  margin-left: 20px;  } |

O combinador acima seleciona todos os parágrafos (elemento “p”) que aparecem depois dos títulos de elemento “h3”.

### Combinadores filhos

O combinador filho, representado pelo símbolo “>”, é inserido entre dois seletores e seleciona somente os elementos em que o segundo elemento é filho direto do primeiro elemento, ou seja, não seleciona elementos “netos”.

|  |
| --- |
| article > p {  font-family: “courier”, “arial”;  } |

### Combinadores descendentes

O combinador descendente - normalmente representado por um caractere de espaço simples (" "), é inserido entre dois seletores e seleciona os elementos em que o segundo elemento é descendente do primeiro elemento, ou seja, pode selecionar elementos filhos, “netos” e por assim adiante.

|  |
| --- |
| aside img {  width: 80%;  } |

1. Aplicações de *smartphones.* [↑](#endnote-ref-1)
2. Conjunto de páginas da internet. [↑](#endnote-ref-2)
3. Páginas da internet. [↑](#endnote-ref-3)
4. Programa utilizado para acessar páginas da internet. [↑](#endnote-ref-4)
5. Extensão do HTML que permite divisões na janela do browser. [↑](#endnote-ref-5)
6. Complementos para personalizar programas e aplicativos. [↑](#endnote-ref-6)
7. Elementos do HTML. [↑](#endnote-ref-7)
8. Instruções da linguagem HTML. [↑](#endnote-ref-8)
9. Área lateral em um site. [↑](#endnote-ref-9)
10. Menor unidade de imagem digital. [↑](#endnote-ref-10)