HTML5

# Elemento HTML

Diagrama

Descrição gerada automaticamente

<!DOCTYPE html> Apesar de parecer um elemento HTML ela apenas diz ao navegador que ele está lidando com um arquivo do tipo HTML5.

<html> A tag html é a raiz do seu documento, todos os elementos HTML devem estar dentro dela. E nela nós

informamos ao navegador qual é o idioma desse nosso documento, através do atributo lang, para o português

brasileiro usamos pt-BR.

<head> A tag head contém elementos que serão lidos pelo navegador, como os metadados - um exemplo é o

charset, que é a codificação de caracteres e a mais comum é a UTF-8, o JavaScript com a tag script,

o CSS através das tags style e link - veremos a diferença quando falarmos sobre CSS - e o título da

página com a tag title.

<meta charset="utf-8">

<title>Curso HTML5</title>

</head>

<body> E dentro da tag body colocamos todo o conteúdo visível ao usuário: textos, imagens, vídeos.

<h1>Hello World!!</h1>

</body>

</html>

# Semântica

<section>: Representa uma seção genérica de conteúdo quando não houver um elemento mais específico para isso. Ex.:

<header>: É o cabeçalho da página ou de uma seção da página e normalmente contém logotipos, menus, campos de busca.

<article>: Representa um conteúdo independente e de maior relevância dentro de uma página, como um post de blog, uma notícia em uma barra lateral ou um bloco de comentários. Um article pode conter outros elementos, como header, cabeçalhos, parágrafos e imagens.

* <aside>: É uma seção que engloba conteúdos relacionados ao conteúdo principal, como artigos relacionados, biografia do autor e publicidade. Normalmente são representadas como barras laterais.
* <footer>: Esse elemento representa o rodapé do conteúdo ou de parte dele, pois ele é aceito dentro de vários elementos, como article e section e até do body. Exemplos de conteúdo de um <footer> são informações de autor e links relacionados.

# Textos e Links

* <h1>-<h6>: Eles não foram criados na versão 5 do HTML e nem são específicos para semântica, mas servem para esse propósito. São utilizados para marcar a importância dos títulos, sendo <h1> o mais importante e <h6> o menos. Uma dica: use apenas um <h1> por página, pois ele representa o objetivo da sua página.
* <p>: representa um parágrafo, mas ele não suporta apenas texto, podemos adicionar imagens, código, vídeos e vários outros tipos de conteúdo dentro dele.
* <a>: que significa anchor/âncora, ele representa um hyperlink, é ele que interliga vários conteúdos e páginas na web. Ex.: <a href="mailto:lucas@vilaboim.com" target="\_blank">lucas@vilaboim.com</a> (Atributos: href -> Link da página; target -> Quando "\_blank" o link será aberto em outra aba.).

# Listas

<ul>: ul cria uma lista não ordenada, onde a ordem dos elementos não é importante, e é representada com pontos, círculos ou quadrados.

<ol>: serve para criar lista ordenadas, nessas a ordem importa, portanto elas são representadas com números, algarismos romanos ou letras.

<li>: Elemento contido por <ul> ou <ol>. É um item dentro de uma dessas listas. Pode conter vários tipos de conteúdos, como parágrafos, imagens e até outras listas.

# Imagens

* <img>: Elemento sem tag de fechamento. O elemento src é obrigatório e guarda o caminho para a imagem que você quer mostrar na página. O alt não é obrigatório mas é altamente recomendado por melhorar a acessibilidade, ele mostra a descrição da imagem caso ela não carregue e leitores de tela usam esse atributo para descrever a imagem para o usuário saber o que ela significa. Ex.: <img class="post\_img" src="https://picsum.photos/300/350" alt="The Lorem Ipsum for photos.">

CSS3

# Definição de Seletores

A sintaxe é bem simples e pode ser explicada com a frase "você cria regras de estilo para elementos ou grupos de elementos".

Texto

Descrição gerada automaticamente

No arquivo HTML, adicionar a tag: <link rel="stylesheet" href="style.css">

# ID x Classes

ID: é representado pelo símbolo # (hash) seguido de um nome para esse ID.

Classe: a classe é representada de forma parecida do ID, mas é precedida por um ponto em vez do hash.

* E a diferença mais importante entre eles é a forma como devem ser usados: o ID só pode ser usado uma vez em uma página HTML enquanto a classe não tem restrições.

# Box-model

Quando estamos criando o layout de um site o navegador representa cada elemento HTML como uma caixa retangular, isso é o box-model. E com CSS nós alteramos a aparência dessa caixa (largura, altura, cor de fundo etc.). Essa caixa é composta por 4 áreas: o conteúdo, o padding, a borda e a margem.

As margens (margin) são espaçamentos entre elementos;

As bordas (border) ;

O padding é um espaçamento entre as bordas e o conteúdo, a diferença para as margens é que declarações de imagem de fundo funcionam nele;

O conteúdo (content) é o que o seu bloco representa, um texto, uma imagem, um vídeo;

Diagrama

Descrição gerada automaticamente com confiança baixa

# Padding e Margin

Atribuir um valor ao Padding/Margin:

Texto

Descrição gerada automaticamente com confiança média

Alterando a parte inferior e superior:

Texto

Descrição gerada automaticamente

Alterando a parte superior, direita, inferior e esquerda (nessa ordem):

Texto

Descrição gerada automaticamente

Utilizando uma propriedade pra cada lado:

Texto

Descrição gerada automaticamente

# Background

E aqui temos 3 formas de colocar uma cor de fundo, e ainda existem outras. A primeira é pelo nome da cor em inglês, a segunda é pelo código hexadecimal e a terceira é usando apenas o atalho background.

Texto

Descrição gerada automaticamente

# Border

Vimos que a propriedade border pode ter 3 valores: a largura, a cor e o estilo, mas existem algumas particularidades nisso.

Texto

Descrição gerada automaticamente

É possível estilizar cada lado:

Texto

Descrição gerada automaticamente

Utilizando as propriedades específicas:

Texto

Descrição gerada automaticamente

Border-radius:

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

# Textos:

font-family: Com o font-family podemos alterar a fonte dos nossos textos, como uma fonte da internet ou uma que esteja instalada no nosso computador, mas vamos nos ater às fontes seguras, chamadas de web safe fonts (Verdana, Geneva, Tahoma, sans-serif). Essas fontes são chamadas assim pois são encontradas em quase todos os sistemas e podem ser usadas sem preocupação.

* font-size: O font-size nos ajuda a mudar o tamanho do texto, existem algumas unidades de medida para ele mas por enquanto os pixels são suficientes para nós.
* font-style: Usamos o font-style para tornar um texto itálico, na maioria das vezes você usará apenas o valor italic para ele, mas se precisar tirar o itálico de um texto você pode usar o valor normal.
* Text-transform: Valores que podem ser atribuídos uppercase (todas as letras maíusculas); lowercase (todas as letras minúsculas); capitalize (primeira letra de cada palavra minúscula).
* Text-decoration: underline (linha abaixo); overline (linha acima); line-through (linha ao centro);

# Dimensão e Alinhamento

* Width:
* Height:
* Max-Width:
* Max-height:
* Margin:
* Text align:

BOOTSTRAP

# Introdução

Framework para sites responsivos

* Biblioteca de componentes
* <https://getbootstrap.com/>
* Segue o seguinte padrão de classes: .container (div), .row (div) e .col (div)
* Possível dividir até 12 partes

# Para utilizar o framework

Em <https://getbootstrap.com/docs/5.1/getting-started/download/> baixar Compiled CSS and JS.

* Fazer o link na tag head <link rel="stylesheet" type="text/css" href="bootstrap/css/bootstrap.min.css">
* Em [https://getbootstrap.com/docs/5.1/components/](https://getbootstrap.com/docs/5.1/components/buttons/) é possível encontrar componentes já prontos e para utiliza-los, basta incluir a tag HTML.

Interface gráfica do usuário, Site

Descrição gerada automaticamente