HTML dicas e TAGs

TECLAS DE ATALHO

**Ctr + spaço :** ajuda a indicar comandos que possa tilizar

Envelopamento: **Ctr shift P**, envelopamento de código para acrescentar tag,

Ganhar tempo quando tem que colocar TAG em um conteúdo grande ou que não tem TAG

TAGs

Inserir comentário <!-- ex: <!—descrição-->

H1 Titulo de primeiro nivel: <h1> </h1>

H2 subtitulo, indo ate o H9 <h2> </h2>

P paragragos, para a inserir textos: <p>

HR linha horizontal: <hr>

BR quebra de linha: <br>

Maior Q > : &lt;

Menor Q <: &gt;

CODE Ideal para códigos Tag: Code <code>texto</code>

Citações como fosse entre aspas dupla : Tag q <q>texto</q>

Citações grandes, parágrafos: Tag blockquote <blockquote>texto</blockquote>

Abreviações abreviar uma palavra ou texto: Tag abbr<abbr title=”insira o texto a ser abreviado ”>texto</abbr>

SIMBOLOS

Espaço em branco:  &nbsp;

® Marca registrada: &reg;

© Copyright: &copy;

™ Trade mark: &trade;

€ Euro: &euro;

£ Libra (pound): &pound;

¥ Yen: &yen;

¢ Cent: &cent;

∅ Vazio: &empty;

∑ Soma: &sum;

Δ Delta: &Delta;

← Seta esquerda: &larr;

↑ Seta acima: &uarr;

→ Seta direita: &rarr;

↓ Seta baixo: dar;

EMOJIS

Site de emojis: <https://emojipedia.org/>

Sempre começa com &#x: e o código sempre depois do U+

EX: &#x1F596; [🖖](https://emojipedia.org/emoji/%F0%9F%96%96/)

Imagens

CENTRALIZAR IMAGEM:

Tem que transformar a imagem em bloco

Display: block;

E depois utilizar o margin: auto.

Ex:

Display: block;

Margin: auto;

Ex código:

display: block;

    margin: auto;

Tag: img pressione enter e já cria automaticamente:

<img src=”” alt=””>

Img: imagem

Src:(Sorce ou caminho)

Alt(Texto alternativo “legenda da imagem”, importante preencher)

<img src=”” alt=””>

Dentro das aspas dupla, de src pressione (control + espaço)

Aparecerá as imagens contidas na pasta do projeto.

Imagens externas: copie o enedereço da imagem e cole dentro do src=https://bognarjunior.files.wordpress.com/2018/01/1crcyaithv7aiqh1z93v99q.png “.

Negrito / Destaque

Negrito tag com semântica: <Strong> negrito </Strong>

Itálico / Ênfase

Tag com semantica: em <em> itálico</em>

Marcador de texto (marca texto)

Tag: mark <mark>texto </mark>

Texto Grande e Texto Pequeno

Texto grande Tag: big <big>texto</big>

Texto pequeno Tag: small <small>texto</small>

Texto deletado ou riscado

Texto deletado: Tag del <del>texto</del>

Texto Inserido

Texto Inserido: Tag ins <ins>texto</ins>

Sobrescrito

Texto subscrito: Tag sup <sup>texto</sup>

Subscrito

Texto subscrito: Tag sub <sub>texto</sub>

TRECHOS DE CÓDIGOS

Para inserir trechos de códigos no HTML, usa a tag <code> ‘ex de código’</code>

Assim fica formatado como uma linha de código.

Ex

<code>

dl>

        <dt>Parágrafo</dt>

        <dd>p</dd>

        <li>Títulos</li>

        <li>Imagens</li>

        <li>Abraviações</li>

        <li>Códigos-fonte</li>

        <li>Listas numerádas</li>

        <li>Listas com demarcações</li>

    </dl>

</code>

CITAÇÕES

Para criar uma citação em HTML, podemos usar a tag **<q>** (do Inglês quote, que significa citar). O texto que estiver entre **<q>** e **</q>**já vai receber automaticamente as aspas, mas não terá nenhum deslocamento. Essa técnica é mais usada quando queremos uma citação no meio de um parágrafo.

Também podemos criar citações mais longas (em bloco) e que tenham um parágrafo só para si. Nesse caso, colocaremos tudo dentro de **<blockquote>** e **</blockquote>**

e o texto ganha um recuo automaticamente. Podemos também colocar um link para o texto original, usando o parâmetro cite dentro da tag.

Ex:

<blockquote cite=”https://www.curosoemvideo.com”>

O curso de JS é para criação e estilização de conteúdo

</blockquote>

ABREVIAÇÕES

Essa é uma novidade da HTML5 e que ajuda muito em áreas como a de Tecnologia, que usa muitas siglas e abreviações. Sempre que você quiser escrever uma sigla, mas deixar claro ao usuário (e aos mecanismos de busca) o significado dela, use a **tag <abbr>.**

Ex:

<p> Eu estou estudando <abbr title=”Hyper text markup language”>HTML</abbr> para a criação de sites. </p>

LISTAS

Listas ordenadas

**Tag OL** (“Orderne LIst(lista ordenada)”)

<ol>

Ex: **Tag li** (list item(‘’item da lista’’)) <li> Acordar</li>

<li> escover os dentes</li>

<li> tomar cafe</li>

</ol>

Ex em código:

<body>

    <h1>Trabalhando com listas</h1>

    <h2>Listas Ordenadas</h2>

    <ol>

        <li>Acordas</li>

        <LI>Tomar café</LI>

        <LI>Escovar os dentes</LI>

        <li>Ligar para o joão </li>

    </ol>

</body>

Personalizar a lista:

Em numerações começando do: 1

Ex:

Código no site

|  |  |
| --- | --- |
| <ol type=”1”>  <li> acordar</li>  <li> tomar café</li>  </ol> | 1. Acordar  2.Tomar café |

Para alterar o início da contagem:

Ex:

Código no site

|  |  |
| --- | --- |
| <ol type=”1” star”4”>  <li> acordar</li>  <li> tomar café</li>  </ol> | 4. Acordar  5.Tomar café  ... |

Contagem em numeral ROMANOS maiúsculo ‘I’

Código no site

|  |  |
| --- | --- |
| <ol type=”I”>  <li> acordar</li>  <li> tomar café</li>  </ol> | I. Acordar  II.Tomar café  ... |

Contagem em numeral ROMANOS maiúsculo ‘i’

Código no site

|  |  |
| --- | --- |
| <ol type=”i”>  <li> acordar</li>  <li> tomar café</li>  </ol> | i. Acordar  ii.Tomar café  ... |

Contagem em alfabeto minusculo‘a’

Código no site

|  |  |
| --- | --- |
| <ol type=”a”>  <li> acordar</li>  <li> tomar café</li>  </ol> | a. Acordar  b.Tomar café  ... |

Contagem em alfabeto maiúsculo ‘A’

Código no site

|  |  |
| --- | --- |
| <ol type=”A”>  <li> acordar</li>  <li> tomar café</li>  </ol> | A. Acordar  B.Tomar café  ... |

FIM

Listas não ordenadas

ul(“unorder List (listas não ordenadas)”)

Tag: <ul>

<li>pão</li>

<li>leite</li>

<ul>

<body>

<h2>Listas não Orenadas</h2>

    <ul>

        <li>pão</li>

        <li>tomate</li>

        <li>arroz</li>

        <li>feijão</li>

    </ul>

</body>

Código no site

|  |  |
| --- | --- |
| <ul>  <li> pão</li>  <li> leite</li>  </ul> | pão  leite  ... |

Personalizar a lista:

Dentro do type ” ”

São as:

Disc: bolinha cheia

Cicle: circulo

Square: quadrado

<ul type=”circle”>

Ex:

Código no site

|  |  |
| --- | --- |
| <ol type=”square”>  <li> acordar</li>  <li> tomar café</li>  </ol> | Acordar   * acordar * Tomar cafe |

Fim

Juntando listas / Aninhando Listas

<h2>Minhas linguagens favoritas</h2>

    <ol>

        <li>Antigas</li>

        <ol type="a">

            <li>Cliper</li>

            <li>Fortran</li>

            <li>fortran</li>

            <li>delphi</li>

        </ol>

        <li>Novas</li>

        <ol type="a" start="5">

            <li>PHP</li>

            <li>Python</li>

            <li>Java Script</li>

            <li>Kotlin</li>

        </ol>

    </ol>

    <h2>Meus jogos favoritos</h2>

    <ol>

        <li>NES</li>

        <ul type="square">

            <li>Mario Bros</li>

            <ul type="circle">

                <li>Mario Bros 3</li>

                <li>Mario: Lost Levels</li>

            </ul>

            <li>Ninja Gaiden</li>

        </ul>

        <li>SNES</li>

        <ul type="square">

            <li>Donkey Kong</li>

            <li>Mario</li>

        </ul>

        <li>PlayStation</li>

        <ul type="square">

            <li>Final Fantasy</li>

            <li>Castelvania</li>

        </ul>

    </ol>

Listas de Definição

Lista de definição serve para criar uma lista com titulo e a descrição do título, como fosse um dicionário

dl tag (list definition”Lista de definição”)<dl></dl>

dd ta (termo”titulo”)<dt></dt>

dt tag (Descrição do titulo) <dd></dd>

ex:

<h2>Lista de definição</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 a criação do desing de um site</dd>

       <dt>Java Script</dt>

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

    </dl>

Algumas CSS em listas

**Colocando emogi nas listas**

**List-style-type: ‘\codigo do emogi (só o número)’;**

Ex:

**List-style-type: ‘\2714’;**

Ex código:

list-style: '\2714';

Mesma coisa para inserir símbolo de link, porem coloca uma classe nos links.

a.externo::after{

    content: '\1f517';

}

**Posição das listas:**

**List-style-postion:** posição da lista

**List-style-position: inside;** posição da lista dentro.

Ex código:

list-style-position: inside;

**Divisão de colunas**

**Columns:** qtd de colunas.

Ex código:

columns: 2;

ex em CSS:

aside ul{

   list-style-position: inside;

   columns: 2;

}

LINKS EXTERNOS

Para criar um hyperlink, devemos criar âncoras através da tag . O principal atributo dessa tag é o href, que cria uma referência hipertexto. Vamos ver um exemplo simples:

Tag<a href=”link”>o texto âncorado</a>

<a href=”https://gustavoganabara.github.io”>acesse o perfil do github></a>

Note que dentro do atributo href, o que colocamos foi uma URL completa para outro

site.

Ex:

<body>

    <h1>usando links externos</h1>

    <p>Você pode acessar o repositorio <a href="https://gustavoguanabara.github.io" target="\_blank" rel="external">público do curso em vídeo no GitHub</a></p>

</body>

Fim

LINKS INTERNOS

Para criar um link interno, tendo outras páginas já criadas

Ex <h2>Links Internos</h2>

<p>acessando <a href ”link da página” target”\_self” rel”next” >segunda página</a></p>

**href**””: é onde colocar o link dentro de aspas

**target**: significa alvo, para onde quer levar levar ou manter o link : **\_self:** para manter na mesma página **\_blank:** levar para uma nova janela.

**rel:** significa sistema de navegação onde você não deixa a mercê dos buscadores: **rel”next”**: indicando que irá para a próxima página. **rel”prev”**: indicando que vai voltar a página.

**rel”nofllow”:** pedir para o buscador não seguir, não dar o aval para seguir

ex:

<p>Esta é a primeira página do site se quiser, se você quiser, pode acessar também a mnha <a href="pag002.html" rel="next">segunda página</a></p>

Pagina em outra pasta:

Para acessar uma página em outra pasta, deve colocar o novo caminho com o nome da pasta x e nome da página

Ex:

<p>Você também pode acessar nossa <a href="noticias/pag003.html" target="\_self" rel="next">página de noticías</a></a>

E para voltar a página você deve colocar utilizar o comando **../nome da pagina**

**Significado comando: ../ :** comando para voltar uma pasta para trás.

**Ex:**

    <p>Você pode voltar para a <a href="../index.html" rel="prev">página principal</a></p>

</body>

LINKS PARA DOWNLOAD

Incluindo links para download:

Ex

Para se fazer teste antes de incluir links, usas o “ # ” para deixar funcionando porem o link vazio, igual ao exemplo abaixo

ex

<a href”#” </a>

Para incluir o link deve-se colocar o caminho/pasta e o nome do arquivo.

ex

<a>href” pasta onde o arquivo esta/nome do arquivo”</a>

<li><a href="LIVROS/02 - Como funciona a Internet.pdf" download<a/>

Parâmetros para conseguir efetuar o download dos arquivos:

Download =” nome da pasta . tipo do arquivo”

Ex: <li><a href="LIVROS/02 - Como funciona a Internet.pdf" download="LIVROS.pdf"<a/>

Type=”application/e o tipo de arquivo”

Ex: <a href="LIVROS/02 - Como funciona a Internet.pdf" download="LIVROS.pdf" type="application/pdf">

Abaixo temos um exemplo completo de como deve ficar para funcionar ARQUIVO EM PDF e arquivo compactado ZIP

Ex:

<H1>linkis para download</H1>

        <ul>

            <li><a href="LIVROS/02 - Como funciona a Internet.pdf" download="LIVROS.pdf" type="application/pdf">Baixar livor em PDF</a></li>

            <li><a href="LIVROS/xboxonehdd-master-9.zip"

download="LIVROS.zip" type="application/zip">Baixar arquivo em ZIP</a></li>

        </ul>

Fim.

Imagens Flexíveis

Nosso primeiro passo no caminho de adaptar nosso conteúdo ao tamanho da tela vai ser aprender a gerar imagens de tamanho diferentes e a fazer o navegador carregar a imagem certa para cada situação. Para isso, devemos conhecer as **tags** **<source>** e **</source>**

Para esse exemplo, criamos as três imagens ao lado: a menor tem **300x300px**, a média tem **700x700px** e a maior tem **1000x1000px.**

**Ex:**

**<picture>**

**<img src=”imagem-G.png” alt=”**A imagem flexível **”>**

**</picture>**

a tag **<img>** para inserir a maior imagem **imafe-G.png** que foi a de 1000px.

O problema vai começar a surgir quando a janela do navegador chegar perto dos 1000 pixels de largura, pois a foto não vai mais caber lá. Para resolver:

Ex:

**<picture>**

**<source media=”max-width 1050px” srcset=”imagem-M.png” type”imagem/png”>**

**<img src=”imagem-G.png” alt=”**A imagem flexível funciona**”>**

**<picture>**

Note que a tag **<source>**possui três atributos:

‣ type vai indicar o media type da imagem que usamos (veja mais informações sobre media types no capítulo 10)

‣ srcset vai configurar o nome da imagem que será carregada quando o tamanho indicado for atingido

‣ media indica o tamanho máximo a ser considerado para carregar a imagem indicada no atributo srcset.

Agora, recarregue seu código e mude o tamanho da janela do navegador. Você vai perceber que a imagem muda automaticamente conforme aumentamos ou diminuímos o tamanho da tela. Vamos continuar e acrescentar mais um **<source>** à nossa imagem:

<picture>

<source media= ”(max-width: 750px)” srcset=”imagem-P.png” type “ imagem/png”>

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

<img src=”imagem-G.png” alt=”Imagem Flexível” >

</picture>

É importante que existe uma ordem entre os **<source>**, e nessa nossa configuração, os itens mais acima sejam os menores tamanhos para max-width e que os seguintes sejam maiores, de forma crescente. O último item dentro de **<picture>** deve ser a imagem padrão.

Exemplo código:

<picture>

        <source media="(max-width: 750px)" srcset="imagens/imagem-P.png" type="image/png">

        <source media="(max-width: 1050px)" srcset="imagens/imagem-M.png" type="image/png">

        <img src="imagens/imagem-G.png" alt="Imagem flexível">

    </picture>

Reproduzindo áudio

Para inserir um áudio de forma simples só utilizar a **tag <audio>**

**Ex:**

**<áudio src=”pasta/nome da musica.mp3” e paramentro de controlador (controls)></audio>**

**Ex código:**

**<áudio src=”mídia/zula – minivandals.mp3” controls></audio>**

Só atualizar a página e aparece o controlador de **aúdios**.

Vamos analisar os principais atributos da **tag <audio>** antes de mais nada:

‣ O atributo **preload** indica se o áudio será pré-carregado ou não e aceita três valores: ๏ **metadata** vai carregar apenas as informações sobre o arquivo (tamanho, tempo, informações de direitos, etc)

๏ **none** não vai carregar absolutamente nada até que o usuário clique no botão play ou um script inicie a reprodução

๏ **auto** (padrão) vai carregar o arquivo de áudio inteiro assim que a página for carregada, mesmo que o usuário nunca aperte o play.

‣ O atributo **controls** vai apresentar o player na tela. Caso não seja colocado na **tag <audio>**, o controle será transparente e o usuário não poderá interagir com ele.

‣ O atributo **autoplay**, quando inserido, vai iniciar a reprodução do áudio assim que a página for carregada.

‣ O atributo **loop** vai fazer com que o áudio seja repetido eternamente assim que terminar a sua reprodução.

Dentro da **tag <áudio>**, adicionamos vários **<source>** com formatos diferentes do mesmo áudio. Coloque na parte de cima o seu formato favorito. Os demais só serão carregados caso o de cima falhe. Caso todos falhem, criamos um parágrafo que permite o download do arquivo MP3 para ouvir no player padrão do dispositivo.

**Ex de código:**

<audio controls loop>

        <source src="midia/Zula - Mini Vandals.mp3" type="audio/mpeg">

        <source src="midia/Zula - Mini Vandals.mp3" type="audio/ogg">

        <source src="midia/Zula - Mini Vandals.mp3" type="audio/WAV">

        <p>Infelizmente seu navegador nao consegue reproduzir</p>

        <a href="midia/Zula - Mini Vandals.mp3">Faça o download</a>

    </audio>

Estilos CSS inline

CSS Inline: Significa que o estilo é na mesma linha ou mesma TAG, sendo assim todas as demais TAGs terão que ser feitas uma por uma. **NÃO INDICADO.**

EX os STYLE NAS TAGs:

<!DOCTYPE html>

<html lang="Pt-br">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

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

    <title>desafio 09</title>

</head>

<body style="background-color: black">

    <h1 style="text-align: center; color: aliceblue;" >Vídeos legais para assistir</h1>

    <a href="PageHtml.html"><img src="thumb-html.png" alt="curso HTML 5" style="padding-left: 35%"></a>

    <a href="PageJS.html"><img src="thumb-js.png" alt="curos Java Script"></a>

    <br>

    <a href="PageHardware.html"><img src="thumb-hardware.png" alt="Curso hardware" style="padding-left: 35%;"></a>

    <a href="PagePython.html"><img src="thumb-python.png" alt="Curso de Python"></a>

</body>

</html

CORES

 <!-- Representação por nomes    -->

    <h2 style="background-color: blue; color: white;">Exemplo de cores</h2>

    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Maiores vitae blanditiis tempora dolorum facere placeat quidem voluptatum assumenda magnam laboriosam iure tenetur, nostrum, sint hic, libero porro commodi nemo magni!</p>

    <!--Representação por códiogos em hexa decimais-->

    <!--Decimal: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9-->

    <!--Hexadecimal: 0 1 2 3 4 5 6 7 8 9 A B C D E F-->

    <h2 style="background-color: #0000ff; color: #ffffff;" >Exemplo de cores</h2>

    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Maiores vitae blanditiis tempora dolorum facere placeat quidem voluptatum assumenda magnam laboriosam iure tenetur, nostrum, sint hic, libero porro commodi nemo magni!</p>

    <!-- Representação em RGB(Red, green, Blue)-->

    <h2 style="background-color: rgb(0, 0, 255); color: rgb(255, 255, 255);">Exemplo de cores</h2>

    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Maiores vitae blanditiis tempora dolorum facere placeat quidem voluptatum assumenda magnam laboriosam iure tenetur, nostrum, sint hic, libero porro commodi nemo magni!</p>

    <!--Representação por HSL (hue, saturation, Luminosity) -->

    <h2 style="background-color: hsl(240, 100%, 50%) color hsl(0, 0%, 100%);">Exemplo de cores</h2>

    <p>Lorem ipsum dolor sit, amet consectetur adipisicing elit. Maiores vitae blanditiis tempora dolorum facere placeat quidem voluptatum assumenda magnam laboriosam iure tenetur, nostrum, sint hic, libero porro commodi nemo magni!</p>

GRADIENTE COM CSS

Podemos usar estilos em gradiente com cores em CSS

Deve usar **o background-image:**

O **to** significa a posição onde irá iniciar ex**: to right** (para direita)**, to left** (para esquerda), **to top** (para cima), **to bottom** (para baixo).

Também pode colocar a % do lado da cor, referenciando o tanho que ficará na tela

Backgound-image: linear-gradiente

Ex:

<title>Gradiente em CSS</title>

    <style>

        body{

            background-image: linear-gradient(to right, #F538E5, #D23BFF, #9541E8, #673BFF, #3841F5);

        }

    </style>

</head>

ALGUMAS TAGS DE CORES E ESTILOS CSS

Text-shadow (colocar sombra em textos)

Box-shadow(colocar sombra em caixas boxes)

Text-aligin (posicionamento ou aliamento de texto)

Margin: auto; (posicionar automaticamente sempre no cento )

PSEUDO CLASSES EM CSS

Uma PSEUDO CLASSE é representada com o símbolo: (dois pontos )no CSS, uma PSEUDO CLASSE precisa estar relacionada a um elemento ou classe, (uma PSEUDO CLASSE está relacionada a um estado de um objeto: se ele ta ativo, ablitado, marcado, vazio), você consegue configurar o estado.

EX:

Div:hover{

Background-color: red;

}

HOVER: significa: quando passar o mouse vai ter uma ação tipo: mudar de cor.

OUTRAS PSEUDO CLASSES EM CSS

Dessa vez vamos usar como exemplo em links

Ex:

a:visited {

color: darkred

}

A **pseudoclasse VISITED siginica:** quando um link já tiver sido clicado ele mantem alterado o estado TIPO: aterado a cor para darkred, igual foi setado no exemplo

Ex:

a:visited{

            color: darkred;

        }

A **pseudoclasse houver significa:**  quando passar o mouse ele mudará a ação. No caso vai alterar a cor para preta ao passar o mouse e aparecer uma linha abaixo.

Ex:

A:hover {

Text-decoration: underline;

Color: black;

}

Ex:

 a:hover{

            text-decoration: underline;

            color: black;

        }

A **pseudoclasse active:** esse é a ação ao clicar ele irá alterar apenas no click.

Ex:

a: active: {

color: yelow;

}

Ex:

        a:active{

            color: yellow;

        }

PSEUDO ELEMENTO

Ele pode agir: nas classes, nos elementos, nas Ids, ele meche nos conteúdos periférico.

Para referenciar um PSEUDO ELEMENTO referencia **:: (duas sequências de dois pontos).**

**Ex:**

**a :: after {**

**contente: ‘[link]’;**

**}**

SIMBOLOS QUE PODE UTILIZAR NAS CSS DESTE CAPTULO:

SELETORES PERSONALIZADOS

/\*

            # = id

            . = class

            : = pseudo-classe

            :: pseudo-elemento

            > = children (filho)

        \*/

FLEX BOX (MODELOS DE CAIXA)

Um conjunto que ajuda a organizar, tipo como caixa que pode se aninhar um dentro da outra. Aplicavel HTML e CSS. Todos os documentos em HTML tudo aquilo que é exibido em uma tela é exibido em forma de caixa. Todo elemento visível em um site é uma caixa ate o H1. Toda caixa tem um tamanho referenciado com:

**Box-size:**

**HEIGHT:** altura

**WIDITH**: largura

Height: 300px;

Width: 300px;

Conseguimos colocar contorno na caixa chamado **BORDER.**

conseguimos também colocar um espaço entre a borda e o conteúdo interno chamo de:

**PADDING: (ALCOCHOAMENTO OU PREENCHIMENTO).**

**Padding-top:** topo o parte de cima

**Padding-right:** lado direito

**Padding-bottom:** parte de baixo

**Padding-left:** lado esquerdo

Temos também a

**MARGIN:** espaço externo fora da borda.

**Margin-top:** topo ou parte de cima.

**Margin-right:** lado direito.

**Margin-bottom:** parte de baixo

**Margin-left:** lado esquerdo

**Margin: auto; (Para centralizar o box no meio da tela).**

Conseguimos também colocar uma linha na área da MAGIN, chamado de:  
**OUTLINE: contorno ou traçado.**

**Outline-width: 5px:** largura do outline.

**Outline-style: solid:** estilo do contorno

**Outline-color: red:** cor do contorno do outline.

**TEMOS DOIS TIPOS DE CAIXA**

**BOX LEVEL:** Toda box level ela pula uma linha e cria um caixa que ocupa a largura (**width**). inteira da tela.

**TAGS BOX LEVEL:** <div>, <h1>- <h6>, <p>, <main>, <header>, <nav>, <article>, <aside>, <footer>, <form>, <video>.

**INLINE LEVEL:** Ele não pula linha e cria uma box na mesma linha e só é do tamanho do conteúdo, mesmo sendo na mesma linha.

**TAGS INLINE LEVEL:** <span>, <a>, <code>, <small>, <strong>, <em>, <sup>-<sub>, <label>, <botton>, <input>, <select>.

Exemplos:

 h1{

            background-color: lightgray;

            height: 300px;

            width: 300px;

            border-width: 10px;

            border-style: solid;

            border-color: darkgreen;

            padding-top: 10px;

            padding-right: 10px;

            padding-left: 10px;

            padding-bottom: 10px;

            margin-top: 10px;

            margin-right: 10px;

            margin-bottom: 10px;

            margin-left: 10px;

        }

        a{

            border-width: 10px;

            border-style: solid;

            border-color: red;

            padding-top: 10px;

            padding-right: 10px;

            padding-left: 10px;

            padding-bottom: 10px;

        }

**SIMPLIFICANDO O CODIGO USANDO SHORT HANDS**

**Temos 3 formas de simplificar o código:  
ex:**

**Sem simplificação:**

Border-top: 10px;

Border-right: 10px;

Border-bottom: 10px;

Border-left: 10px;

**Simplificando largura**

**Simplificando acima e mantendo e podendo alterar a largura individual:**

**Ex:**

Border: 10px 10px 10px 10px;

**Simplificando com tamanhos padrões, todos os lados terão a mesma largura:**

Border: 10px;

**Simplificando o código com outras propriedades**

**Ex:**

**Sem simplificação:**

Border-width 10px;

Border-style: solid;

Border-color: red;

**Simplificando:**

Border: tamanho estilo cor; nessa sequencia.

Border: 10px solid red;

GROUPING TAGS OU TAGS DE AGRUPAMENTO

Os group tags são tags semânticas.

**<header>:** Cabeçalho do site

**<nav>:** para criar menus

**<main>**: Conteúdo principal

**<sections>:** seções

**<article>:** artigos

**<aside>:** conteúdo periférico ao artigo, para referenciar um altor.

**<footer>:** rodapé

SOMBRAS

A CSS para a criação de sombra é: **box-shadow**

**Box-shadow:** 1px 1px 1px preto;

**Box-shadow**: **deslocamento horizonta,** **deslocamento vertical,** **espalhamento**, **intensidade** e **sombra(cor**).

**Deslocamento vertical:** para deslocar para **esquerda** basta colocar um tamanho negativo **ex: -10px.** Para deslocar para **direita** basta colocar um tamanho positivo **ex: 10px.**

**Deslocamento horizontal:** para deslocar para **cima** basta colocar um tamanho negativo **ex: -10px.** Para deslocar para **baixo** basta colocar um tamanho positivo **ex: 10px.**

**Espalhamento:** paraaumentar o espalhamento da sombra so aumentar o tamanho **ex: 10px**

**intensidade:** paraaumentar a intensidade da sombra so aumentar o tamanho **ex: 10px.**

**Sombra:** a sombra pode ser feita com qual quer cor **ex: red, sempre indicado colocar uma transparecia na cor.**

Para fazer uma sobra interna basta colocar **box-shadow: inset**

**Ex;**

**Box-shadow:** inset 1px 1px 1px preto;

Indicado uma sobra mais transparente, não trocar de cor, apenas deixar mais transparente.

VÉRTICES OU ARREDONDAMENTOS

Forma de colocar bordas em caixas no geral a **CSS** que se utiliza é **border-radius.**

**Ex:**

**Border-radius:** 10px; todas as bordas arredondadas

**Border-radius:** 10px 0px; cima esquerda e baixo direita.

**Border-radius:** 0px 10px; cima direita, baixo esquerda.

**RESPONSIVIDADE PARA SITES**

A largura mais ideal para os dispositivos grande largura máxima de 1200 px.

Comandos em CSS mais indicado seria

**PARA CONTEÚDO**

Min-width= colocar o tamanho mínimo de celular 320px

Max-width= colocar a largura até os 1200px indicados

Margin: auto= para o conteúdo centralizar na tela

Ex:

Mini-width: 300px;

Max-width: 800px;

Margin: auto;

Ex código:

        main{

            background-color: white;

            padding: 20px;

            border-radius: 10px;

            min-width: 300px;

            max-width: 800px;

            margin: auto;

        }

PARA IMAGEM

Width: 100%.

Assim a imagem irá se adaptar ao tamanho do conteúdo.

Ex código:

 img{

            width: 100%;

        }

Caso tenha outra imagem já reduzida, indicado usar a tag <picture>

Sempre usando o (max-width tamanho)

Ex: (max-width: 600px)

<picture>

            <source media="(max-width: 600px )" srcset="imagens/irina-blok-pq.jpg">

            <img src="imagens/irina-blok.jpg" alt="Irina Blok">

        </picture>

**PARA VÍDEOS**

Para centralizar o vídeo, tem que utilizar uma gambiarra, por conta dele já vir setado largura e altura, então envelopa o **iframe** com uma **<div>** e sempre utilizando uma **class** para edição em **CSS**

Ex código:

<div class="video"><iframe width="560" height="315" src="https://www.youtube.com/embed/l2UDgpLz20M" title="YouTube video player" frameborder="0" allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share" allowfullscreen></iframe></div>

Configurando o CSS do vídeo:

Essa configuração abaixo é para a **div do vídeo:**

div.video{

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

    padding: 20px;

}

E essa configuração de baixo é para o **iframe do vídeo do Guanabara:**

div.video{

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

    padding: 20px;

    padding-bottom: 58%;

    position: relative;

}

div.video > iframe{

    position: absolute;

    top: 5%;

    left: 5%;

    width: 90%;

    height: 90%;

}

Solução que eu achei, mais fácil porem em telas grandes e dependendo do vídeo fica umas bordas pretas na lateral

div.video > iframe{

    width: 100%;

    display: block;

    margin: auto;

}

**GIT E GITHUB**

**Git:** serve para armazenar as versões de um projeto, ajudando em algum erro podendo recuperar ou voltar a versão anterior.

**GitHub:** Servidor para armazenamento e hospedagem dos projetos, portifólio dos projetos criados.