Curso em vídeo.

HTML5 e CSS3

Endereços - github.com/gustavoguanabara & gustavoguanabara.github.io

**MÓDULO 01**

Idioma = pt-br

<hr> = linha horizontal

<br> = quebra parágrafo

<h1>

<p> = parágrafo

&lt; escrever a tag &gt; = mostrar a tag no navegador

&lt = less than

&gt = greater than

HTML entities = &euro

<!- - comentário - - > ou /\* \*/ = escrever comentário

Clique + ALT = apaga dois ao mesmo tempo

Apertar Tab = move para direita

Apertar Shift + Tab = move para esquerda

**EMOJIS**

Emojis = &#x + código referência emoji

<img> = imagem Texto alternativo (alt = definição escrita da imagem) ctrl + espaço dentro aspas

Head

Link.favicon. colocar favicon

Head

Dentro do body

H1, h2, h3, h4, h5, h6

<p> dentro do parágrafo escreve lorem = texto pronto

Fecha body

**TAGS**

&nbsp = espaço em branco

Tag b(não semãntico) / strong (negrito e destaque)

Tag i (não semântico) / em (itálico e ênfase) semântico

Seleciona o trecho + Ctrl + shift + P = wrap abreviation (ENVELOPAR)

<mark> = texto marcado

<big> <small> = texto grande e pequeno

<del> = texto deletado

<ins> = texto inserido

<u> = texto sublinhado (não semântico)

<sup> = texto sobreescrito

<sub> = texto subescrito

<code> = colocar código fonte

<pre>

<code>

<pre>

Do jeito acima aparece da forma correta, com espaços e indentações

Dentro da tag <pre> selecione o conteúdo dentro da tag <code> e aperte SHIFT + TAB e direcionar para esquerda ou somente TAB para direita

<q> = citações

<blockquote> = citações completas

<blockquote cite = “ URL” > = facilita para indicar os detentores dos direitos da citação

<abbr””> = colocar a definição da abreviação. Aparece no site quando encosta o mouse na abreviação.

<bdo> = inverter texto para esquerda e direita

**LISTAS ORDENADAS E NÃO ORDENADAS.**

<ol> = listas ordenadas. Colocar dentro de <ol> a tag <li>. Pode colocar <ol type “ A ou a ”> e <ol type “ I ou i”>

Ex:

<ol> ou <ul>

<li>texto</li>

<li>texto</li>

</ol> ou </ul>

<ol> ou <ul>

<li>texto</li>

<ol> ou <ul>

<li>Texto</li>

</ol> ou </ul>

</ol> ou </ul>

OBS = o </li> se tornou opcional.

<ul> = listas não ordenadas. Mesmo padrão acima, basta trocar pela tag <ul>.

Pode colocar <ul type= “disc, square ou circle”>

<ol type=” ” start=”número após o último da lista encima”>, quando quiser dar sequência a numeração da lista

Lista de definição = <dl>. Termo = <dt></dt>. Definição = <dd></dd>. Ajuda muito na questão de pesquisa do Google.

<dl>

<dt>Termo</dt>

<dd>Conceito do termo acima</dd>

</dl>

Para trocar todas as <ol></ol> ou <ul></ul> de uma vez basta clicar em cada tag segurando a tecla ALT.

**LINKS INTERNOS E EXTERNOS.**

<a href=”https://dominiosite.com/”>Acesse o site</a>. Serve para colocar hyperlinks ou links externos no site.

<a href=”https://dominiosite.com/” target=”\_blank” rel=”externos”>Acesse o site</a>. Ajuda a identificar e abrir uma nova aba.

Target = “\_blank” (abrir nova aba) rel=”external” (indica para o navegador que é link externo) ==> geralmente usado para links externos, sites que não sejam seu.

<a href=”https://dominiosite.com/” target=”\_blank” rel=”externos” hreflang=”en”>Acesse o site</a>. Ajuda para sites em outro idioma.

<a href=”arquivo criado dentro da pasta”>segunda página</a>.Link interno

não precisa https:/...

Página inicial para frente = rel=”next”

Páginas posteriores para voltar = rel=”prev”

Caso queira colocar links patrocinados = rel=”nofollow”

<a href=”arquivo criado dentro da pasta” rel=”next ou prev”>segunda página</a>. Semântica. Rel=’’next ou prev” ajudam a dizer para o Google a ordem correta. Rel=’’nofollow” relacionado com dispositivo de busca.Target=’’self” abre na mesma aba.

<a href=”../index.html”>voltar página principal</a>../ volta na pasta

Links download = <a href=”nome pasta/nome documento.formato” download = ‘’nomearquivo.formato’’ type = ‘’mediatype/formato’’ >Texto</a>

<a href=”#”>texto</a> = quando voce quiser fazer testes

**IMAGENS DINÂMICAS.**

<picture>

<img src=’’arquivo.formato’’ alt=’’definição’’>

<picture>

Obs: bounce rate = índice rejeição google

<picture>

<source media type = source media=’’(max/min width)’’ srcset=’’” type=’’image/formato arquivo’’>

max/min width = 1050px srcset=”imagem menor’’ type=’’image/png’’

Obs: começa pela imagem maior depois vai colocando o source em cima

<img src=’’arquivo.formato’’ alt=’’definição’’>

</picture>

**REPRODUZINDO ÁUDIO.**

<audio src =’’nome pasta/nome arquivo.mp3’’ controls autoplay><audio>

Formatos = MP3, WAV, OGG

Outro jeito:

<audio>

Source src’’nome pasta/nome arquivo’’ type’’audio/mpeg’’

Source src’’nome pasta/nome arquivo’’ type’’audio/OGG’’

Source src’’nome pasta/nome arquivo’’ type’’audio/WAV’’

</audio>

<audio preload=”metadata” ou “none” “auto”; autoplay controls loop></audio>

**INSERINDO VÍDEOS.**

<video controls width”tamanho px” poster”nome pasta/nome arquivo” autoplay loop>

Source src’’nome pasta/nome arquivo’’ type’’video/m4v ou mp4’’

Source src’’nome pasta/nome arquivo’’ type’’video/OGG’’

Source src’’nome pasta/nome arquivo’’ type’’video/webm’’

</video>

Inserir vídeos externos = clicar em <incorporar>, copiar e Ctrl+V

**CSS.**

Inline= na própria linha

<body style=”background color e/ou font size e/ou font family”>

<h1 style= “color”>Texto</h1>

<h2>Texto</h2>

<p>lorem</p>

</body>

Estilos locais ou internos

<head>

<style>

Body{

background color;

}

H1{

Color: red;

}

P{

Textalign: justify;

}

</style>

</head>

Estilos externos.

CTRL + clique mouse no link dentro do <a href /> = cria o arquivo

<a href=”pagina02.html”>Ir para a página dois</a> Ctrl+clique no link em destaque para criar novo arquivo.

Criando arquivo CSS = link rel=”stylesheet” href=”nomearquivo.css”> Deve ser alocado entre a tag <head></head> Ctrl+ clique no link em destaque para criar o arquivo.

Início do arquivo CSS = @charset ”UTF-8”; indicando compatibilidade com acentuação. **Também chamadas de regras**.

No mesmo arquivo pode conter dois links CSS, CSS interno e Inline.

===============================//==================================

**MÓDULO 02**

Psicologia das cores.

**CORES EM CSS.**

Representação por nome:

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

Representação por códigos hexadecimais:

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 - colour : #0000ff; color :#ffffff;” > Exemplo de cores </h2>

OBS: #0000ff = dois primeiros zeros (red) dois segundos zeros (greeen) ff máximo 255

Representação em RGB:

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

OBS: RGB = red, green, blue

Representação por HSL:

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

OBS: HSL = hue(matiz), saturation, luminosity.

OBS: RGBA = RGB com alpha (transparência) / HSLA = HSL com alpha (transparência).

**HARMONIA DE CORES.**

Círculo cromático.

Paletas de cores = 3 a 5 cores.

Cores primárias = amarelo, vermelho e azul.

Cores secundárias = laranja, verde e violeta.

Cores terciárias = amarelo - esverdeado, amarelo - alaranjado, vermelho - alaranjado, vermelho - arroxeado, azul - arroxeado e azul - esverdeado.

Hemisfério da esquerda = cores frias

Hemisfério da direita = cores quentes

Cores complementares = roxo - amarelo, vermelho - verde. Uma linha reta que liga duas cores opostas no círculo cromático( contraste).

Cores análogas = as cores vizinhas no círculo cromático.

Cores análogas e uma complementar = cores vizinhas com uma reta oposta indicando uma cor para contrastar.

Cores análogas relacionadas = duas cores vizinhas pula uma cor e escolhe a seguinte.

Cores intercaladas = escolhe uma cor, pula a seguinte, escolhe a próxima cor, pula a seguinte e escolhe uma cor.

Cores triádicas = escolhe uma cor, pula três, escolhe outra cor, pula três e escolhe uma cor.

Cores em quadrado = escolhe uma cor, pula duas, escolhe uma cor, pula duas, escolhe uma cor, pula duas, escolhe uma cor e pula duas.

Cores tetrádicas = escolhe uma cor e escolhe a oposta, escolhe uma cor e escolhe a oposta.

Monocromia = escolhe a cor no círculo cromático e modifica a saturação e luminosidade da cor (efeito degradê).

Sites = Coolors, Adobe Colors,

Captura de cor em sites = instalar plugin Colorzilla no Chrome.

**GRADIENTE EM CSS.**

<head>

<style>

\*{ = configuração global

\*{

Height: 100%,

Body {

Background - image: linear – gradient (to top, white, blue); OBS = dentro do parênteses voce coloca a direção [ to right, to bottom, to left, to top] ou [90deg, 45 deg, -45deg, -90deg] e as cores.

Background attachment: fixed;

}

</style>

</head>

EXEMPLO CSS:

@charset "UTF-8";

\* {

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

}

p {

    text-align: justify;

}

body {

    background-image: linear-gradient(to right, #A36E4C, #8D5023);

}

main {

    background-color: white;

    width: 600px;

    padding: 10px;

    margin: auto;

    border-radius: 10px;

    box-shadow: 5px 5px 15px rgba(119, 67, 21, 0.635);

}

h1 {

    color: #211D1A;

    text-align: center;

    text-shadow: 1px 1px 1px rgba(33, 29, 26, 0.781);

}

h2 {

    color: #9F968F;

}

**TIPOGRAFIA**

Tipos métricos = altura das maiúsculas, altura x, ascendentes, descendentes e corpo.

Componentes anatômicos geométricos = serifa, haste, filete, arco, esporão, vértice.

Componentes anatômicos de letras = pé, barriga, braço, orelha, cauda, perna, espinha e olho.

Glifo = letra

Conjunto de glifos = fonte

Família tipográfica

CATEGORIAS DE FONTES.

Serifadas, não serifadas, monoespaçadas, handwriting e display.

Web safe font combinations = site W3

EX:

<style>

        body{

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

        }

      h1, h2{

            font-family: monospace;

        }

</style>

Medidas absolutas = cm (centímetro), mm (milímetro), in (polegada), px (pixel), pt (ponto), pc (paica). Ponto e paica não são recomendados para exibições em tela.

Medidas relativas = em (tamanho atual da fonte), ex ( altura x de uma fonte), rem (fonte configurada no <body>), vw(view width = largura do view port ou tamanho da tela), vh(view height = altura da view port), % (porcentagem).

1 em = 16 px

Medidas oficialmente recomendadas pela W3C = px ou em.

**Peso, estilo e shorthand font.**

Weight, width e height.

Font-weight = lighter, normal, bold e bolder ou escolher de 100 a 900.

Font-style = normal ou italic.

Text decoration = underline.

SHORT HAND FONT segue essa ordem = font-style —> font-weight —> font-size —> font-family.

EX: font: italic bolder 3 em “work-sans”, sans-serif;

Google Fonts —> seleciona a fonte —> select style —> import —> ctrl +c no <style> + ctrl +c CSS rules

<style>

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

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

        body{

        font-family: 'Alumni Sans Inline One', cursive;

        font-size: 2em;

        color: black;

        }

        h1{

        font-family: 'Rubik Dirt', cursive;

        font-size: 3em;

        color: black;

        }

</style>

**Fontes externas = Dafont.com**

Tipos de FORMAT():

* Opentype (OTF)
* Truetype (TTF)
* Embedded-opentype
* Truetype-aat (Apple Advanced Tipography)
* Svg

EX:

<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>Fontes externas</title>

    <style>

        @font-face {

            font-family: 'war';

            src: url(*fontes/VIKING-N.TTF*) format('truetype');

            font-weight: normal;

            font-style: normal;

        }

        body{

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

        }

        h1{

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

            font-size: 3em;

            font-weight: normal;

        }

    </style>

</head>

**OBS**: descobrir a fonte usada em um texto de um site = Fonts Ninja (extensão Google Chrome).

**OBS**: descobrir a fonte usada em um texto numa imagem de um site = Whatfontis.com, Fontsquirrel, Myfont.com.

**Alinhamento de textos com CSS.**

Text align = left, center , right, justify.

<!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>Alinhamentos</title>

    <style>

        body{

            font: normal 16px Arial, Verdana, serif;

            text-align: left;

        }

        h1{

            text-align: center;

        }

        h2{

            text-align: left;

        }

        p{

            text-align: justify;

            text-indent: 30px;

        }

    </style>

</head>

**Seletores personalizados.**

Colocar o “id=principal” depois da tag para personalizá-lo. Utilizar # na CSS.

<!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">

    <link *rel*="stylesheet" *href*="style.css">

    <title>Seletores personalizados</title>

</head>

<body>

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

    <h1>Aprendendo HTML</h1>

    <h2>HTML básico</h2>

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

STYLE CSS

@charset "UTF-8";

/\*

Em HTML é id = em CSS é #

Em HTML é class = em CSS é .

\*/

body{

    background-color: rgb(241, 90, 120);

    font-family: Verdana, Geneva, Tahoma, sans-serif;

}

h1*#principal*{

    text-align: center;

    background-color: crimson;

    color: white;

}

h1{

    color: rgb(230, 8, 53);

}

h2{

    color: rgb(175, 49, 75);

}

**ID e Classes.**

ID é único. Classepode ser múltiplo.

Consigo utilizar mais de uma classe no mesmo elemento basta colocar espaço sem vírgulas. EX: <h1 class=”basico avancado”>texto</h1>

Consigo colocar um ID mais uma classe no mesmo elemento. EX: <h1 = “principal” class “basico”>texto</h1>

HTML

<!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">

    <link *rel*="stylesheet" *href*="style.css">

    <title>Seletores personalizados</title>

</head>

<body>

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

    <h1 *class*="destaque">Aprendendo HTML</h1>

    <h2 *class*="basico">HTML básico</h2>

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

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

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

    <h2 *class*="avancado">HTML avançado</h2>

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

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

    <h2 *class*="basico">CSS básico</h2>

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

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

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

    <h2 *class*="avancado">CSS avançado</h2>

    <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Temporibus voluptatum necessitatibus illo blanditiis. Ipsam eum amet illo et recusandae beatae repellat ullam itaque fugiat tenetur corrupti, atque architecto rem! Fuga!</p>

</body>

</html>

STYLE CSS

@charset "UTF-8";

/\*

Em HTML é id = em CSS é # . (só pode um elemento)

Em HTML é class = em CSS é . (pode vários elementos)

\*/

body{

    background-color: #E64F3B;

    font-family: Verdana, Geneva, Tahoma, sans-serif;

}

h1*#principal*{

    text-align: center;

    background-color: #BD4131;

    color: white;

}

h1{

    color: #3D1510;

}

h2{

    color: #802C21;

}

*.basico*{

    color: green;

}

*.intermediario*{

    color: yellow;

}

*.avancado*{

    color: red;

}

*.destaque*{

    color: darkmagenta;

}

**Pseudoclasses.**

Em HTML é pseudoclasse. Em CSS é **:**

Pseudoclasses são palavras que definem estados de elementos dentro da HTML.

Exemplos de pseudoclasses = hover/visited/active.

Exemplo 1.

<!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>Pseudo-classes</title>

    <style>

        div {

            background-color: lightblue;

            text-align: center;

            height: 200px;

            width: 200px;

            border: 1px solid black;

            display: inline-block;

        }

        div*:hover* {

            background-color: tomato;

        }

    </style>

</head>

<body>

    <div *id*="d1">01</div>

    <div *id*="d2">02</div>

    <div *id*="d3">03</div>

</body>

</html>

Exemplo 2.

<!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>Exemplo de Hover</title>

    <style>

        body {

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

        }

        div*:hover* {

            color: blue;

        }

        div > p {

            display: none;

        }

        div*:hover* > p {

            display: block;

            color: white;

            background-color: blue;

            width: 200px;

        }

    </style>

</head>

<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>

    <p>Fim do exemplo</p>

</body>

</html>

**Modelo de caixas.**

HTML e CSS box model.

Aninhamento.

Height. Width.

Border = linha que circunda seu conteúdo.

Padding = engordar o elemento para que a borda fique distante do conteúdo (espaço interno).

Margin = da borda para fora (espaço externo).

Outline = contorno ou traçado. Fora do elemento.

TIPOS DE CAIXA.

Box-level / Inline level.

Box-level:

Sempre se inicia em uma linha nova.

Sempre ocupa a largura total da tela.

EX: <div>, <h1> <h6>, <p>, <main>, <header>, <nav>, <article>, <aside>, <footer>, <form>, <video>.

Inline-level:

Não quebra a linha. Dá continuidade do conteúdo na mesma linha.

Não ocupa o tamanho inteiro da tela.

EX: <span>, <a>, <code>, <small>, <strong>, <em>, <sup>, <sub>, <label>, <button>, <input>, <select>.

DEV TOOLS CHROME

User agent = meu navegador.

Border style = solid, dashed, dotted, double

EX:

<!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>Modelos de caixa</title>

    <style>

        h1{

            background-color: lightgreen;

            height: 300px;

            width: 300px;

            border-width: 10px;

            border-style: solid;

            border-color: darkgreen;

            padding-top: 10px;

            padding-bottom: 10px;

            padding-left: 10px;

            padding-right: 10px;

        }

        a{

            border-width: 10px;

            border-style: dashed;

            border-color: yellowgreen;

            padding-top: 10px;

            padding-bottom: 10px;

            padding-left: 10px;

            padding-right: 10px

        }

    </style>

</head>

<body>

    <h1>Exemplo de caixas box-level</h1>

    <p>Parágrafos também são exemplos de box-level, mas os <a *href*="#">links são exemplos de caixas de inline level.</a>Vamos ver como tudo isso funciona. </p>

</body>

</html>

OBS: ordem para configuração do padding ou outro elemento => top, right, bottom e left (sentido horário).

Margin: auto; (usado para centralizar).

Outline-width: 10px;

Outline-style: 10px;

Outline-color: 10px;

Display: block;

Display: inline;

EX:

<!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>Modelos de caixa</title>

    <style>

        h1{ /\* box-level\*/

            background-color: lightgreen;

            height: 300px;

            width: 300px;

            /\*border-width: 10px;

            border-style: solid;

            border-color: darkgreen;\*/

            border: 10px solid darkgreen;

            /\*margin-top: 10px;

            margin-right: 10px;

            margin-bottom: 10px;

            margin-left: 10px;\*/

            margin: 20px;

            /\*padding-top: 10px;

            padding-right: 10px;

            padding-bottom: 10px;

            padding-left: 10px;\*/

            padding: 10px;

            /\*outline-width: 10px;

            outline-style: dashed;

            outline-color: yellow;\*/

            outline: 10px dashed yellow;

        }

        a{ /\* inline level\*/

            /\*border-width: 10px;

            border-style: dashed;

            border-color: yellowgreen;\*/

            border: 10px dashed yellowgreen;

            /\*padding-top: 10px;

            padding-right: 10px;

            padding-bottom: 10px;

            padding-left: 10px;\*/

            padding: 10px;

        }

    </style>

</head>

<body>

    <h1>Exemplo de caixas box-level</h1>

    <p>Parágrafos também são exemplos de caixas box-level, mas os <a *href*="#">links são exemplos de caixas de inline level.</a>Vamos ver como tudo isso funciona. </p>

</body>

</html>

**Grouping tags.**

<Header>

<Nav>

<Main>

<Article>

<Aside>

<Footer>

}

<!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>Grouping tags</title>

    <style>

        body{

            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

            background-color: steelblue;

            margin: 0px;

            text-align: justify;

        }

        header{

            background-color: lightskyblue;

            padding: 10px;

            margin: 10px;

        }

        nav{

            background-color: rgb(158, 158, 158);

            padding: 3px;

        }

        nav > a{

            text-decoration: none;

            color: blue;

            font-weight: bold;

            margin-right: 30px;

        }

        nav > a*:hover*{

            text-decoration: underline;

        }

        main{

            background-color: lightsteelblue;

            padding: 10px;

            margin: 10px;

        }

        article{

            background-color: lightskyblue;

            padding: 5px;

        }

        article > aside{

            background-color: rgb(158, 158, 158);

        }

        footer{

            background-color: black;

            text-align: center;

            color: white;

            padding: 1px;

            margin: 0px;

        }

    </style>

</head>

<body>

    <header>

        <h1>Meu Site</h1>

        <nav>

            <a *href*="#">Link</a>

            <a *href*="#">Link</a>

            <a *href*="#">Link</a>

            <a *href*="#">Link</a>

        </nav>

    </header>

    <main>

        <section *id*="assuntos">

            <article>

                <h2>Principais cotações</h2>

                <p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. Officia cum ipsum dignissimos! Id, ea, amet quod cumque voluptatem facilis, velit dolore possimus quidem accusamus cupiditate delectus tempora similique? Veniam, saepe.</p>

                <aside>

                    <p>T3 Capital</p>

                </aside>

            </article>

        </section>

        <section *id*="noticias">

            <article>

                <h2>Noticias sobre o Brasil</h2>

                <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Cum delectus culpa asperiores quis odio commodi, ipsa modi aliquam veniam alias odit dolorem nulla, facilis magnam esse atque, consectetur distinctio ex?</p>

                <aside>

                    <p>Artigo escrito por Karl Gunter</p>

                </aside>

            </article>

            <article>

                <h2>Noticias no exterior</h2>

                <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Dignissimos asperiores eveniet ab sunt ipsam iusto suscipit, quo voluptatum odio, aliquid pariatur rem. Iste vitae error officia eaque praesentium quibusdam assumenda!</p>

                <aside>

                    <p>Fonte KWW</p>

                </aside>

            </article>

        </section>

    </main>

    <footer>

        <p>Desenvolvido por Klotz produções</p>

    </footer>

</body>

</html>

**Sombras.**

Boxshadow: 1px 1px 1px 1px black; =>horizontal, vertical, blur, spread, cor, inset, outset. (shorthand)

Pode ser feito pelo DevTools.

**Bordas.**

Border radius: 1px ou 1px 1px 1px 1px; => border top left radius, border top right radius, border bottom right radius, border bottom left radius.

OBS: pode-se usar porcentagem. EX: border radius: 50%;

EX 02: HTML => <div id=”bola”</div>

CSS => div#bola{

Height: 100px;

Width: 100px;

Margin: 10px;

Border radius: 50%;

Background color: white;

}

**Bordas personalizadas ou decoradas.**

<style>

        h1{

            border: 20px solid black;

            padding: 10px;

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

            border-image-slice: 27;

            border-image-repeat: repeat;

            /\*

            SHORTHAND

            border-image: url("borda.png") 27 repeat;

            \*/

        }

    </style>

**Projeto inicial.**

Site Mockflow = criação de wireframe.

Ctrl + Shift + = Zoom in

Ctrl + Shift - = Zoom out

Inkscape

Configurações globais = :root{} \*{}

**HTML**

<!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>Como surgiu o mascote do Android?</title>

    <link *rel*="shortcut icon" *href*="imagens/favicon.ico" *type*="image/x-icon">

    <link *rel*="stylesheet" *href*="estilo/style.css">

</head>

<body>

    <header>

        <h1>Curiosidades de tecnologia</h1>

        <p>Tudo aquilo que voce sempre quis saber sobre o mundo Tech, em um único lugar.</p>

    </header>

    <nav>

        <a *href*="#">Home</a>

        <a *href*="#">Notícias</a>

        <a *href*="#">Curiosidades</a>

        <a *href*="#">Fale conosco</a>

    </nav>

    <main>

        <article>

            <h1>História do Mascote do Android</h1>

            <p>Provavelmente você sabe que o sistema operacional <strong>Android</strong>, mantido pelo <strong>Google</strong> é um dos mais utilizados para dispositivos móveis em todo o mundo. Mas tavez você não saiba que o seu simpático mascote tem um nome e uma história muito curiosa? Pois acompanhe esse artigo para aprender muita coisa sobre esse robozinho.</p>

            <h2>A primeira versão</h2>

            <p>A primeira tentativa de criar um mascote surgiu em 2007 e veio de um desenvolvedor chamado <a *href*="https://androidcommunity.com/dan-morrill-shows-us-the-android-mascot-that-almost-was-20130103/" *target*="\_blank" *class*="externo">Dan Morrill</a>. Ele conta que abriu o <a *href*="https://inkscape.org/pt-br/" *target*="\_blank" *class*="externo">Inkscape</a> (software livre para vetorização de imagens) e criou sua própria versão de robô. O objetivo era apenas personificar o sistema apenas para a a sua equipe, não existia nenhuma solicitação da empresa para a criação de um mascote.</p>

            <picture>

                <source *media*="(max-width: 600px)" *srcset*="imagens/dan-droids-pq.png">

                <img *src*="imagens/dan-droids.png" *alt*="Primeiro mascote do Android">

            </picture>

            <p>Essa primeira versão bizarra até foi batizada em homenagem ao seu criador: seriam os <strong>Dandroids</strong>.</p>

            <h2>Surge um novo mascote</h2>

            <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/android" *target*="\_blank" *class*="externo">Irina Blok</a>, também funcionária do <strong>Google</strong>, ficou com a missão de representar o pequeno robô de uma maneira mais agradável.</p>

            <picture>

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

            <img *src*="imagens/irina-blok.jpg" *alt*="Irina Blok, criadora do Bugdroid">

            </picture>

            <p>A ideia principal da Irina era representar tudo graficamente com poucos traços e de forma mais chapada. O desenho também deveria gerar identificação rápida com quem o olha. Surgiu então o <strong>Bugdroid</strong>, o novo mascote do Android.</p>

            <img *src*="imagens/bugdroid.png" *class*="pequena" *alt*="Bugdroid">

            <p>A principal inspiração para os traços do novo <strong>Bugdroid</strong> veio daqueles bonequinhos que ilustram portas de banheiro para indicar o gênero de cada porta. Conta a lenda que a artista estava criando em sua mesa no escritório do <strong>Google</strong> e olhou para o lado dos banheiros e a identificação foi imediata: simples, limpo, objetivo.</p>

            <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" *allowfullscreen*></iframe>

            </div>

            <aside>

                <h3>Quer aprender mais?</h3>

                <p>Outro assunto curioso em relação ao Android é que cada versão sempre foi nomeada em homenagem a um doce, em ordem alfabética a partir da versão 1.5 até a 9.0.</p>

                <ul>

                    <li>1.5 - <abbr *title*="Bolo de copo">Cupcake</abbr></li>

                    <li>1.6 - <abbr *title*="Rosquinha">Donut</abbr></li>

                    <li>3.0 - <abbr *title*="Bomba">Eclair</abbr></li>

                    <li>2.2 - <abbr *title*="Iogurte congelado">Froyo</abbr></li>

                    <li>2.3 - <abbr *title*="Biscoito de gengibre">Gingerbread</abbr></li>

                    <li>3.0 - <abbr *title*="Favo de mel">Honeycomb</abbr></li>

                    <li>4.0 - <abbr *title*="Sanduíche de sorvete">Ice Cream Sandwich</abbr></li>

                    <li>4.1 - <abbr *title*="Jujuba">Jelly Bean</abbr></li>

                    <li>4.4 - <abbr *title*="Chocolate">KitKat</abbr></li>

                    <li>5.0 - <abbr *title*="Pirulito">Lolipop</abbr></li>

                    <li>6.0 - <abbr *title*="Marshmallow">Marshmallow</abbr></li>

                    <li>7.0 - <abbr *title*="Torrone">Nougat</abbr></li>

                    <li>8.0 - <abbr *title*="Biscoito">Oreo</abbr></li>

                    <li>9.0 - <abbr *title*="Torta">Pie</abbr></li>

                </ul>

                <p>Infelizmente, o <strong>Android Q</strong> não existiu, pois o <strong>Google</strong> resolveu pôr fim a essa divertida prática e começou a usar numerações, o que deu origem ao Android 10.</p>

                <p>Acesse aqui o site <a *href*="#">Android History</a> para conhecer a sequência das versões "adocicadas" e o que cada uma trouxe para o sistema Android.</p>

            </aside>

            <p>Então é isso! Espero que você tenha gostado do nosso artigo com essa curiosidade sobre o sistema <strong>Android</strong> e seu simpático mascote.</p>

        </article>

    </main>

    <footer>

        <p>Site criado por <a *href*="#">Klung&Klotz</a>.</p>

    </footer>

</body>

</html>

**CSS**

@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;

}

*:root* {

    --cor0: #c5ebd6;

    --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;

}

\* {

    margin: 0px;

    padding: 0px;

}

body {

    background-color: var(--cor0);

    font-family: var(--fonte-padrao);

}

a*.externo::after* {

    content: '\1F517';

}

header {

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

    min-height: 150px;

    text-align: center;

    padding-top: 20px;

}

header > h1 {

    color: white;

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

    margin-bottom: 20px;

    font-size: 3em;

    font-weight: normal;

    text-shadow: 2px 2px 0px black;

}

header > p {

    font-family: var(--fonte-padrao);

    font-size: 1.2em;

    color: white;

    max-width: 600px;

    margin: auto;

    margin-bottom: 20px;

    padding-right: 10px;

    padding-left: 10px;

    text-shadow: 2px 2px 0px black;

}

nav {

    background-color: var(--cor5);

    padding: 10px;

    box-shadow: 0px 7px 7px rgba(0, 0, 0, 0.192);

}

nav > a {

    color: var(--cor1);

    padding: 10px;

    border-radius: 5px;

    text-decoration: none;

    font-weight: bold;

}

nav > a*:hover* {

    background-color: var(--cor3);

    color: var(--cor5);

}

main {

    min-width: 300px;

    max-width: 1000px;

    background-color: white;

    margin: auto;

    padding: 20px;

    margin-bottom: 30px;

    border-bottom-left-radius: 10px;

    border-bottom-right-radius: 10px;

}

main img{

    width: 100%;

}

main img*.pequena*{

    max-width: 350px;

    margin: auto;

    display: block;

}

main h1 {

    color: var(--cor5);

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

    font-weight: normal;

    font-size: 1.8em;

}

main h2 {

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

    font-weight: normal;

}

main p {

    margin: 15px 0px 15px 0px;

    text-align: justify;

    text-indent: 20px;

    line-height: 2em;

    font-size: 1em;

}

main a {

    text-decoration: none;

    font-weight: bold;

}

main a*:hover* {

    text-decoration: underline;

}

div*.video* {

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

    padding: 20px;

    position: relative;

    padding-bottom: 57%;

}

div*.video* > iframe {

    position: absolute;

    top: 5%;

    left: 5%;

    width: 90%;

    height: 90%;

}

aside {

    background-color: var(--cor2);

    padding: 10px;

    border-radius: 10px;

}

aside > h3 {

    background-color: var(--cor4);

    color: white;

    padding: 10px;

    margin: -10px -10px 0px -10px;

    border-radius: 10px 10px 0px 0px;

}

aside > ul {

    list-style-type: '\2714\00A0\00A0';

    list-style-position: inside;

    columns: 2;

}

footer {

    background-color: var(--cor5);

    color: white;

    text-align: center;

    font-size: 1em;

    padding: 5px;

}

footer a {

    color: white;

    font-weight: bolder;

    text-decoration: none;

}

footer a*:hover* {

    text-decoration: underline;

    color: black;

}

=====================================//==================================

**MÓDULO 3:**

Versionamento de software.

Git => mantém versões do site dentro do seu computador. (Repositório local)

Github => mantém um backup das versões na nuvem. (Repositório remoto)

Commit = primeiro

Push = depois