**Capítulo 14 Aula 07 – Usando fontes externas baixadas**

Para utilizar fontes baixadas, deve-se baixar o arquivo (algumas fontes vem com arquivo otf, outras ttf, outras com os dois). Depois disso criar um parâmetro dentro de style no head, a parte de format é opentype para otf e truetype para ttf.

 <style>

        body{

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

            font-size: 1em;

        }

        h1{

            font-family: 'Angel', serif;

            font-size: 3em;

        }

        @font-face {

            font-family: 'Angel';

            src: url(fontes/Angel\ wish.otf) format(opentype);

            src: url(fontes/Angel\ wish.ttf) format(truetype);

            font-weight: normal;

            font-style: normal;

        }

    </style>

# Capítulo 14 Aula 08 – Capturando as fontes usadas em um site

Usar extensão do Google chamada Fontes Ninja

# Capítulo 14 Aula 09 – Detectando fontes dentro de imagens

Usar o whatfontis.com, fontsquirrel.com ou myfonts.com

# Capítulo 15 Aula 01 – Usando o id com CSS

Para criar um estilo específico para algo, como por exemplo, existem vários H1 no documento e quero que apenas um deles seja diferente, criamos um id para ele, identificando de forma diferente. Id só pode ser usado para um elemento.

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

E na página de CSS identificamos com hashtag e criamos o estilo.

h1#principal{

    text-align: center;

    background-color: rgb(41, 170, 52);

    color: white;

}

# Capítulo 15 Aula 02 – As diferenças entre id e class

Como apenas um elemento pode ser identificado por Id, se quiser uma configuração para vários elementos, usa-se o class.

 <h2 class="basico">HTML básico</h2>

Na página de CSS identificamos por ponto.

.basico{

    color: rgb(41, 199, 54);

    font-weight: bolder;

}

É possível adicionar mais de uma classe no elemento.

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

Também é possível adicionar id e class no mesmo elemento, porém o id sobrepõe se for uma mesma configuração, por exemplo a cor, vai prevalecer o id.

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

# Capítulo 15 Aula 03 – Pseudo-classes em CSS

As pseudo-classes representam estados de elementos e podem envolver qualquer elemento, até mesmo classes e id. Nesse exemplo, usa-se hover, que seria a div (caixa, espaço) dicar amarela quando passar o mouse por cima. As pseudo-classes são identificadas por dois pontos.

 div:hover{

            background-color: yellow;

        }

Extra:

-ficar todos na mesma linha

-cor do fundo

-configurações de borda

-alinhamento do texto

-altura

-largura

 div{

            display: inline-block;

            background-color: lightgrey;

            border: 1px solid black;

            text-align: center;

            height: 200px;

            width: 200px;

        }

Nesse exemplo fizemos um texto escondido, em um parágrafo dentro da div, que se passar o mouse em cima iria mostrar. Primeiro identificamos o parágrafo dentro da div, como filho da div, div > p. Colocamos para não mostrar, em seguida, usando a pseudo-classe ele iria mostrar.

  <style>

        body{

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

        }

        div > p{

            display: none;

        }

        div:hover > p{

            display: block;

            color: white;

            background-color: red;

            width: 200px;

        }

        div:hover{

            color: red;

        }

    </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>

# Capítulo 15 Aula 04 – Pseudo-elementos em CSS

Duas novas pseudo-classes, a:visited (o que fazer quando o link já foi visitado), a:active (como o link fica quando clicar).

Pseudo-elementos adicionam conteúdo em qualquer elemento, são indicados por duas vezes os dois pontos.

 a::after{

            content: '\27F8';

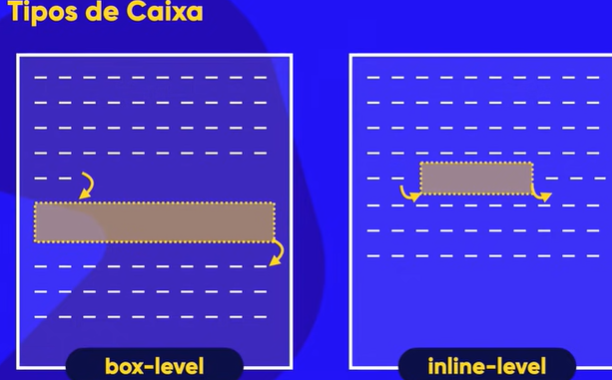
        }

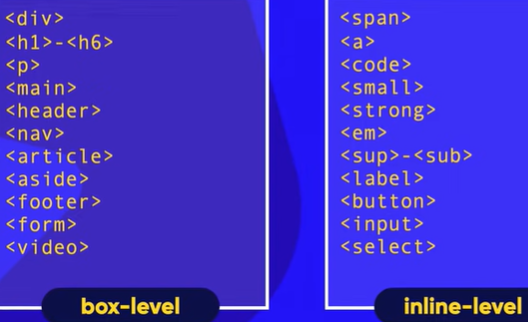
# Capítulo 16 Aula 01 – Modelo de Caixas: primeiros passos











# Capítulo 16 Aula 03 – Modelo de Caixas na prática (Pt.2)

Comandos para configuração de border, padding, margin e out-line. Caso queira a caixa centralizada, usa-se

margin: auto;

h1{

            background-color: lightgray;

            height: 300px;

            width: 300px;

            border-width: 5px;

            border-style: solid;

            border-color: rgb(2, 2, 119);

            padding-top: 10px;

            padding-right: 10px;

            padding-bottom: 10px;

            padding-left: 10px;

            margin-top: 10px;

            margin-right: 10px;

            margin-bottom: 10px;

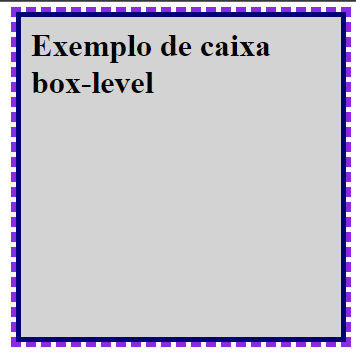
            margin-left: 10px;

            outline-width: 5px;

            outline-style: dashed;

            outline-color: blueviolet;

        }



Simplificando as configurações com shorthands:

  h1{

            background-color: lightgray;

            height: 300px;

            width: 300px;

            border: 5px solid rgb(2, 2, 119);

            padding: 10px 10px 10px 10px;

            margin: 10px 10px 10px 10px;

            outline: 5px dashed darkmagenta;

        }

Em relação às propriedades com tamanhos, se utilizar um valor só, será atribuído para os 4 lados. Se colocar dois valores, será compensado como de cima=de baixo, direita=esquerda.

margin: 10px;

Para mudar a característica da caixa, como ser inline ou ou box level, usa-se o comando display.

display:block /\*box-level\*/

display:inline /\*inline-level\*/

# Capítulo 17 Aula 6 – Variáveis em CSS

Criar um padrão de cores e fontes para ser usado várias vezes no projeto.

:root{

    --cor0: #c5ebd6;

    --cor1: #83e1ad;

    --cor2: #3ddc84;

    --cor3: #2fa866;

    --cor4: #1a5c37;

    --cor5: #063d1e;

    --fonte-padrao:arial, verdana, helvetica, sans-serif;

    --fonte-destaque:"Bebas Neue", sans-serif;

    --fonte-android:'Android', sans-serif;

}

E para ser usado já aparecem quando tentar selecionar a cor ou a fonte

body{

    background-color: var(--cor0);

    font-family: var(--fonte-padrao);

}

header{

    background-color: var(--cor4);

}

Criar uma configuração geral para todos os elementos HTML do projeto:

\*{

    margin: 0px;

    padding: 0px;

}

# Capítulo 17 Aula 8 – Responsividade na prática

Para deixar o conteúdo escrito resposivo, usa-se max-width e min-width, os valores usados são perto do padrão normalmente usado por conta dos tamanho das telas. E o margin: auto é para centralizar o conteúdo.

main{

    background-color: white;

    min-width: 300px;

    max-width: 1000px;

    margin: auto;

No caso das imagens para ficarem responsivas usa-se width: 100% e caso esteja muito grande e queira mudar o tamanho é so usar max-width Se precisar centralizar a imagem, é necessário usar display:block porque a imagem naturalmente não é um bloco.

main img{

    width: 100%;

}

main img.pequena{

    max-width: 350px;

    display: block;

    margin: auto;

}

Se for necessário fazer com que quando diminuir a tela apareça outra imagem é preciso usar o source:media, envelopando em picture da seguinte forma:

            <picture>

                <source media="(max-width: 670px)" srcset="imagens/dan-droids-pq.png">

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

            </picture>

# Capítulo 17 Aula 10 – Melhorando o formato do conteúdo

**Conteúdo à parte:** Na hora de criar seletores de CSS criamos main > p {} sugerindo que é um parágrafo diretamente dentro do main, se criar main p {} significa que é qualquer parágrafo dentro de main que inclusive pode estar dentro de um article que está dentro do main.

header > h1{

    color: white;

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

    font-size: 3em;

    margin-bottom: 20px;

    text-shadow: 2px 2px 0px rgba(0, 0, 0, 0.301);

}

main p{

    margin: 15px 0px;

}

Text-indent para criar um recuo no inicio do parágrafo e line height para criar um espaçamento esntre as linhas e facilitar a leitura.

main p{

    margin: 15px 0px;

    text-align: justify;

    text-indent: 30px;

    line-height: 2em;

}

# Capítulo 17 Aula 12 – Tornando um vídeo responsivo

Quando for necessário colar uma div que está dentro de outra, ou seja, expandir, usa-se margem negativa:

div.video{

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

}

Adaptação e responsividade de vídeo importado

div.video{

    background-color: var(--cor5);

    margin: 0px -20px 30px -20px;

    padding-bottom: 57.5%;

    position: relative;

}

div.video > iframe{

    position: absolute;

    top: 5%;

    left: 5%;

    width: 90%;

    height: 90%;

}