**Curso de HTML5 e CSS3 – Guanabara – Módulo 1**

**Capítulo 1**

Reposittórios Públicos

github.com/gustavoguanabara

gustavoguanabara.github.io

A Internet veio da ARPANET que originalmente protegia centros militares.

**Capítulo 2**

MOdulação

DEModulação

MODEM

UTF-8?

TLD – top level domain

GTLD – genérico = .com .edu . gov

CCTLD – country code

www. = subdomínio

/… = caminho

http = HyperText Transfer Protocol

https = HyperText Transfer Protocol Secure

**Capítulo 3**

HTML – conteúdo (texto, imagens, vídeos, tabelas)

CSS – design (cores, sombras, tamanhos, posicionamento)

JavaScript – interações (menus, animações, popups, validações)

HTML – src = parâmetro / foto.png = valor

CSS – h1 = seletor / font-family: Arial; = declaração / font-family = propriedade / Arial = valor

\*identação – organização da estrutura

**Capítulo 4**

Estrutura básica de documento HTML

<!DOCTYPE html>

<html lang=”pt-br”>

<head>

<meta charset=”UTF-8”>

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

<title>Document</title>

</head>

<body>

<h1>Olá, Mundo!</h1>

</body>

</html>

<head> = configurações do site, app...

<body> = conteúdo

<hr> = horizontal row

**Capítulo 5**

<br> = break row

&lt; = < less than (sinal)

&gt; = > greater than (sinal)

Para emoji no html:

Iniciar com &#x e finalizar com ;

**Capítulo 6**

JPEG = compressão, compactação.

Guanagabra disse que 1500px de largura para imagem em site já é um tamanho bom (usar em fundo, por exemplo). 650px pode ser para usar no conteúdo.

Três maneiras de colocar imagens, pasta local, subpastas e link de outro site.

Favicon, ícones que ficam nas abas dos navegadores.

Precisa estar em formato ico

iconarchive.com = tem um banco de ícones

favicon.cc = criar esses ícones

favicon.io = você pode fazer favicons de texto, emojis ou converter imagens (png)

Guanabara recomendou que o arquivo escolhido para o Favicon seja nomeado de favicon.ico.

**Capítulo 7**

H1 = heading

**Capítulo 8**

HTML5 foca em significado (semântica)

<b> = não semântica

**<strong> = semântica**

<i> = não semântica

**<em> = semântica**

Comando = COMMAND + SHIFT + P ir em Wrap with abbreviation e digitar a tag que quer para “envelopar” o que vai dentro da tag.

Tags

<mark> = marca o texto em amarelo

<small> = letras pequenas

<del> = risca as palavras, é um texto deletado, pode ser desconsiderado pelo leitor

<ins> = é como se fosse inserir em lacunas, foi inserido e requer atenção do leitor

<sup> = sobrescrito

<sub> = subscrito

Tags

<pre> = mantém a formatação, pré-formatação. e eu der espaços no código ele vai replicar no navegador também. É bastante utilizada com a tag <code> = <pre><code>

<q> = quote (citações)

<blockquote> = citação de livros, é possível colocar o parâmetro **cite=””** e colocar o endereço original (fonte). Mais utilizada para parágrafos.

<abbr> = abreviação, para siglas (muito usado em tecnologia). Dentro do parâmetro **title** pode colocar o nome “cheio” => HTML – HyperText Markup Language

<bdo> => inverte o texto, tem a opção ‘l’ e ‘r’ => ‘l’ left to right e ‘r’ right to left

**Capítulo 9**

No HTML5 o </li> se tornou opcional, não é preciso fechar a tag.

Parâmetro da tag <ol>: type = “1, A, a, I, i”.

Posso usar o parâmetro type e escolher o tipo da minha lista. Posso também usar o parâmetro start para começar a partir de um determinado ponto. Lembrando que o start é numérico, se eu quiser que ele comesse da letra “e”, por exemplo, tenho que colocar o número 5.

Para listas não ordenadas tenho as seguintes opções no parâmetro type: disc, circle, square.

Aninhamento, organização.

ALT ou OPTION, no Visual Studio marcar mais de um ponto para edição.

Tags

<dl> = lista de definição

<dt> = termo

<dd> = descrição

**Capítulo 10**

Links externos são recomendados usar o parâmetro target=”\_blank”.

É bom usar também o parâmetro rel=”external”. Diz que é um link tipo externo.

Tags

<Address> = semântica para endereço

Dentro de <a> colocar rel=”nofollow”, para dizer ao mecanismo de busca que você não dá o aval para o conteúdo continuado. Links pagos, por exemplo.

Dentro de <a> colocar rel=”author”, direciona ao site do autor do artigo atual.

O parâmetro de <a>, tarfet=”\_self”, abre os links na mesma aba. Geralmente é assim que funciona.

Para links internos, se você precisa voltar para uma página que está em outra página. Usar ../ e o nome da página.

Em <a>, href=”#”, significa link vazio.

Parâmetro para download de pdfs (exemplo) download=”nomedoarquivo”.

type=”application/pdf”, type=”application/zip”

hreflang=”en”, indica qual é o idioma principal do site para onde o link está levando.

**Capítulo 11**

Tag <picture>, indica que irá aparecer uma imagem.

Bounce rate, pode se considerar como taxa de desistência do usuário. Mede o quanto o usuário sai do seu site. Se o bounce rate for alto o Google não vê isso com bons olhos.

Tag <source media=”(max-width: 1050px)” srcset=”caminho da imagem.png” type=”image/png”>

Nessa tag é possível definir o tamanho máximo de tela para aplicar uma imagem e escolher qual imagem usar caso esse tamanho seja alcançado.

**Usar uns 50px a mais por segurança para não criar uma barra de rolagem.**

**É importante também colocar em primeiro lugar a menor (da menor para a maior) foto a ser carregada, exemplo:**

<source media=”(max-width: 750px)” srcset=”caminho da imagem menor.png” type=”image/png”>

<source media=”(max-width: 1050px)” srcset=”caminho da imagem.png” type=”image/png”>

Lembrando que dentro da tag <picture> deve ir a imagem padrão que se quer carregar.

Tag <audio> para áudios, alguns atributos controls e autoplay.

Nem todos os navegadores executam arquivos .mp3. Por isso, se você quiser ter compatibilidade com todos os navegadores, ter uma versão de arquivos WAV e OGG também.

Para isso usar apenas a tag <audio> e dentro dela colocar o <source src=”nome do arquivo.extensão” type=”ver qual o tipo de acordo com a extensão do arquivo”>. O que estiver em cima, será o primeiro que o navegador irá carregar:

<source src=”nome do arquivo.mp3” type=”áudio/mpeg”>

<source src=”nome do arquivo.wav” type=”áudio/wav”>

<p>Infelizmente o seu navegador não carrega áudio.</p>

Nesse caso .mp3 será carregado primeiro. Assim por diante até possivelmente a tag <p> nesse caso.

Mesmo usando <source>, dentro de <audio> colocar os atributos: controls, autoplay.

Atributo preload=”metadata”, vai carregar apenas informações sobre o arquivo (tamanho, tempo, informações de direito, etc), é o mais indicado.

É possível colocar “auto”, mas só irá carregar depois de ter baixado o arquivo todo e isso é ruim para arquivos grandes. E também é possível “none”, que não carrega até que o usuário dê play ou um script inicie a reprodução.

Formatos de vídeos mais usados no HTML5: mp4(m4v), webm, ovg. mp4 e webm são os mais populares entre os navegadores.

Handbrake, app para conversão de vídeo.

Para colocar vídeo dá para usar a tag <video> e os atributos “src”, outros que podem ser usadas “width” e “controls”.

Porém dá para fazer como as imagens e usar a tag <video>, seguido da tag <source> como nas imagens. O HTML vai priorizar o primeiro que você colocou no código (de cima para baixo).

Atributos importantes de usar “width” e “poster” (coloca uma imagem ao invés do primeiro frame do vídeo). Lembrando que é recomendável reduzir o tamanho do vídeo ao invés de diminuir o tamanho de pixels em width.

Vídeos hospedados dessa maneira, local, pode gerar problemas e gastos maiores. Eles consomem muito tráfego de dados.

Parâmetro loop também pode ir dentro da tag <video>.

O recomendado é usar vídeos em código de incorporação, YouTube, Vimeo, etc.

Vimeo tem uma ótima qualidade, porém o algoritmo não é tão bom quanto o YouTube. O vídeo do Vimeo pode dar umas travadinhas se a internet de quem está assistindo não for muito boa.

**Capítulo 12**

CSS interno ou local.

Seletor = body

Dica para criar uma página Visual Studio Code, dentro da tag <a> cmd + click e ir em criar arquivo.

Usar dentro da tag <head> do HTML:

<link rel=”stylesheet” href=”style.css”>

Regras no CSS começam com @

Usar @charset “UTF-8”; - isso ajuda a não ter problemas com os caracteres, acentos etc.

É possível usar mais de uma tag link nas páginas.

Ordem dos poderes do estilo:

1º inline

2º interno

3º externo (porém o mais versátil)

Evite usar o inline a todo custo, só se for para uma coisa muito específica mesmo.

Estilo interno, apenas para estilos pequenos. Se não fica muita informação visual no arquivo.

Externo é o mais indicado, principalmente caso tenha mais de uma página.

Comentários em CSS, ficam entre /\* \*/.