**CURSO DE PROGRAMAÇÃO**

**Curso de HTML5 e CSS3**

Domínio - <https://gustavoguanabara.github.io/>

***HTML5***

{} chaves

< > Colchetes angulares

Gustavoguanabra – sub domínio

Github – domínio

IO – TLD

HTTP’s – protocolo

**HTML – conteúdo, constitui-se de:**

* Textos
* Tabela
* Números
* Imagens
* Vídeos

**CSS – Design, constitui-se de:**

* Cores
* Sombras
* Tamanhos
* Posicionamento

**JAVA SCRIPT – Interatividade, constitui-se de:**

* Menus
* Animações
* Pop ups
* Validações

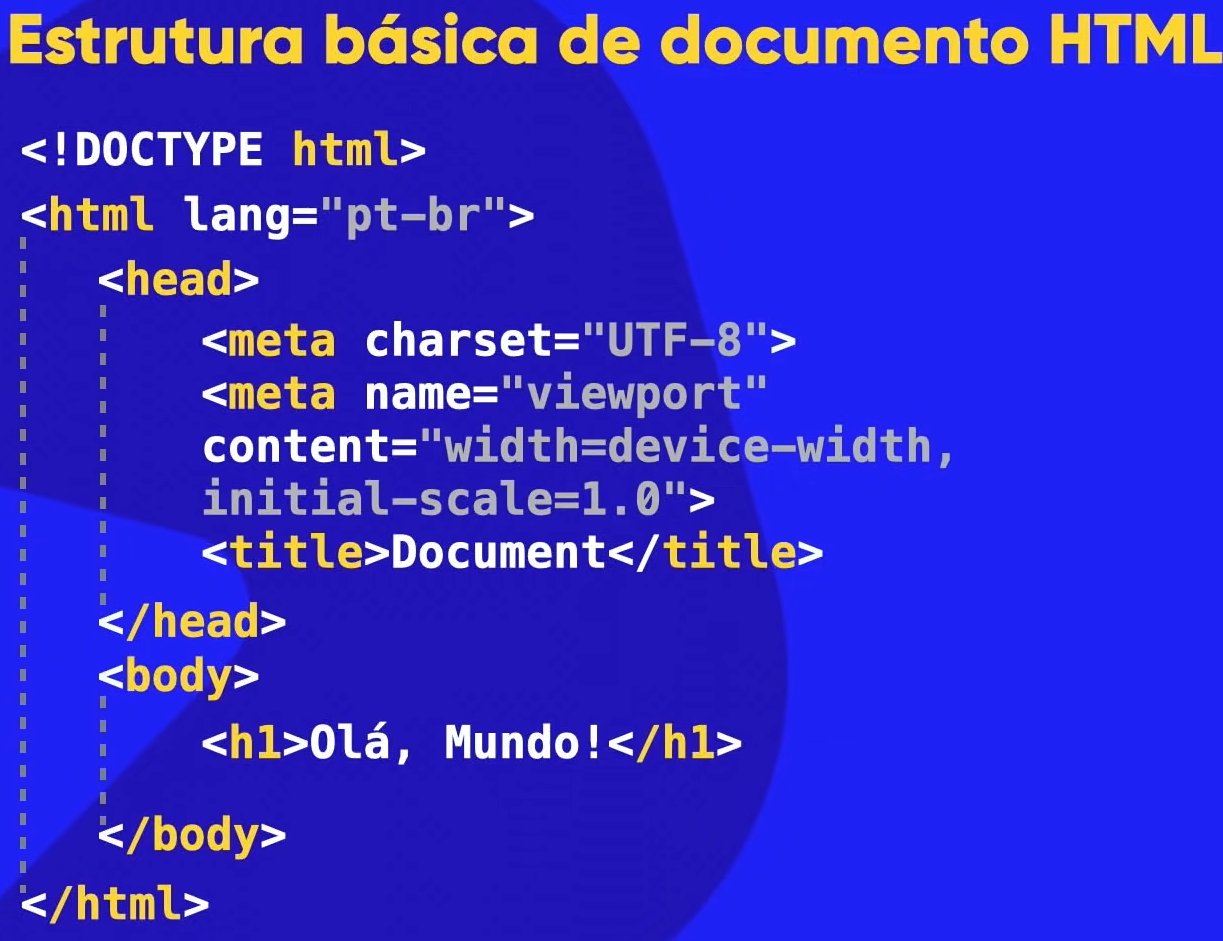
**Comandos em HTML**

**Para colocar títulos ou Tags em HTML** usamos a tag <h> para iniciar e </h> para fechar.

Exemplo: <h1>titulo</h1>

**Para usarmos parágrafos em HTML** usando a tag <p> para iniciar e </p> para fechar.

Exemplo: <p>parágrafo</p>



* Dentro do Visual estúdio code, ao criarmos um novo arquivo com a tag index.html e digitamos ! (ponto de exclamação) o mesmo insere automaticamente a estrutura básica do HTML.
* Para criarmos uma linha horizontal no Visual estúdio code usamos a tag <hr>
* Para usarmos quebras de linha no visual estúdio code usamos o comando <br> que significa Break Roll = quebra de linha.
* Para destacarmos ícones em um site usamos as tags &lt;ICONE&gt;
* Para adicionarmos comentários dentro do Visual code, a tag <!-- com isso podemos usar comentários dentro do programa sem que ele afete nosso layout externo do site.

Para adicionarmos Emojis dentro do visual code sempre usamos as tags &#x+ o código do emoji e finalizamos com ; (ponto e vírgula).

* Para adicionarmos imagens dentro do visual code usamos a tag img, sendo que a tag ALT é o campo de explicação da imagem.
* Para mudarmos o Favicon de um site, usamos a tag Link encima da linha de tittle, escolhemos a opção link:favicon, apagamos o conteúdo entre as “ “ (aspas) e apertamos CTRL+Espaço para selecionarmos um ícone dentro da pasta do index ou subpastas.
* Para adicionarmos parágrafos randômicos dentro do visual code primeiro criamos um parágrafo <p> e depois inserimos a tag lorem.
* H1 e o título principal, sendo h2 o subtítulo e h3 o subtítulo do subtítulo, podendo haver até 6 camadas sendo o h6 o máximo.
* Para colocarmos um texto em negrito usando a tag <b> ou usando a tag STRONG.
* Para usarmos textos em itálico usamos a tag <i> e arrastamos o texto para dentro do comando ou podemos selecionar a parte do texto que queremos colocar em itálico usar a tag CRTL+SHIT+P e digitar o comando ABB, apertamos a tecla I e damos Enter.
* O comando CRTL+SHIT+P Pode ser usado para diversas funções como colocar títulos h1, parágrafos <p>, destacar palavras <b>, <STRONG>, <I> ou <EM>.
* Paga marcarmos um texto usando a tag MARK junto com a tag CRTL+SHIT+P.
* Para mudarmos a cor da marcação usamos a tag MARK junto com a tag STYLE juntamente coma opção background-color.
* Para mudarmos todas as marcações em geral no texto inserimos a tag STYLE logo abaixo do títulos, onde vamos inserir a tag MARK+{} onde vamos selecionar a opção background-color.
* Para colocarmos um texto onde o mesmo deve ser lido, mas não considerado usamos a tag CRTL+SHIFT+P junto da tag del.
* Para marcarmos um texto como inserido para dar uma ênfase e indicar que ele foi adicionado depois usamos a tag CRTL+SHIFT+P seguido da tag ins.
* Para adicionarmos uma palavra, número sobre um texto usamos a tag CRTL+SHIFT+P com o comando SUP (SUPERIOR).
* Para adicionarmos uma palavra, número abaixo de um texto usamos a tag CRTL+SHIFT+P com a tag SUB(INFERIOR).
* Para deixarmos uma fonte com os mesmos padrões de tamanho envelopamos com a tag CODE.
* Para adicionarmos tubulações em códigos envelopamos a tag CODE com a tag PRE, então poderemos fazer várias coisas como adicionar espaços apertando ENTER, ou recuar parágrafos selecionando o texto e apertando os comandos SHIFT+TAB para recuar ou somente TAB para darmos espaço.
* Para colocarmos citações simples e sem quebra, usamos o envelopamento junto da tag Q.
* Para colocarmos abreviações das palavras envelopamos a palavra que queremos abreviar com a tag ABBR.
* Para fazer citações completas com link usamos a tag Blackquote.
* Para elaborarmos um texto em listas simples, usamos a tag OL e dentro do comando OL usamos a tag LI.
* Para colocarmos uma lista em sequência usamos a tag type para termo opções de numeração podemos usar a tag start (padrão numérico) se quisermos começar a lista de numerações dizentes do número 1.
* Para colocarmos listas não ordenadas usamos a tag UL e podemos colocar a tag type+circle,disk ou square.
* Para colocarmos um termo usamos o comando DT e para colocarmos a descrição de um termo usamos a tag DD abaixo de DT.
* Para adicionarmos links externos no site, usamos a tag de envelopar seguido da tag A, onde colocamos o endereço dentro da tag href.
* Para mantermos o cliente em nossa página quando ele clicar em links em nosso site, usando a tag rel=\_ blank e targuet=external assim o novo link irá abrir em uma nova janela.
* Para definirmos qual será a próxima página que queremos acessar clicando nos links usamos a tag rel=next e a anterior usamos a tag rel=prev.
* Para voltarmos na página inicial ou secundaria, envelopamos a frase que queremos e usamos a tag ../ e então escolhes quais das opções queremos colocar como opção de volta.
* Para inserirmos um link de download usamos a tag de lista desorganizada UL ou lista organizada OL e dentro colocamos a tag LI de lista e envelopamos com a tag de link A.
* Para adicionarmos imagens referentes a links de youtube, facebook, linkedin ou afins salvamos uma imagem com proporção 16x16 no gimp ou baixamos direto do site icon archive e inserimos dentro do visual code usando as tags UL ou OL seguidos de LI e img, depois envelopamos usando o comando CRTL+SHITF+P selecionamos e usando a tag de inserir links A e posteriormente inserimos o link da rede social e depois usamos a tag rel\_blank e target\_external para abrir os links em uma nova aba.
* Para inserirmos imagens dinâmicas para um site usamos a tag Picture e depois inserimos a tag source:media:type. É importante sempre colocar o número de dimensionamento da imagem e a sigla px no final da numeração.
* Podemos pegar músicas gratuitas da biblioteca de áudio do youtube.
* Para inserirmos áudio em um site usamos a tag áudio em seguida selecionamos o arquivo de música que deve estar dentro de uma pasta do projeto que está em aberto e usamos a tag controls autoplay para adicionarmos um miniplayer de áudio dentro do site com a música escolhida, a maioria dos navegadores são compatível com 3 formatos, sendo eles mp3/ogg/wav, podemos fazer as músicas rodarem automaticamente simplesmente adicionando a tag loop a tag controls autoplay, ficando assim controls autoplay loop.
* Usamos o gimp para editar imagens e o handbreak para redimensionar formatos de vídeos.
* Para inserir vídeos dentro de um site temos que nos atentar ao tamanho do vídeo, um tamanho maior fará o cliente pagar mais valor pelo trafego dos dados, usamos a tag vídeo, podemos ajustar o tamanho do vídeo no site usando a tag width e depois inserimos a tag controls para colocarmos o player de vídeo, lembrando que o arquivo do vídeo deve estar na mesma pasta do projeto.
* Para economizarmos trafego de dados em nosso site, podemos usar um link do vídeo externo para economizar banda, podemos usar o vimeo/youtube, simplesmente clicamos em incorporar, copiamos o conteúdo e colamos dentro do VScode.

***CSS3***

* Para mudarmos o fundo do site vamos até a tag body e inserimos a tag style e depois a tag backgroud:color e inserimos a cor que queremos.
* Para mudarmos a corda dos títulos h1 ou subtítulos h2 e assim subsequente usamos o comando style:color dentro do h1 afins, exemplo: h1 style:color:blue.
* Para mudarmos a fonte adicionamos a tag font logos depois de body e selecionamos a que nos agrada, importante sempre adicionar os comando de CSS após o ponto e vírgula ( ; ).
* Para alinharmos um texto usamos o comando style:text-align e escolhemos entre as várias opções, justificar, centralizar e afins.
* Podemos mudar o tamanho da fonte usando o comando font:size, colocamos o tamanho de fonte queremos e inserimos o tamanho e colocamos px.
* Podemos aplicar estilos dentro de um site usando o comando style abaixo do title, dentro do comando style usamos o comando body usando chaves {}e usamos a tag que queremos aplicar no site. Lembrando tudo que estiver dentro da tag body terá o efeito aplicado, importante sempre colocar ponto e vírgula no fim de cada declaração, não precisa colocar espaço entre as tag’s.
* Podemos aplicar o mesmo estilo de CSS em vários sites usando a tag link:css e criando uma nova página, podemos colocar o nome que queremos, após isso apertamos CRTL+clique sobre o nome do arquivo que colocamos e criamos uma nova página, dentro desta página vamos trabalhar todos os estilos de CSS que queremos, podendo ser aplicado para inúmeras páginas, outro modo e copiando o conteúdo style e body e colando nas páginas, porém e muita mais trabalhoso, **exemplo abaixo:**

    <title>Site de Exemplo</title>

    <link rel="stylesheet" href="cores02.css">

* CRTL+x recorta o conteúdo dentro do **VScode**, em caso de incompatibilidade usamos a tag **@charset "UTF-8";**
* Sempre colocamos os estilos dentro do início das tags, se quisermos estilos únicos por páginas, precisamos aplicar direta e de maneira individual em cada página.
* O **style inline** sempre vai priorizar o site, depois o local e depois o externo.
* Para inserirmos degrade em CSS em nosso site, usamos a tag **style** seguido pela tag **body** com as chaves {} depois inserimos a tag **background-image:linear-gradient**(escolhemos a posição do degrade, geralmente usamos to right, left, top ou bottom, colocamos virgula e adicionamos as cores que queremos sem espaço separado por virgula, **exemplo abaixo:**

    <title>Gradiente em CSS</title>

    <style>

        \*{ /\*configurações globais para CSS\*/

            height: 100%;

        }

        body{

  background-image:linear-gradient(to top,#FF8102,#C72394,#8832B1);

background-attachment: fixed;

        }

    </style>

* Usamos a tag **background-attachment: fixed;** para deixarmos o gradiente fixo dentro do site.
* Podemos mudar o ângulo do gradiente substituindo o to top,left,right,bottom por ângulos+deg ou até mesmo negativo, além disso podemos colocar porcentagem de exibição nas cores, **exemplo abaixo:**
* **background-image: linear-gradient(45deg,#FF8102,#C72394,#8832B1);**
* Usamos a tag **background-attachment: fixed;** para fixar o degrade em nosso site.
* Usamos a tag **width** com o dimensionamento em **px** para dimensionar o tamanho do fundo de um conteúdo ou texto.
* Usamos a tag **padding** com o dimensionamento em **px** para aumentar ou diminuirmos a margem de um texto ou fundo.
* Usamos a tag **margin:auto** para centralizarmos um texto ou conteúdo no meio do site.
* Usamos a tag **main** para representarmos o conteúdo principal do site, podendo ele ser diverso, imagem, texto ou afins.
* Usamos a tag **border-radius** para arredondarmos as bordas do conteúdo principal (main).
* Usamos a tag **box-shadow** para colocarmos sombra em nosso conteúdo principal, sendo o parâmetro inicial:
* Para o lado
* Para baixo
* Espalhar
* Transparência
* Usamos a tag **text-shadow** para adicionarmos cores aos títulos, subtítulos ou parágrafos, sempre usando o dimensionamento que queremos com a tag **px** no final, **exemplo abaixo:**

@charset "UTF8";

\*{

    font-family: Arial, Helvetica, sans-serif;

}

body{

 background-image:linear-gradient(to right,#F2A679,#D9695F);

}

main{

    background-color: white;

    width: 600px;

    padding: 50px;

    margin: auto;

    border-radius: 20px;

    box-shadow: 5px 5px 15px #3a1d1dcb;

}

h1{

    color:#8C4646;

    text-align: center;

    text-shadow: 1px 1px 2px #552b2b98;

}

h2{

    color: #5D8C7B;

    text-shadow: 1px 1px 1px #5D8C7B;

    text-align: left;

}

p{

    text-align: justify;

}

* Um site tem como tamanho natural de suas fontes 16px, recomendação oficialmente recomendadas, em ou px.
* Para mudarmos o peso da fonte usamos a tag font-weight, usando o **font-weight:bolder** o texto acaba ficando em negrito.
* Para mudarmos o estilo da fonte usamos a tag font-style.
* Podemos usar uma tag chamando **shorthand** para mudarmos a identidade das letras somente copiando o conteúdo das tag em uma única linha, adicionando suas propriedades, **exemplo abaixo:**

        h1{

            /\*shorthand font

            font-style->font-weight->font-size->font-family

            font-style: italic;

            font-weight: bolder;

            font-size: 3em

            font-family: 'work sans',sans-serif;

            font: italic bolder 3em 'work sans',sans-serif;

        }

        h2{

            font: bold 2em 'work sans';

        }

* Usando o **google fonts,** podemos copiar e colar estilos que gostaríamos dentro do **VSCODE**, **exemplo abaixo:**

<style>

@import url('https://fonts.googleapis.com/css2?family=Pacifico&display=swap');

 @import url('https://fonts.googleapis.com/css2?family=Cinzel:wght@400;500;600;700;800;900&display=swap');

        body{

           font-family: 'pacifico',Arial, Helvetica, sans-serif;

            font-size: 16px;

            color: black

        }

        h1{

            font-family: 'cinzel', Times, serif;

            font-size: 5em;

        }

        h2{

            font-family: 'pacifico', Times, serif;

            font-size: 3em;

        }

</style>

* Para adicionarmos fontes externas, baixamos a fonte, um site bom e o Dafont, depois do download, extraímos e recortamos ou copiamos o arquivo para dentro da pasta do projeto ou criamos uma pasta dentro da pasta do projeto. Depois usamos a tag **@font-face**{ para adicionarmos essa fonte externa, no campo src: url podemos usar o comando CRTL+ESPAÇO e selecionarmos a fonte diretamente da pasta, **exemplo abaixo:**

<style>

        @font-face {

            font-family: 'lol';

            src: url('Black\ Way\ -\ Personal\ Use.otf')

        format('opentype')}

        @font-face {

            font-family: 'lol2';

            src: url('love\ larry\ ttf.ttf')

format('truetype')}

</style>

* Depois podemos livremente adicionar esse estilo de fonte para onde quisermos, títulos <h1>, subtítulos <h2> ou parágrafos <p>, basta colocarmos o nome da **font-family** que escolhemos para a fonte externa, **exemplo abaixo:**

        }

        h1{

            font-family: 'lol';

            font-weight: lighter;

            font-size: 5em;

        }

        h2{

            font-family: 'lol2';

            font-weight: normal;

            font-size: 5em;

        }

* Podemos capturar fontes de qualquer site usando a extensão para o google chrome **Fonts Ninja**, após identificada a fonte que queremos copiamos o nome e a buscamos em sites de fontes como o **Google fonts** ou o **DAfont.**
* Podemos capturar fontes direto de imagens, para isso somente salvamos e depois colamos e arrastamos as imagens para dentro do site **find Picture**.com, selecionamos a área da imagem que contem a fonte que queremos e depois ajudamos o site a identificar os caracteres da imagem, após isso o site nos dará fontes iguais ou semelhantes aquelas, então copiamos o nome da fonte e buscamos a mesma caso seja gratuita em site de fonts, abaixo um exemplo de uma fonte encontrada de uma imagem e identificado no site **font finder** e baixada do site **DAfont**, **exemplo abaixo:**

    <title>Capturando fontes em imagens</title>

    <style>

        @font-face {

            font-family:'hoedown';

            src: url('Hoedown-Shadow.ttf')

            format('opentype')

        }

        @font-face {

            font-family:'hoedown2';

            src: url('Hoedown.ttf')

            format('opentype')

        }

        h1{

            font-family: 'hoedown';

            font-size: 5em;

        }

        h2{

            font-family: 'hoedown2';

            font-size: 5em;

        }

    </style>

* Abaixo estão os formatos de tipos de fontes que devemos inserir na tag **format,** **exemplo abaixo:**

            tipos de formatos

            - opentype (otf)

            - truetype (ttf)

            - embedded-opentype

            - truetype-aat

            - svg

* Para darmos espaçamento nos parágrafos usamos a tag **text-indent: 20px**, podemos alinhar os textos usando o comando **text-align**: ele nos dará um leque de opções que podemos escolher conforme acharmos melhor, podemos centralizar ao meio, a direita ou a esquerda, **exemplo abaixo:**

    <title>Alinhando textos e imagens em CSS</title>

    <style>

        body{

            font: normal 16px arial, verdana, serif;

        }

        h1{

            text-align: center;

        }

        h2{

            text-align: right;

        }

        p{

            text-align: justify;

            text-indent: 30px;

        }

    </style>

* Para mudarmos os estilos dos textos de maneira individual, além do comando inline podemos identificar as tags como principais, assim atribuindo estilos e afins de maneira única neles, **exemplo abaixo:**

 <h1 id="principal">Criando Sites com HTML e CSS</h1>

h1#principal{

    font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;

    text-align: center;

    background-color: rgb(66, 117, 11);

    color: rgb(182, 209, 233);

}

* Temos seletores únicos para personalização que sofrem transformações quando aplicado de HTML para CSS, **exemplo abaixo:**

em HTML é id = em CSS é # (cerquilha)

  em HTML é class = em CSS é . (ponto)

: (dois pontos) significa uma pseudo-classe

* Quando já tivermos nosso **ID** principal, para usamos estilos ou afins em outros textos que gostamos usamos a tag **class=’nome que quisermos’** isso em HTML, depois vamos para nossa página de CSS e colocamos a **tag+ponto+nome** que escolhemos para a class, e então trabalho dentro do texto o que quisermos, cores, tamanhos, fundos ou afins, **exemplo abaixo:**

<h2 class="básico">HTML Básico</h2>

<h2 class="intermediário">HTML intermediário</h2>

<h2 class="avançado">HTML Avançado</h2>

<h2 class="básico">CSS básico</h2>

<h2 class="intermediário">CSS intermediário</h2>

<h2 class="avançado">CSS Avançado</h2>

<h1 class="destaque">Aprendendo CSS</h1>

**PARTE EM HTML5 ACIMA**

.básico{

    color: red;

}

.intermediário{

    color: blue;

}

.avançado{

    color: rgb(163, 163, 43);

}

}

.destaque{

    background-color: darkmagenta;

}

**PARTE EM CSS3 ACIMA**

* Podemos colocar mais de uma classe, sendo necessário somente separá-las com espaço.
* Podemos usar a tag **div** para criarmos formas dentro de um site, assim usamos varias tag para dimensionar a largura e altura da forma, cor e afins, **exemplo abaixo:**

<style>

        div{

          /\*tudo na mesma linha\*/  display: inline-block;

          /\*cor de fundo\*/  background-color: lightgray;

          /\*borda\*/         border: 1px solid black;

          /\*alinhamento de texto\*/  text-align: center;

         /\*altura\*/   height: 200px;

         /\*largura\*/   width: 200px;

        /\* hover significa: quando eu passar o mouse encima\*/

}

        div:hover{

            background-color: yellow;

        }

    </style>

</head>

<body>

    <div id="d1">01</div>

    <div id="d2">02</div>

    <div id="d3">03</div>

* Podemos inserir texto escondidos e mostrarmos os mesmo quando passar o mouse sobre eles usando as tags **div > p { e div:hover > p{** é necessário somente inserimos um parágrafo dentro de uma div e inserirmos o texto que queremos deixar temporariamente oculto, primeiro usamos a tag **div > p { e a tag display:none;** para escondermos o texto, em seguida usamos a tag **div:hover >p{** e a tag **display:block.** Para revelarmos o texto quando passarmos o mouse por cima, **exemplo abaixo:**

<body>

    <h1>Exemplo de hover</h1>

    <p>Passe o mouse sobre o texto abaixo:</p>

    <div>Passe o mouse aqui

        <p>TEXTO ESCONDIDO</p>

    </div>

    <div id="d2">Passe o mouse aqui 2

        <p>TEXTO ESCONDIDO 2</p>

    </div>

    <p>fim do exemplo</p>

</body>

 div > p {

            display: none;

        }

        div:hover > p {

            display: block;

        }

* Podemos mudar a cor dos links de sites inserindo a tag **a**{ **exemplo abaixo:**

        a {

            color: red;

            text-decoration: none;

            font-weight: bold;

        }

* Podemos também mudar a cor dos links já visitados inserindo a tag **a:visited{** **exemplo abaixo:**

a:visited {

            color: darkgoldenrod;

        }

* Podemos também mudar a cor do link quando colocamos o mouse sobre ele e também adicionarmos um sublinhado usando a tag **a:hover**{ **exemplo abaixo:**

 a:hover {

            color: black;

            text-decoration: underline;

        }

* Podemos também inserirmos uma cor ao clicar em um site usando a tag **a:active**{ **exemplo abaixo:**

a:active{

            color:yellow

        }

* Podemos indicar em nosso site que a palavra do texto ou link é realmente um link, inserindo a tag **a::after**{ **exemplo abaixo:**

 a::after{

            content: '[link]';

            text-decoration: none;

            font-weight: normal;

            color: darkgray;

        }

* Podemos inserir ícones tanto antes quanto depois dos links usando a tag **a::before{ e a** tag **a::after{** então inserimos a tag **content:** e colamos os ícones que copiamos, **exemplo abaixo:**

        }

        a::before{

            content:'▶' ;

        }

        a::after{

            content: '◀'

        }

* Podemos personalizar link únicos adicionando uma classe a eles usando a tag class=’nome que quisermos’ e depois em CSS3 colocamos a tag .nome que escolhemos::before{ content:’icone’ para ícones, imagens e afins antes e a tag .nome que escolhemos:after{contente: ‘icone’ para ícones, imagens e afins, depois, **exemplo abaixo:**

        }

        .especial::before{

            content:'▶' ;

        }

        .especial::after{

            content: '◀'

<h1>Personalizando Links</h1>

   <ul>

       <li><a href="https://gustavoguanabara.github.io" target="blank">Repositório github</a></li>

       <li><a href="https://youtube.com/cursoemvideo/" class="especial">Canal do Youtube</a></li>

       <li><a href="https://www.cursoemvideo.com">Site do CursoemVideo</a></li>

   </ul>

* Para configuramos a altura e largura de um texto usamos as tag **height** (altura) e **width** (largura),
* podemos traçar uma linha ao redor do conteúdo usando a tag **border**,
* podemos mudar o distanciamento da borda do conteúdo usando a tag **padding**.
* Podemos mudar o distanciamento da borda externa do conteúdo usando a tag **margin**.
* Podemos mudar a corda da linha da borda usando a tag **outline**.
* Podemos verificar informações de sites, altura, largura margens e afins usando a ferramenta do google chrome **DevTools**, basta clicarmos em inspecionar.
* Podemos também testar os tamanhos e estilos dos conteúdos aplicados no site inspecionando o mesmo com o **DevTools,** as modificações usando o **DevTools** não mudam o formato original do conteúdo, **exemplo abaixo:**



* Podemos centralizar a margem usando a tag margin:**auto**
* Podemos usar a tag **outline** para colocarmos uma linha entre a margem e a borda, **exemplo abaixo:**

            outline-width: 5px;

            outline-style: dashed;

            outline-color: blueviolet;

**SHORTHANDS**

* Podemos simplificar as tags usando uma única sequencia simples de tag, **exemplo abaixo:**

            border-width: 10px;

            border-style: solid;

            border-color: darkblue;

abaixo podemos ver o comando simplificado que da o mesmo resultado do que da imagem acima:

           border:10px solid darkblue;

Para usarmos o **shorthand** para a tag **padding**, usamos a tag **padding** seguindo o sentido de aplicação no sentido horário de um relógio sendo top,right,bottom,left, **exemplo abaixo:**

            padding-left: 10px;

            padding-top: 10px;

            padding-bottom: 10px;

            padding-right: 10px;

abaixo podemos ver o comando simplificado que dá o mesmo resultado do que da imagem acima:

            padding: 10px 10px 10px 10px;

Para usarmos o **shorthand** para a tag **margin**, usamos a tag **margin** seguindo o sentido de aplicação no sentido horário de um relógio sendo top,right,bottom,left, **exemplo abaixo:**

            margin-top: 20px;

            margin-right: 20px;

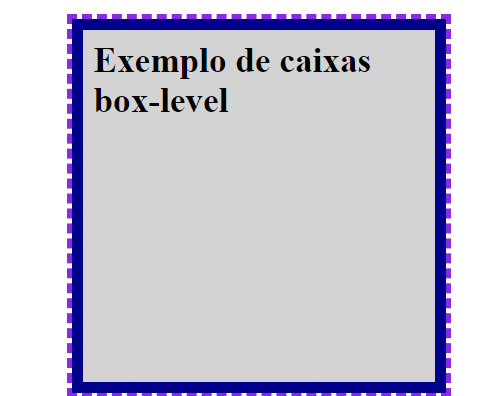
            margin-bottom: 40px;

            margin-left: 20px;

abaixo podemos ver o comando simplificado que dá o mesmo resultado do que da imagem acima:

            margin: 20px auto 20px auto;

Abaixo temos um exemplo de box-level:

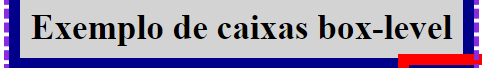


Abaixo temos um exemplo de inline-level:



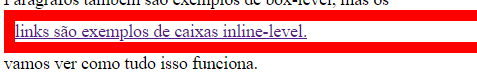
Podemos transforma um estilo **box-level** para um estilo **inline-level** usando a tag **display:inline;** porém como estamos mudando para inline-level não podemos regular altura (height) nem largura (width) **exemplo abaixo:**

            display: inline;



Podemos também transformas estilos **inline-level** para estilos **box-level** usando a tag **display:block**, **exemplo abaixo:**

            display: block;



Para adicionarmos um cabeçalho usamos a tag **header**.

Para adicionarmos um conteúdo principal usamos a tag **main**.

Para adicionarmos um roda pé usamos a tag **footer**.

Para adicionarmos um menu com link usamos a tag **nav**.

Para adicionarmos seções em nosso site usamos a tag **section**.

Para adicionarmos artigos usamos a tag **article**, para usarmos uma side, ou seja, um assunto relativo ao assunto do artigo usamos a tag **aside**.

Devemos organizar os estilos e seções do HTML em CSS, **exemplo abaixo:**

 <style>

        body{

            font-family: Arial, Helvetica, sans-serif;

            background-color: dimgray;

            margin: 0px;

        }

        header{

            background-color: white;

            padding: 10px;

            margin: 10px;

        }

        article{

            background-color: lightgray;

            padding: 5px;

        }

        nav{

            background-color:rgb(172, 161, 161);

            padding: 3px;

        }

        nav > a{

            text-decoration: none;

            color: white;

            font-weight: bold;

            margin-right: 5px;

        }

        nav > a:hover{

            text-decoration: underline;

            color: chartreuse;

        }

        }

        article > aside {

            background-color: rgb(156, 155, 155);

        }

        main{

            background-color: white;

            padding: 10px;

            margin: 10px;

        }

        footer{

            background-color: black;

            color: white;

            text-align: center;

            padding: 1px;

            margin: 0px;

        }

    </style>

**Conteúdo em CSS acima**

<body>

    <header>

        <h1>Meu site</h1>

        <nav>

            <a href="#">link</a>

            <a href="#">link</a>

            <a href="#">link</a>

            <a href="#">link</a>

            <a href="#">link</a>

        </nav>

    </header>

    <main>

        <section id="assuntos"></section>

        <p>Esportes Politica e Tecnologia</p>

        <section id="noticias">

            <article>

                <h2>Noticias sobre o Brasil</h2>

                <p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Consequuntur itaque adipisci expedita totam consequatur perspiciatis sunt distinctio fuga? Animi temporibus at quam architecto unde necessitatibus sunt laboriosam nisi odit ab?</p>

                <aside>

                    <p>Artigo escrito por julio cesar baccin</p>

                </aside>

            </article>

        </section>

    </main>

    <footer>

        <p>Desenvolvido pelo julio cesar baccin</p>

    </footer>

</body>

**Conteúdo em HTML acima**

Podemos copiar estilos diretamente do **DevTools** para p **VSCcodem** basicamente precisamos somente copias os estilos configurados do **Devtools** para o **VScode** (dica de ouro), **exemplo abaixo:**

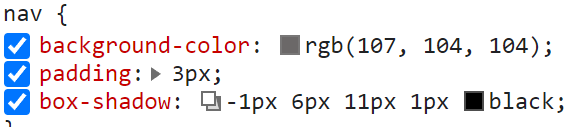
        nav{

            background-color:rgb(107, 104, 104);

            padding: 3px;

            box-shadow: -1px 6px 11px 1px black;

        }



O estilo da tag **box-shadow** foi configurada diretamente do **devtool**s e depois os valores somente foram copiados e colados no VScode adicionando assim a propriedade ao site.

Podemos fazer um círculo em CSS adicionando as seguintes configurações, **exemplo abaixo:**

   <div id="circulo"></div>

        #circulo{

          height: 100px;

          width: 100px;

          background-color: white;

          border-radius: 50%;

        }

Podemos criar uma borda personalizada em nosso site usando a tag **border-image-source: url**(‘**escolhemos o arquivo ou colocamos o nome da borda**’);

Podemos cortar a borda usando a tag **border-image-slice**.

Podemos repetir a borda usando a tag **border-image-repeat**, **exemplo abaixo:**

    <style>

        h1{

           border: 20px solid black;

           padding: 10px;

           border-image-source:url('borda.png');

           border-image-slice: 27;

           border-image-repeat: repeat

        }

    </style>

Podemos usar o site Mockflow para fazermos pré-visualizações de projetos de site.

Podemos dar várias denominações e pré modelos de cores e letras, onde podemos trocá-las facilmente usando a tag **:root,** **exemplo abaixo:**

@charset "UTF-8";

@import url('https://fonts.googleapis.com/css2?family=Bebas+Neue&display=swap');

@font-face {

    font-family:'Android' ;

    src: url("../fontes/idroid.otf") format('opentype');

    font-weight: normal;

}

/\*

#c5ebd5

#83elad

#3ddc84

#2fa866

#1a5c37

#063d1e

\*/

:root{

    --cor0:#c5ebd5;

    --cor1:#83e1ad;

    --cor2:#3ddc84;

    --cor3:#2fa866;

    --cor4:#1a5c37;

    --cor5:#063d1e;

    --fonte-padrao: Arial ,Verdana ,Helvetica ,sans-serif;

    --fonte-destaque: 'Bebas Neue', cursive;

    --fonte--android: 'android', cursive;

}

body{

    background-color:var(--cor0);

    font-family: var(--fonte-padrao);

}

header > h1{

    color:var(--cor5);

    font-family: var(--fonte-destaque);

}

main h1{

    color:var(--cor3);

    font-family: var(--fonte--android);

}

Podemos váriar as corres e o tipo de fonte adicionando a tag **var** tanto em font quanto em color, então podemos escolher aquelas aplicadas na tag **:root,** assim podendo mudar facilmente as opções desejadas.

Podemos tirar as margens e paddings usando uma configuração global, sendo \*{}, **exemplo abaixo:**

\*{

    margin:0px;

    padding: 0px;

}

Podemos adaptar o conteúdo de nosso site para vários dispositivos, usando as tag **min-width** e **max-width**, também usamos a tag **margin:auto** para que as imagens e textos se adaptem melhor dentro do site e para várias tamanhos de tela de exibição, dessa maneira podemos também usar a **variável img**{ com a tag **width:100%** para adaptarmos as imagens dentro do site, melhorando assim a visualização e evitando que a imagem se quebre, **exemplo abaixo:**

<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>Teste de responsividade</title>

    <style>

        body{

            background-color: darkgray;

        }

        main{

            background-color: white;

            padding: 20px;

            border-radius: 10px;

            min-width: 320px;

            max-width: 800px;

            margin: auto;

        }

        img{

            width: 100%;

        }

    </style>

</head>

Podemos usar a extensão do google Chrome **window resizer** para ver o tamanho da nossa tela e simular tamanhos de telas diferentes.

Podemos usar a tag **Picture** junto com as tag **source media**=’max-width: 800px)” e srcset=”localdaimagem/nomedaimagem.tipodaimagem para mostrarmos versões diferentes da mesma imagem, claro que devemos adaptar o tamanho de exibição que queremos usando a tag **width** e também o tamanho da própria imagem, **exemplo abaixo**:

<picture>

            <source media="(max-width: 800px)" srcset="imagens/irina-blok-pq.jpg">

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

        </picture>

No código acima estamos dizendo que até 800px de largura queremos que seja exibida a imagem menor, pois se adapta melhor a dispositivos mobile, hoje onde a grande maioria dos acessos a sites se concentra, então temos uma versão da mesma imagem porem com uma largura maior que é apresentada quando a viewport ou seja a tela de exibição da pessoa que está acessando nosso site for maior que 800px ou seja largura, mostrará essa imagem maior.

Podemos centralizar imagens usando a tag **margin:auto**, também podemos colocar a proporção em % que queremos da imagem usando width:100%, **exemplo abaixo:**

main img.pequena{

    width: 50%;

    display: block;

    margin: auto;

}

Podemos adicionar um estilo gradiente onde quisermos apenas colocando o elemento que queremos e depois usando a tag **background-image: linear-gradient**(**então escolhemos a direção que iniciaremos, posteriormente colocamos a cor inicial do degrade e a final**, **exemplo abaixo:**

header {

    background-image: linear-gradient(to bottom, var(--cor3), var(--cor5));

Podemos colocar uma animação no menu de navegação (nav) usando a tag **transition-duration:** **exemplo abaixo:**

nav > a{

    color: white;

    padding: 10px;

    text-decoration: none;

    font-weight: bold;

    transition-duration: 0.5s;

}

nav > a:hover{

    background-color: var(--cor3);

    color: var(--cor5);

Podemos atribuir uma função especial aos parágrafos usando a tag **margin** e a **tag text-align,** sendo assim a mesma aplicada a todos os parágrafos, **exemplo abaixo:**

main p {

    margin: 15px 0px;

    text-align: justify;

}

O ideal quando formos dimensionar imagens dentro de nosso site e termos 2 tamanhos de imagens 1 para aparelhos menores e outra para aparelhos com uma **viewport** maior, o ideal e configuramos o **max-width** com a largura máxima que a imagem tem, assim não perdemos qualidade da mesma na hora da exibição.

Para colocarmos o recuo nos parágrafos usamos a tag **text-indent:** **exemplo abaixo:**

text-indent: 30px;

Para adicionarmos espaçamento entre as linhas de texto usamos a tag line-heigth: **exemplo abaixo:**

line-height: 2em;

Podemos colocar um gradiente transparente usando a tag **transparent**, **exemplo abaixo:**

  background-image: linear-gradient(to right, var(--cor1),transparent);

Podemos organizar uma lista dentro do conteúdo da side usando a tag **list-style-position:** então escolhemos a opção que queremos, **exemplo abaixo:**

aside >ul {

    list-style-position: inside;

}

Lembrando somente usamos > na declaração para colocarmos efeito que estão dentro de outras declaraçãos, **exemplo:**

Aside > a – isso significa que todos os links dentro de aside terão as propriedades que forem atribuídas a declaração.

Main a – isso significa que todo link dentro do conteúdo principal terá as propriedades que forem atribuídas na declaração

Podemos dividir nossa lista usando a tag **colums:** **exemplo abaixo:**

aside>ul {

    list-style-position: inside;

    columns: 2;

}

Podemos adicionar símbolos e emolis em uma lista usando a tag **list-style-type:** então colocamos o próprio emoji ou símbolo ou então pegamos somente a numeração antes de u+ e colamos dentro das aspas “ “ sempre usando barra invertida \ **exemplo abaixo:**

aside>ul {

    list-style-position: inside;

    columns: 2;

    list-style-type: "\2714"

}

Podemos adicionar espaços entre o texto e os emojis usando a tag **00A0**, sendo que cada declaração dessas significa um espaço que deve ser separado por uma barra invertida, **exemplo abaixo:**

list-style-type: "\✔️\00A0\00A0";

Para colocarmos ícones ou emojis em links externos usamos a tag com uma subclasse, **exemplo abaixo**:

}

a.externo::after{

    content: "\🔗";

}

<p>A ideia de ter um mascote foi amadurecendo e a missão foi passada para uma profissional da área. A

                ilustradora Russa <a href="https://www.irinablok.com/" target="\_blank" class="externo">Irina Blok</a>, também

                funcionária do Google, ficou com a missão de representar o pequeno robô de uma maneira mais agradável.

            </p>

Em HTML coloco class e em CSS usamos o. (ponto)

Em HTML coloco ID e em CSS usaremos #

Para dimensionarmos o tamanho de um vídeo usamos os **comandos abaixo:**

div.video{

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

    padding: 20px;

    padding-bottom: 59%;

    position: relative;

}

div.video > iframe {

    position: absolute;

    top: 5%;

    left: 5%;

    width: 90%;

    height: 90%;

}

Além das declarações em CSS acima envelopamos o link da incorporação, ou seja, o iframe com a tag **div** e colocamos o elemento class, **exemplo abaixo:**

    <div class="video">

                <iframe width="560" height="315" src="https://www.youtube.com/embed/62JB64aQC9A"

                    title="YouTube video player" frameborder="0"

                    allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture"

                    allowfullscreen></iframe>

            </div>

Quando tivermos vários elementos, é melhor classificarmos com a classe **class**, exemplo: vídeos, botões, parágrafos.

Quando tiver elementos específicos e melhor classificarmos com a classe **ID**, exemplo: meus botões, meu vídeo e afins.

**GIT/GITHUB**

Podemos subir sites após fazermos alterações no arquivo base abrindo o mesmo pelo **GitHub Desktop** e selecionarmos o repositório em que está o arquivo e clicarmos em Open in **Visual Studio Code**, após as alterações o programa irá detectar que nosso arquivo base sofreu alterações, então clicamos em **commit** e depois em **push** para que as alterações sejam lançadas em nosso arquivo no servidor remoto.

**Planos de fundo**

Abaixo vemos 3 declarações de como mudar a imagem de fundo, observamos que primeiramente damos uma classe a eles com o nome quadrado e depois os damos um **ID** onde colocamos cores dos planos de fundos distintos, sendo que usando a classe quadrado escolhemos primeiramente as dimensões dos nossos elementos e com os **ID** as propriedades internas dos elementos, **exemplos abaixo:**

 <style>

        div.quadrado {

            border: 2px solid black;

            width: 300px;

            height: 300px;

            display: inline-block;

            background-color: lightgray;

            border-radius: 10px;

        }

        div#q1 {

            background-color: skyblue;

        }

        div#q2 {

            background-image: linear-gradient(to bottom, yellow, red);

        }

        div#q3 {

            background-image: url(imagens/pattern003.png)

        }

    </style>

</head>

<body>

    <div class="quadrado" id="q1">

    </div>

    <div class="quadrado" id="q2">

    </div>

    <div class="quadrado" id="q3">

    </div>

</body>

Podemos colocar imagens diretamente do google, simplesmente clicando com o botão direito e selecionando o comando copia endereço da imagem, então colocamos na tag **background-image: url();** **exemplo abaixo:**

    <style>

        body{

            background-image:url("https://static.wikia.nocookie.net/naruto/images/8/8d/Nice\_Guy.PNG/revision/latest?cb=20151127190814&path-prefix=pt-br");

            background-size: 500px 500px;

            background-repeat: repeat;

        }

    </style>

Podemos escolher quais ângulos da imagem vamos colocar em nosso background usando a tag **background-position**, **exemplo abaixo:**

<style>

        div.bloco{

            background-image: url("imagens/wallpaper003.jpg");

            height: 200px;

            border: 1px solid black;

            border-radius: 20px;

            margin: 10px;

        }

        div#q1{

            background-position: left top;

        }

        div#q2{

            background-position: left center;

        }

        div#q3{

            background-position: right bottom;

        }

        div#q4{

            background-position: center center;

        }

    </style>

</head>

<body>

    <div class="bloco" id="q1">

    </div>

    <div class="bloco" id="q2">

    </div>

    <div class="bloco" id="q3">

    </div>

    <div class="bloco" id="q4">

    </div>

</body>

Podemos fazer o posicionamento de uma imagem usando a tag height e dimensionando esse posicionamento com uma terminação em VH (VIEW HEIGHT) sendo **height:100vh** o centro do site, **exemplo abaixo:**

<style>

    body{

        background-image:url("https://static.wikia.nocookie.net/naruto/images/8/8d/Nice\_Guy.PNG/revision/latest?cb=20151127190814&path-prefix=pt-br");

        background-size: 100px;

        background-repeat:no-repeat;

        background-position: center center;

        border: 1px dashed black;

        height:98vh;

    }

</style>

Podemos dimensionar o tamanho de uma imagem de fundo usando a tag **background-position: cover;** essa tag faz com que a imagem de fundo seja picotada e para que tenha total exibição em qualquer tamanho de view port ou telam **exemplo abaixo:**

<style>

        body{

            background-color:black;

            background-image:url(imagens/wallpaper002.jpg);

            background-position: center top;

            background-repeat: no-repeat;

            height: 100vh;

**background-size: cover;**

        }

        main{

            background-color:white;

            width: 400px;

            padding: 10px;

            margin:auto;

            border: 1px solid darkblue;

            border: radius 15px;

        }

    </style>

Também podemos usar as tag **background-size:auto** e **background-size: contain**, porém em algumas viewports a tela de exibição pode apresentar bordas laterais.

Podemos usar **shorthands** para backgrounds, usando a tag **background** inicialmente e depois inserindo a seguinte sequência:

**Color > image > repeat > [size] > attachment**

Exemplo da shorthand abaixo:

background: black url(imagens/wallpaper002.jpg) center center no-repeat fixed;

A tag size dentro da shorthand não funciona, assim temos que colocá-lo separado, **exemplo abaixo:**

    body{

             height: 100vh;

            background: black url(imagens/wallpaper002.jpg) center center no-repeat  fixed;

            background-size:cover;

        }

**Aprenda CSS Position em 10 Minutos - Tutorial CSS para Iniciantes**

Usando a tag **position:relative** habilita para que consigamos trabalhar a posição dos conteúdos da maneira que quisermos, usando as tag.

* Top: 10px ou 100%
* Bottom: 10px ou 100%
* Rigth: 10px ou 100%
* Left: 10px ou 100%

Usando a tag **position:absolute** faz com que a forma tenha seu próprio fluxo de posicionamento absoluto, como se ignorasse todo conteúdo existente no documento, desde que essa forma não esteja dentro de uma outra forma com a tag **position:relative. Exemplo abaixo:**

<title>Alinhamento Vertical</title>

    <style>

        #container{

            background-size: 100% 100%;

            background-image:url(imagens/target001.png);

            height: 500px;

            background-color:purple;

            padding:10px;

        }

        #conteudo{

            background-size: 100% 100%;

            background-image:url(imagens/target001.png);

            position: absolute;

            height: 200px;

            width: 400px;

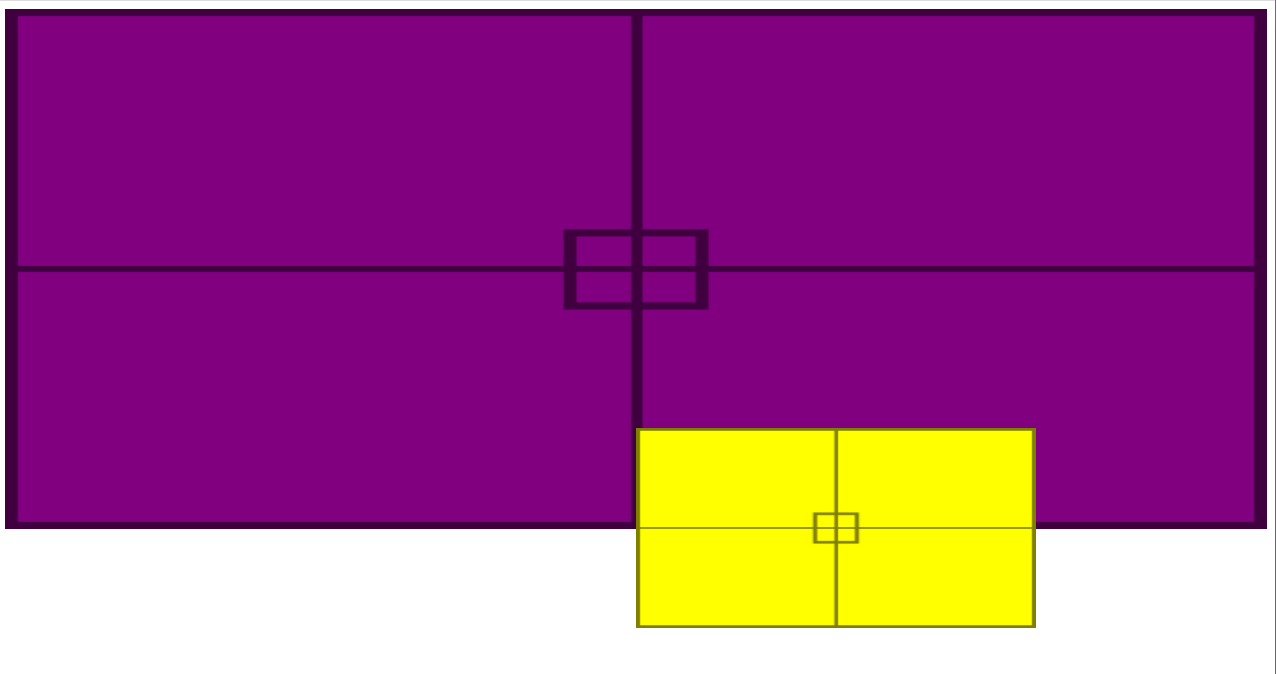
            background-color:yellow;

            left:50%;

            top:50%;

        }

    </style>

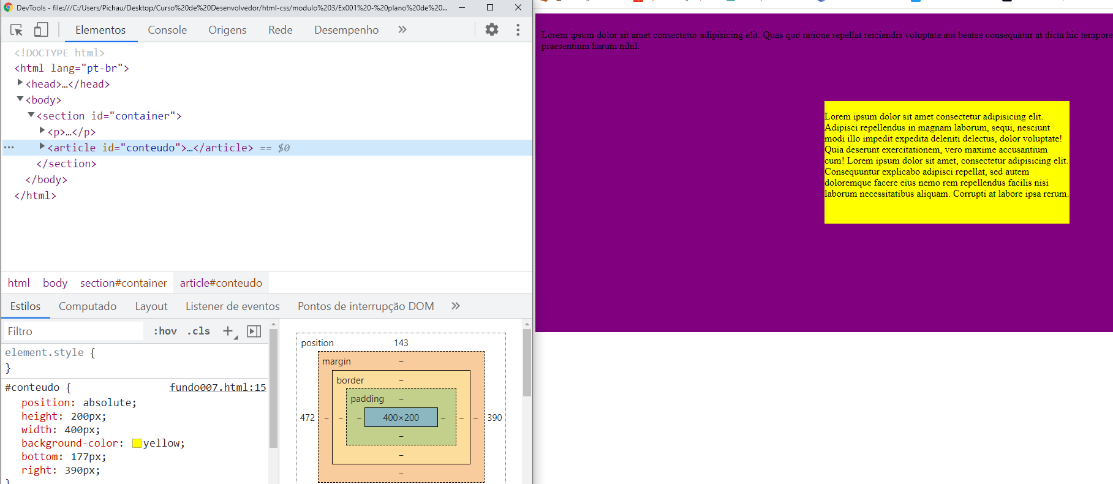


Usando a tag **position:fixed** deixa o elemento fixo na tela, não se importando com as propriedades de textos ou afins que estão abaixo dele, ideal para fazer botões ou header para navegação.

* Position:fixed
* Top:10px

Usando a tag **position:sticky** deixa o elemento grudado na tela ao scrolar a tela, geralmente é uma tag usada a elaboração de menus de navegação.

Podemos simular o local onde queremos dos elementos usando a ferramenta inspecionar código fonte do devtool, então precisamos somente alterarmos as propriedades conforme o posicionamento que desejamos, **exemplo abaixo:**



Podemos fazer também o posicionamento de elementos usando a tag **transform:translate** e colocando a quantidade que queremos de movimentação, **exemplo abaixo:**

transform:translate(-50%, -50%);

Usando **<a> href: target=’blank’** – fará com que os links abram, porém um vai sobressair o clicl do outros, no caso se eu tiver 2 links e clicar em 1 quando o site está carregando após eu clicar no outro este outro irá carregar no lugar do 1 tomando seu lugar na mesma página e janela.

**Usando <a> href: target=’\_blank’** – fará que qualquer link que seja clicado abra uma nova pagina em uma nova janela.

Usando a tag **text-transform: uppercase** podemos deixar todas as letras em maiúsculo, **exemplo abaixo:**

header > h1{

    padding-top: 50px;

    text-transform: uppercase;

}

Usando a tag **font-variant: small-caps** deixamos somente as letras iniciais maiúsculas, **exemplo abaixo:**

header > h1{

    padding-top: 50px;

    font-variant: small-caps;

}

**P**odemos adaptar o tamanho da fonte as diversas viewports colocando a propriedade de VW (viewport) dentro da tag **font-size: 10vw** (10% da viewport), com isso conseguimos fazer com que os textos se adaptem aos tamanhos da tela diminuindo e aumenta conforme o tamanho da tela, **exemplo abaixo:**

section {

    padding-top: 10vh;

    padding-bottom: 10vh;

    line-height: 2em;

    padding-left: 30px;

    font-family: var(--fonte3);

    font-size: 3.5vw;

}

header > h1 {

    padding-top: 50px;

    font-variant: small-caps;

    font-family:var(--fonte2);

    font-size: 10vw;

}

Para colocarmos fontes da internet, podemos copiar o **embed** do site **google fonts** e depois colarmos no projeto, após colarmos os links dentro da pasta de CSS usamos a tag global **:root** para adicionarmos variantes de fontes para adicionarmos em nosso projeto, **exemplo abaixo:**

@charset "UTF-8";

@import url('https://fonts.googleapis.com/css2?family=Passion+One&display=swap');

@import url('https://fonts.googleapis.com/css2?family=Sriracha&display=swap');

:root{

    --fonte1: Verdana, "Geneva", Tahoma, sans-serif;

    --fonte2: 'Passion One', "cursive";

    --fonte3: 'Sriracha', "cursive";

    }

Após isso basta escolher uma das fontes usando a tag **fonte-family:var(crtl+espaço); exemplo abaixo:**

html, body {

    background-color: darkgray;

    min-height: 100vh;

    font-family: var(--fonte1);

}

Para trabalharmos imagens dentro de textos, primeiramente tratamos dos estilos, larguras, fontes e afins das formas, depois inserimos as imagens, **exemplo abaixo:**

    <section class="imagem" id="img01">

        <p>

            Não é mais como era antes<br>

            É tudo abreviado<br>

            "Você" só tem duas letras<br>

            O "O" e o "E" foi riscado<br>

            Para declarar o amor<br>

            Basta botar uma flor<br>

            E um coração desenhado.

        </p>

    </section>

section.imagem {

    background-color: rgb(56, 56, 56);

    color: white;

    box-shadow: inset 6px 6px 13px 0px rgba(0, 0, 0, 0.473);

    background-size:cover;

    background-attachment: fixed;

}

section.imagem > p {

    display: inline-block;

    background-color: rgba(0, 0, 0, 0.514);

    padding: 5px;

    text-shadow: 4px 4px 0 black;

}

section#img01{

    background-image: url(imagens/background001.jpg);

    background-position: right center;

}

section#img02{

    background-image: url(imagens/background002.jpg);

}

Usamos a **classe imagem** para tratarmos da:

Cor de fundo = **background-color**

Cor dos parágrafos = **color**

Sombra da forma = **box-shadow**

Tamanho da forma = **background-size**

Tipo de ligação = **background-attachment**

Depois usamos a **ID”img01”** para adicionarmos a imagem e seu posicionamento de foco:

**Background-image: url (pasta da imagem crtl+espaço)**

**Background-position: right center;**

Para criamos uma tabela em html usamos a tag **table** (tabela) depois inseremos a tag **tr** (linha) e depois a tag **td** (dados da tabela), podemos personalizar a tabela conforme queremos, inserindo **classes** e **ID’s** para personalizações mais individuais, **exemplo abaixo:**

 <style>

        body{

            font-family: Arial, Helvetica, sans-serif;

        }

        table{

           border-collapse:collapse;

           width: 400px;

           height: 400px;

           border: 2px solid red;

           font-size: 5vh;

        }

        td{

            text-align: center;

            padding: 10px;

            font-weight: bolder;

            border: 1px solid black;

        }

        tr.linha1{

            background-color: darkgrey;

        }

        tr.linha2{

            background-color: blue;

        }

        td#amarelo{

            background-color: yellow;

        }

        td#vermelho{

            background-color: red;

        }

        h1{

            font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;

            background-color: aqua;

            display: inline-block;

            padding-bottom: 30px;

            width:400px;

        }

    </style>

</head>

<body>

    <h1>Minha primeira tabela</h1>

    <!-- HIERARQUIA DE TABELAS (simples)

        TABLE = tabela

            TABLE ROW = linha de tabela

                TABLE HEADER = cabeçalho de tabela

                TABLE DATA = dados da tabela-->

    <table>

        <tr class="linha1"><!-- primeira linha-->

            <td>A1

            <td>B1

            <td>C1

        </tr>

        <tr class="linha2"><!-- segunda linha-->

            <td>A2

            <td>B2

            <td>C2

        </tr>

        <tr>

            <td>A3</td><!-- terceira linha-->

            <td>B3</td>

            <td>B3</td>

        </tr>

        <tr>

            <td id="vermelho">A4</td><!-- quarta linha-->

            <td id="amarelo">B4</td>

            <td>C4</td>

        </tr>

    </table>

Podemos inserir uma tabela com cabeçalho, corpo e um rodapé para nossa tabela usando:

**Thead** = cabeçalho

**Tbody =** corpo

**Tfoot** = rodapé

Iniciamos com a tag **table** dentro de **body** depois inserimos a tag **thead** para o cabeçalho, dentro de **thead** inserimos a tag **tr** para as linhas e dentro de **tr** inserimos **th** para colocamos os dados do cabeçalho, **exemplo abaixo:**

<body>

    <h1>Uso de tabelas grandes</h1>

    <!--ANATOMINA PARA TABELAS GRANDES

        TABLE

            THEAD

                TR, TD, TH

            TBODY

                TD, TD, TH

            TFOOT

                TR, TD, TH

    -->

    <table>

        <thead>

            <tr>

                <th class="p1">Posição</th>

                <th class="p1">Estado</th>

            </tr>

        </thead>

        <tbody>

            <tr>

                <td>São Paulo</td>

                <td class="num">45.919.049</td>

            </tr>

            <tr>

                <td>Minas Gerais</td>

                <td class="num">21.169.791</td>

            </tr>

            <tr>

                <td>Rio de Janeiro</td>

                <td class="num">17.264.943</td>

            </tr>

        </tbody>

        <tfoot>

            <tr>

                <th class="p1">Total de habitantes</th>

                <td class="num">84.353.783</td>

            </tr>

        </tfoot>

    </table>

</body>

Podemos inserir uma legenda nas tabelas usando a tag **caption**, **exemplo abaixo:**

<table>

        <caption>População das unidades federativas</caption>

        <thead>

            <tr>

                <th class="p1">Posição</th>

                <th class="p1">Estado</th>

            </tr>

        </thead>

Usamos **th** sempre que queremos colocar um titulo dentro da tabela, e usamos **td** sempre que queremos colocar um dado.

Para dizermos ao HTML que um titulo pertence a uma coluna atribuímos ao **thead (th)** a classe **scope=”col”**

Para dizermos ao HTML que um titulo pertence a uma linha usamos atribuímos ao **thead (th)** a classe **scope=”row” , exemplo abaixo:**

<table>

        <caption>População das unidades federativas</caption>

        <thead>

            <tr>

                <th class="p1" scope="col">Posição</th>

                <th class="p1" scope="col">Estado</th>

            </tr>

        </thead>

<tfoot>

            <tr>

                <th class="p1" scope="row">Total de habitantes</th>

                <td class="num">84.353.783</td>

            </tr>

        </tfoot>

Podemos aplicar um efeito **zebrado** a tabela aplicando propriedades as linhas **(tr)** que estão dentro de **tbody** (corpo), **exemplo abaixo:**

        tbody > tr:nth-child(odd) {

            background-color:lightgrey;

            font-weight: bold;

        }

        tbody > tr:nth-child(even) {

            background-color:white;

            text-decoration: underline;

        }

Para deixarmos o cabeçalho da tabela fixo enquanto rolamos a tabelas podemos usar a tag **position: relative** dentro da tabela **(table),** e posteriormente criamos uma declaração exclusiva para o thead **(th)** e trow **(tr),** inserindo a tag **position: sticky,** **exemplo abaixo:**

        table{

            width: 500px;

            border-collapse: collapse;

            text-align: left;

            font-family: cursive;

            position: relative;

        }

        thead > tr > th {

            position: sticky;

            top: 0;

            background-color: rgba(80, 77, 77, 0.966);

        }

Podemos usar a tag **colspan** (horizontal) para mesclarmos células dentro de uma coluna da tabela e usamos a tag **rowspan** (na vertical)para mesclarmos células dentro de uma linha da planilha, em ambas tag temos que dizer a quantidade de células que queremos que a declaração venha a ocupar, **exemplo abaixo**:

  <tr>

            <td>D</td>

            <td>E</td>

            <td rowspan="3">F</td>

        </tr>

        <tr>

            <td>A</td>

            <td colspan="2">B</td>

        </tr>

Podemos pintar as colunas de maneira individual usando a tag **colgroup** então dentro inserimos a tag **col** e lhe atribuímos uma **class,** lembrando que as ordens das colunas começam em ordem decrescente (de cima pra baixo 1, 2, 3 e afins....) **exemplo abaixo:**

        <colgroup>

            <col class="nome">

            <col class="sexo">

            <col class="idade">

            <col class="profissao">

        </colgroup>

        <caption>Cadastro</caption>

        <tr>

            <th scope="col">nome</th>

            <th scope="col">sexo</th>

            <th scope="col">idade</th>

            <th scope="col">profissão</th>

        </tr>

        }

        col.nome{

            background-color: chartreuse;

        }

        col.sexo{

            background-color: blue;

        }

        col.idade{

            background-color: burlywood;

        }

        col.profissao{

            background-color: darkgreen;

        }

Além disso, podemos usar a tag **span** e colocarmos quantas colunas queremos que tenham a mesma propriedade da atual, **exemplo abaixo:**

<col class="idade" span="2">

Conforme a viewport de exibição em que a tabela está sendo exibida vai diminuindo, e criado uma barra de rolagem na mesma onde acaba rolando não somente o conteúdo da tabela, mas como o conteúdo do texto também, para que a tabela se adapte a viewport podemos usar a tag **overflow**, lembrando que devemos selecionar toda nossa tabela e atribuirmos uma div a ela, então atribuímos uma class na qual vamos aplicar a tag overlfow, **exemplo abaixo:**

 <div id="container">

        <table>

            <caption>Tabelas Responsivas</caption>

            <thead>

                <tr>

                    <th>Produto</th>

 }

        div#container{

            overflow: auto;

        }

    </style>