Endereço do repositório público GitHub

<https://github.com/gustavoguanabara>

<https://gustavoguanabara.github.io>

Links de ajuda:

<https://developer.mozilla.org/pt-BR>

<https://www.w3.org/standards/webdesign/>

<https://html.spec.whatwg.org/>

<https://www.w3schools.com/>

<https://www.pexels.com/pt-br/videos/> (Vídeos grátis)

Link do documentário:

[https://youtu.be/TNQsmPf24go](https://www.youtube.com/watch?v=TNQsmPf24go)

**Código binário**

É composto por 0 e 1, basicamente são onda elétricas que são enviadas, porém em conjuntos de 8 bits (ex: 01000001 = letra A), e a esse conjunto é dado no nome de byte.

O computador só entende ondas binárias.

Diferença entre MB e Mb

MB significa MEGABYTES, é usado como medida de armazenamento

Mb significa MEGABITS, ou seja, é uma unidade bem menor e é usado como medida de transmissão.

**Como o computador encontra um site?**

Através do DNS, onde ficam os domínios dos sites, ou o nome dos sites, por exemplo, o instagram.com, como o IP do site está em constante mudança a localização é feita através da busca do domínio no servidor DNS, o modem envia a solicitação para o servidor DNS, que manda o IP atual de volta e assim ele faz o acesso do servidor do Instagram.

**Código UTF-8?**

**Domínio e hospedagem**

Para que outras pessoas possam ver o meu site eu tenho que ter um domínio que nada mais é a identificação da minha URL, como no exemplo da aula, o github.io, esse domínio é enviado ao servidor DNS, também tenho que hospedar o meu site em um servidor que é onde ele vai ficar guardado com segurança. Em resumo, domínio é o nome do meu site e hospedagem é onde o meu site vai ficar armazenado.

Linha do tempo

Descrição gerada automaticamente com confiança média

Existem dois tipos de TLD, a GTLD e a ccTLD.

**GTLD**: São TLDs genéricos, sem indicação de país. Alguns dos domínios genéricos são .com, .net, .gov, .org, .io, .info, .online, .store, etc.

**ccTLD**: São TLDs com designação do país (coutry code). Alguns dos domínios desse tipo são .com.br, .edu.us, .co.fr, .jp, .es, etc.

Exercício de fixação:

gustavoguanabara.github.io, é uma **URL** tem um caminho: **gustavoguanabara,** um domínio: **github.io** e uma GTLD: .**io**

**Obs:** Errei o caminho, que nesse caso é um subdomínio

**Como funciona a HTML e as CSS**

Primeiro, HTML e CSS não são linguagens de programação, o correto é dizer que eu desenvolvo coisas em HTML, as duas são linguagens, mas não de programação.

**HTML** = **H**yper**T**ex **M**arkup **L**anguage

É uma linguagem focada em conteúdo, pode ser imagem, texto, vídeos, tabelas etc.

**CSS** = **C**ascading **S**tyle **S**heets

São atrativos visuais com foco em design como cor, posição da imagem, texto escrito por cima da imagem, sombras etc.

Para declarar um estilo em uma TAG, por exemplo em um título:

Usa-se a TAG sem os DEV (**<>**), abrimos e fechamos chaves (**{ }**), isso se chama **seletor** e dentro dela colocamos linhas com os comandos que são chamadas de declaração, e todas as declarações têm que ter ponto e vírgula no final (**;**).

Diagrama

Descrição gerada automaticamente

**\*\*Após terminar o curso de HTML e CSS, fazer o de Javascript\*\***

Mas para adiantar, Javascript, ou JS como também é conhecido, é usado na parte de interação do site, com ele é possível criar menus interativos, criar interações para clicar no vídeo e ele carregar e muito mais.

**Estrutura básica de um documento em HTML**

Para dizer ao site que eu estou desenvolvendo o site em HTML5 usamos a declaração

<!DOCTYPE html>

<html lang=”pt-br”>

<head>

<meta charset=”UTF-8”> “Codificação voltada para acentuação e outras coisas”

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

<title>Document</tilte>

“Cabeça” – Área de configurações

</head>

<body>

“Corpo” – Onde fica o conteúdo do site

</body>

</html>

Diagrama

Descrição gerada automaticamente

O navegador que no caso é o Chrome vai analisar linha a linha do código e trazer uma resposta visual na tela com o site desenvolvido.

**Front-end, Back-end e Full Stack**

HTML, CSS e JS são tecnologias Client-side, ou seja, tecnologias para o lado do cliente e o mercado define quem desenvolve esse tipo de tecnologia como desenvolvedor Front-end, é quem vai dar entregar uma experiência visual ao cliente.

Para desenvolvimento em servidores o mercado define como Back-end, está mais focado na interação do código com o servidor.

As linguagens usadas são: PHP, JS, C#, PHYTON entre outras

Full Stack é o profissional que domina o Front-end e o Back-end

**Parágrafo e quebra de linha**

**&lt;** = <, **p&gt;** = >, ou seja, esses códigos com & comercial simbolizam os dev <> em forma de texto, quando usado para exemplificar uma TAG sem que ela gere um comando ela vai aparecer assim no texto da página <br>, ou <p> ou dentro da TAG que eu quiser,

**Emoji e caracteres especiais**

Acessar o site emojipedia e buscar pelo que eu quero, para colocar o emoji no meu código tenho que usar &#x antes do código do emoji para que o HTML entenda que é em hexadecimal, a junção fica assim &#x + o código do emoji, por exemplo &#x1F596; (Dica: não esquecer do ponto e vírgula)

Obs: alguns caracteres especiais dão erro de exibição, é só retirar o “x” que ele funciona 😉

**Direitos autorais de imagem**

Sempre pesquisar se a imagem tem direitos autorais.

**Para inserir o caminho da imagem de forma prática é só digitar img e enter que o VScode já abre a TAG, depois é só apertar CRTL + Espaço e ele busca a imagem que está salva no mesmo local do arquivo index, se estiver em outra pasta é só usar o mesmo comando, e para inserir imagem de um site é só buscar a imagem, abrir ela e em seguida clicar em copiar endereço da imagem e colar dentro da TAG img.**

**Favicon**

São os ícones dos sites que são exibidos na guia junto ao nome





Para colocar um Favicon no site, basta baixar o arquivo com a extensão .ico, no VScode antes de title digitar link, escolher a opção Favicon, depois , depois CTRL + Espaço e escolher o arquivo.

**H1 é o título principal**

**H2 e o subtítulo**

Para inserir por exemplo um efeito itálico em um texto que eu já escrevi basta selecionar o trecho ou palavra, apertar CTRL+Shift+P e escrever Abb para buscar por Wrap with abbreviation, digitar i e apertar enter que a TAG vai ser inserida no trecho que eu selecionei,

Mas também posso fazer isso com qualquer TAG, pois esse comando é um atalho, posso fazer mais rápido selecionando o trecho, apertando CTRL+Shift+P, enter, digito a TAG e enter novamente, com isso não tenho que escrever a TAG, depois copiar o trecho, colar dentro da TAG, pois é muito trabalhoso, e dessa forma economizo tempo,

**TAGs Importantes**

Strong – **Negrito**

Em – *Itálico*

Mark – Deixa o texto marcado assim (Posso escolher a cor depois com CSS)

Del – ~~Deixa o texto tachado~~

Ins – Deixa o texto sublinhado

Sup – Deixa o texto sobrescrito X20+3

Sub – Deixa o texto subscrito H2O

\*\*Essas TAGs são novas pois o HTML5 é semântico, então as TAGs tem que ter significado

Pre – Faz com o trecho do código fique na formatação de como foi escrito, é ideal para quando queremos descrever um código de outra linguagem, e pode ser usado com a TAG code que mostra o trecho com o estilo abaixo.

Code – Usado para demonstrar trechos de códigos e pode ser usada com a TAG PRE, ficando assim:

num = int(input('Digite um número'))

if num % 2 = 0:

print(f'O número {num} é PAR')

else

print(f'O número {num} é ÍMPAR)

print('Fim do programa')

Q – Usado para citação, basicamente coloca aspas no texto selecionado

Blockquote – Usado para citação direta, o texto aparece recuado a direita

Abbr – Usado para marcar siglas e exibir o significado dela quando passamos o mouse em cima da sigla.

Marquee – Faz o texto ficar passando da direita pra esquerda ou vice – versa.

**Listas**

Para adicionar listas é só usar as TAGs

Ol – Para listas ordenadas

Ul – Para listas não ordenadas

Em ambas as TAGs usamos a TAG li que quer dizer Item da lista

Podemos também usar listas aninhadas com vários níveis, mas para facilitar é mais fácil fazer a estrutura com os tópicos principais primeiro e depois ir criando os subníveis para não confundir, assim:

<h1>Lista dos meus jogos favoritos</h1>

<ul>

    <li>Playstation 4</li>

    <ol>

        <li>THE LAST OF US</li>

        <ul>

            <li>THE LAST OF US I</li>

            <li>THE LAST OF US II</li>

        </ul>

    </ol>

    <li>Playstation 3</li>

    <ol>

        <li>GOD OF WAR</li>

        <ul>

            <li>GOD OF WAR III</li>

        </ul>

        <li>UNCHARTED 4</li>

        <ul>

            <li>UNCHARTED 4 - A THIEF'S END</li>

        </ul>

    </ol>

    <li>Playstation 2</li>

    <ol>

        <li>GTA</li>

        <ul>

            <li>GTA SANDREAS</li>

            <li>GTA VICE CITY STORIES</li>

        </ul>

        <li>GOD OF WAR</li>

        <ul>

            <li>GOD OF WAR II</li>

        </ul>

    </ol>

    <li>Playstation 1</li>

    <ol type="1" start="4">

        <li>CRASH</li>

        <ul>

            <li>CRASH BARNDIDCOT 1</li>

            <li>CRASH BARNDIDCOT 2</li>

        </ul>

    </ol>

</ul>

</body>

</html>

**Lista de Definições**

Usa as TAGs:

Dl – Definição da lista

Dt – Definição do termo

Dd – Descrição da definição

Fica assim:

    <h2>Listas de Definições</h2>

    <dl>

        <dt>HTML</dt>

        <dd>Linguagem de marcação para a criação do conteúdo de um site.</dd>

        <dt>CSS</dt>

        <dd>Linguagem de marcação para criação do design de um site.</dd>

        <dt>JavaScript</dt>

        <dd>Linguagem de programação para a criação da interatividade de um site.</dd>

    </dl>

E sai assim no navegador:

Texto

Descrição gerada automaticamente

\*\*Pesquisar SEO depois\*\*

**Trabalhando com Links**

A TAG utilizada para referenciar links é:

A – Ancora, que deve ser usada com o atributo href, ao colocar a TAG a no VScode o href já é preenchido automaticamente, por padrão o direcionamento é feito para que o link abra na mesma guia, mas podemos mudar isso com os atributos Target e rel, Target tem alguma opções, mas vamos usar a \_blank com o underline, isso faz com que o site a ser acessado pelo link seja aberto em uma nova guiam e o rel é mais um suporte ao navegador que informa que o link é externo.