

HTML 5 Responsivo

Criando Sites Responsivos Com HTML5 do Zero

CAPITULO 4 – Começando com CSS 3

Autores:

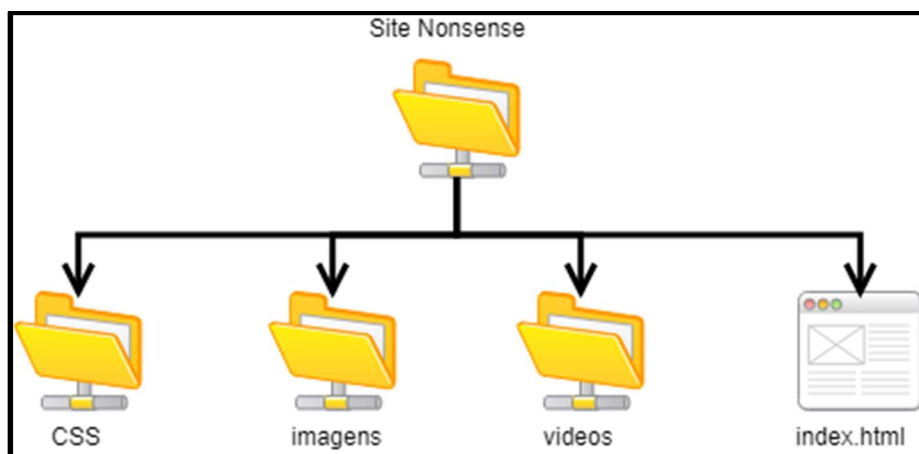
Caique Zaneti Kirilo

Luiz Carlos Machi Lozano

SÃO PAULO, 2019

Passo 36 – Criando um novo projeto

Chegou o momento de nós sermos apresentados formalmente para alguém que já ouvimos falar anteriormente, mas não havíamos nos encontrado pessoalmente. Para que ocorra tudo certo no nosso primeiro encontro, vamos preparar o ambiente criando um novo projeto e organizando as nossas pastinhas desta forma:

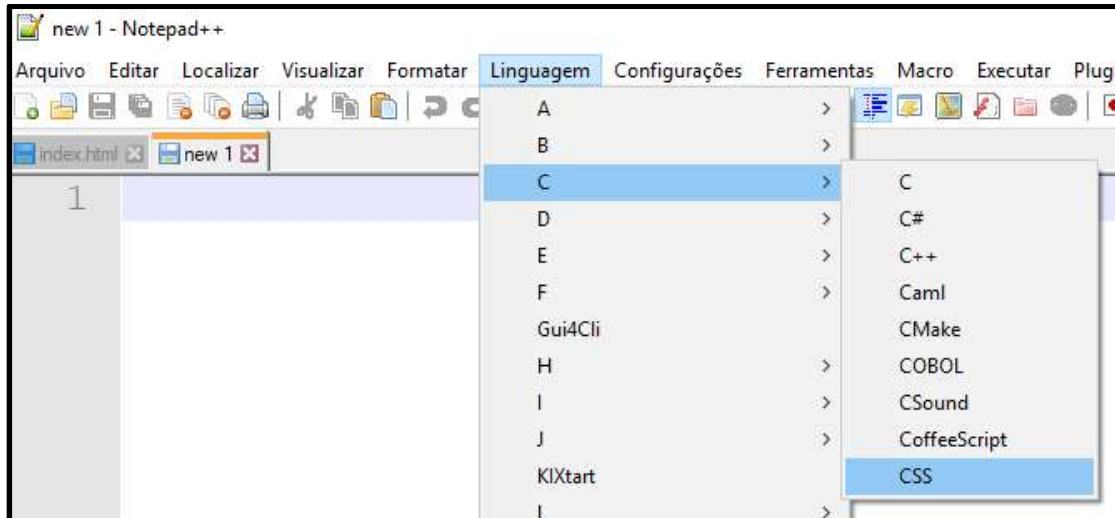


Dentro do nosso novo arquivo index vamos colocar um código com a estrutura básica do HTML:


```
index.html x
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <title>Nonsense</title>
5     <meta charset="utf-8">
6   </head>
7
8   <body>
9
10  </body>
11 </html>
```

Quando o objetivo é garantir uma formatação uniforme em todas as páginas de um site, as folhas de estilo em cascata (Cascading Style Sheets) fazem com que o trabalho seja simples. Essa tal folha de estilo é um mecanismo que adiciona estilos a elementos do HTML, como por exemplo cor de fundo, espaçamentos, fontes e etc. Sendo assim, o CSS nos dá a possibilidade de ir além das barreiras de estilo impostas pelo HTML puro.

Vamos criar nosso arquivo CSS para adicionar estilos nas nossas paginas HTML, sendo assim, dentro de seu editor de texto crie um novo arquivo CSS



Salve dentro da pasta que deixamos reservada para arquivos CSS com o nome “styles”

Nome	Data de modificaç...	Tipo	Tamanho
 styles	23/03/2019 17:15	Documento de fol...	0 KB

37 - Chamando o CSS3

Para que seja possível utilizar o CSS do arquivo que criamos na nossa pagina index, antes de tudo é necessário dizer para o HTML onde se encontra o arquivo de folhas de estilo, então é necessário criar um link novo. Existem dois tipos de link, aqueles que levam você para outros documentos e aqueles que são links para fontes externas, sendo assim criaremos um link relativo para importar nosso arquivo CSS:

```
1 <!DOCTYPE html>
2 <html lang="pt-br">
3   <head>
4     <title>Nonsense</title>
5     <meta charset="utf-8">
6     <link rel="stylesheet" href="CSS/styles.css">
7   </head>
```

Note que criamos um link dentro do HEAD, ou seja, ele é um elemento que não faz parte do corpo da página, além disso, colocamos qual a relação do link, no caso é uma folha de estilos então é um stylesheet.

Passo 38 - Sintaxe do css

Agora já podemos editar nosso arquivo CSS, para isso vamos entender como funciona a sintaxe do CSS

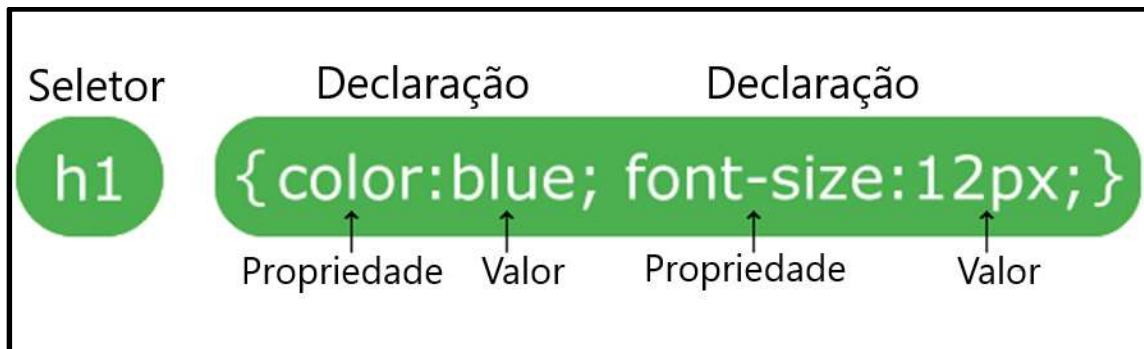


imagem 3 - Sintaxe do CSS (Adaptado de (w3schools, CSS Syntax, 2019))

O seletor no caso pode ser uma tag ou uma classe (falaremos de classes mais adiante), onde suas propriedades devem ser declaradas e em seguida os valores, como o exemplo da imagem 3, que está modificando a tag <h1> para a cor azul e com um tamanho de 12 pixels, o código também pode ser estruturado da seguinte forma:

```
1 h1 {  
2     color: blue;  
3     font-size: 12px;  
4 }
```

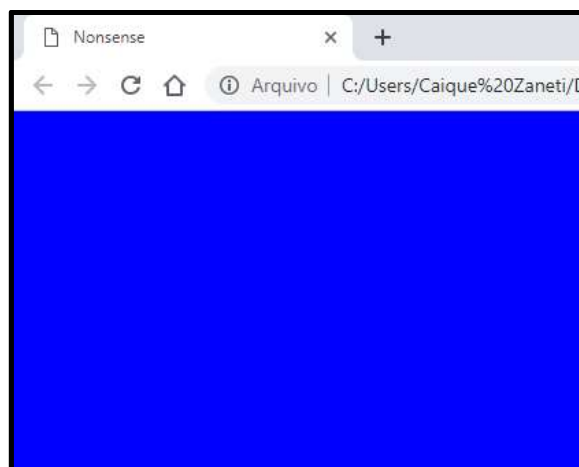
Este código não faz parte do nosso projeto, é apenas uma amostra da estrutura do CSS

Passo 39 – Estilizando o <body> e propriedades de background

Nosso primeiro passo com o CSS será estilizar a tag <body> com várias propriedades de background, vamos primeiramente alterar a cor do nosso background da página index.

```
1 body {  
2     background-color: blue;  
3 }
```

Note que mesmo sem nenhum código extra na nossa pagina HTML o fundo da pagina ficou todo azul



Podemos definir a cor de varias formas, tanto pelo nome, quanto pelo código RGB ou Hexa

```
1 body {  
2     background-color: rgb(254,67,20);  
3 }
```

```
1 body {  
2     background-color: #3cb371;  
3 }
```

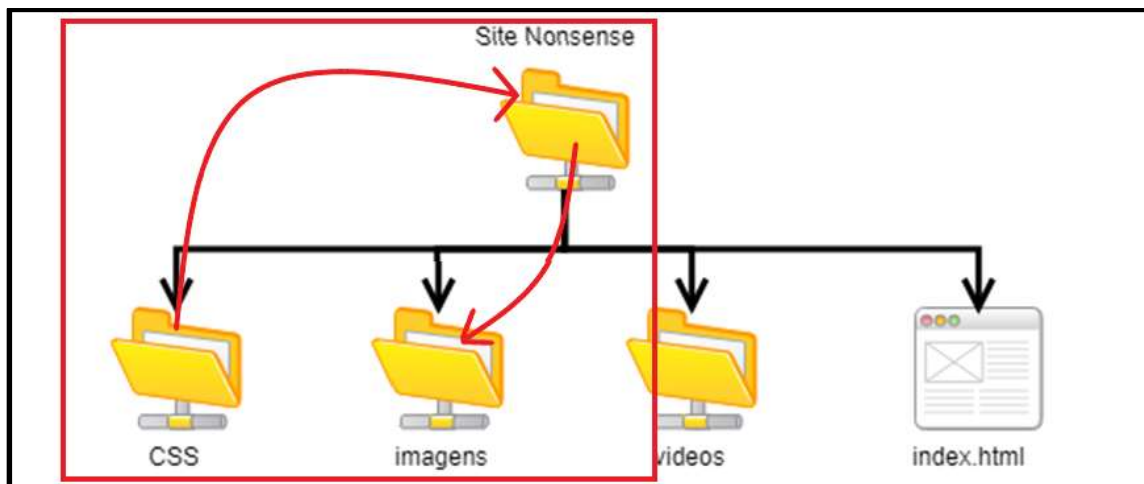
Agora, se estava legal colocar cor no fundo de nossa página, vai ficar muito mais legal no presente momento pois iremos colocar imagens com o elemento background-image

```
1 body {  
2     background-image: url("../imagens/bg2.jpg");  
3 }
```

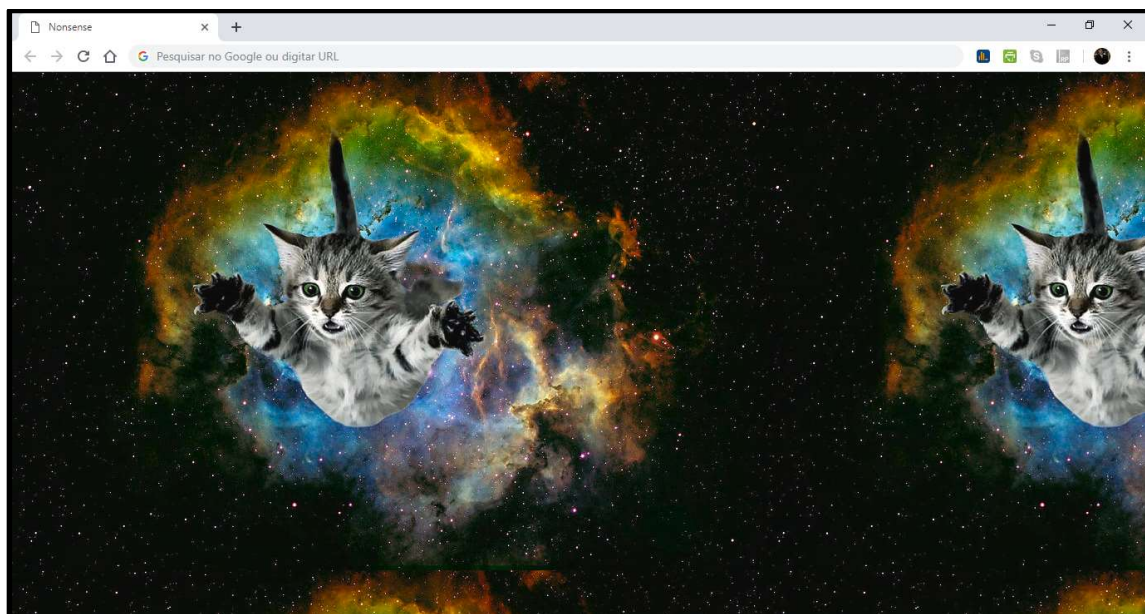
Veja também que antes do caminho para nossa imagem nós adicionamos ../, isso significa que voltamos uma pasta antes de adicionar o caminho original.

E você ainda deve estar se perguntando o motivo de voltarmos uma pasta, então vamos lá, lembra que nós criamos uma pasta chamada CSS e colocamos nosso arquivo CSS lá dentro? Então... nós precisamos chegar na pasta imagens que é onde se encontra a imagem que

baixamos obviamente, então, sem voltar um diretório é impossível acessar a pasta que queremos, lembre nosso fluxo de pastas:



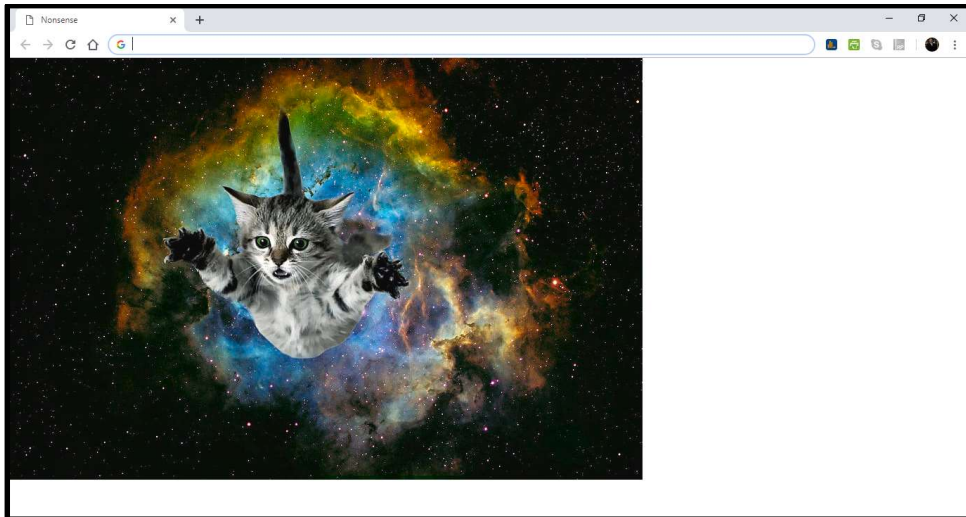
Note que nosso background foi alterado para nossa imagem de fundo



Note também que a imagem por padrão está se repetindo, caso não seja essa a intenção existe a propriedade `background-repeat`

```
1 body {  
2     background-image: url("../imagens/bg2.jpg");  
3     background-repeat: no-repeat;  
4 }
```

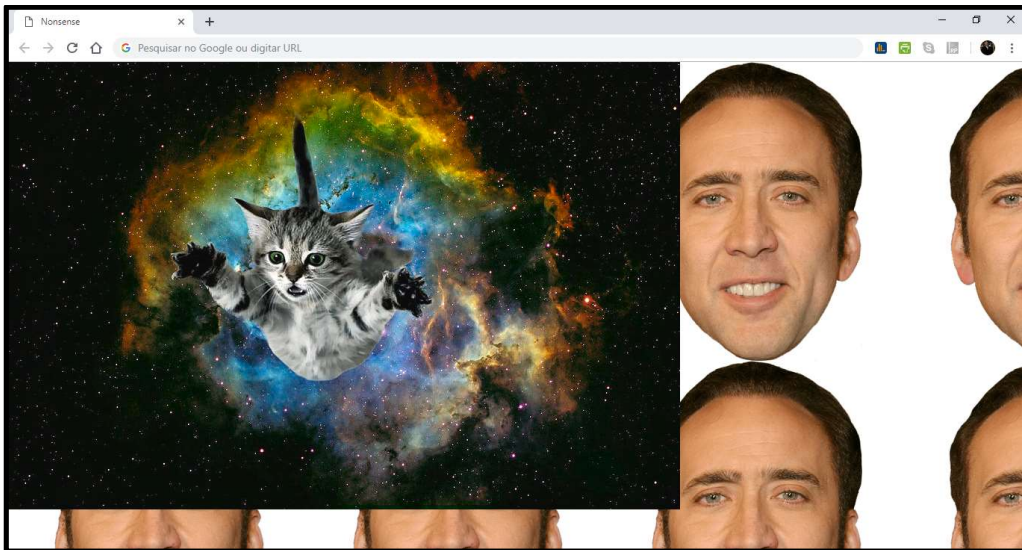
O resultado será uma imagem única no tamanho original do arquivo como fundo do site



Podemos também trabalhar com duas imagens de fundo ao mesmo tempo, utilizando também a propriedade `background-repeat`.

```
1 body {  
2     background-image: url("../imagens/bg2.jpg"), url("../imagens/cage.png");  
3     background-repeat: no-repeat, repeat;  
4 }
```

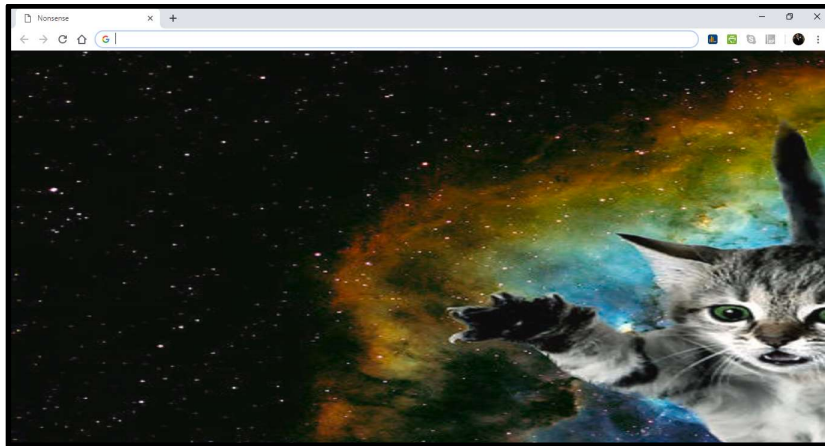
Note que a primeira imagem não se repete, mas a segunda sim.



Podemos também escolher o tamanho da imagem, vamos continuar utilizando o `no-repeat` para facilitar o entendimento

```
1 body {  
2     background-image: url("../imagens/bg2.jpg");  
3     background-repeat: no-repeat;  
4     background-size: 3000px 1000px;  
5 }
```

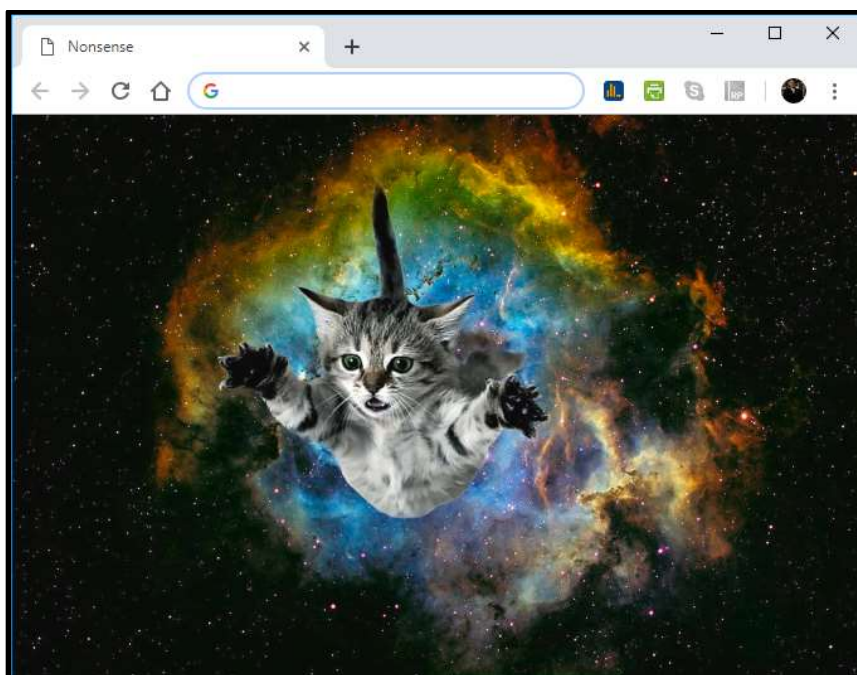
Como resultado teremos uma imagem gigante e estourada pois não fazemos idéia de quantos pixels tem o monitor da pessoa que estará acessando nosso site... isso sem contar a perda de qualidade na imagem.



Para conseguirmos colocar a imagem no plano de fundo inteiro e distorcer-la o menos possível, podemos utilizar no lugar de pixels valores em porcentagem, e agora ativando o repeat novamente.

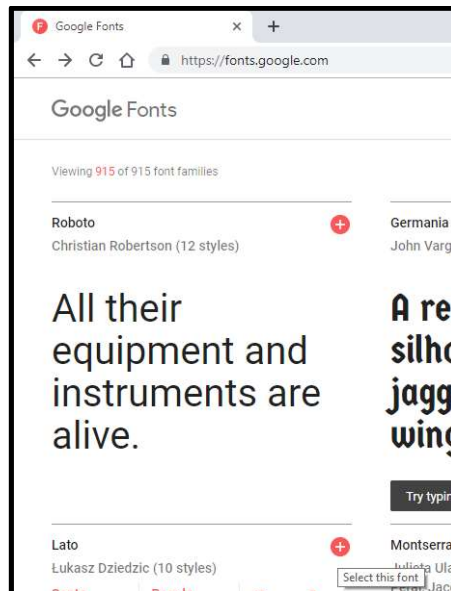
```
1 body {  
2     background-image: url("../imagens/bg2.jpg");  
3     background-repeat: repeat;  
4     background-size: 100%;  
5 }
```

Note que no caso, a imagem ocupará 100% da página, não importando o tamanho do browser, vamos adotar essa configuração para continuar nosso projeto.

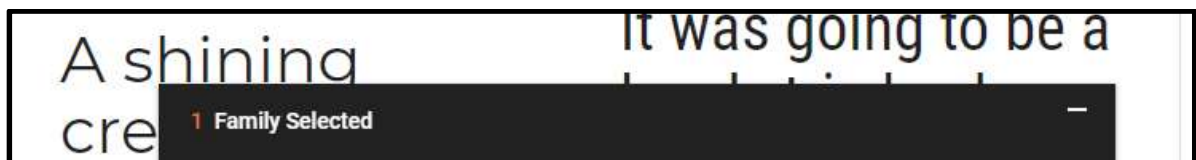


Passo 40 – utilizando fontes de terceiros

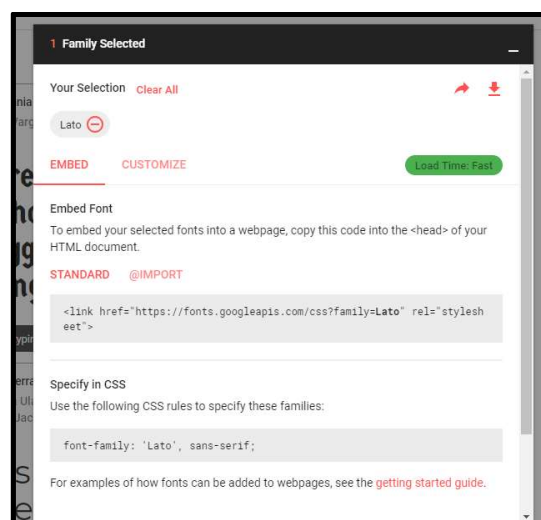
Nosso site precisa ser épico, então precisamos escolher tudo do bom e do melhor para ele, sendo assim, fontes padrões não nos satisfazem. Precisamos de fontes novas e mais legais, para isso entraremos no site <https://fonts.google.com/> e procuraremos alguma fonte bacana.



Ao escolhermos nossa fonte nós clicaremos no símbolo de + que aparece logo ao lado do nome dela, logo em seguida subirá uma barrinha avisando que selecionamos uma fonte



Ao clicarmos nela, subira todas as especificações de como devemos referencia-la no nosso código



Assim sendo, nós adicionaremos o link a nossa pagina HTML index dentro do head

```

3 <head>
4   <title>Nonsense</title>
5   <meta charset="utf-8">
6   <link rel="stylesheet" href="CSS/styles.css">
7   <link href="https://fonts.googleapis.com/css?family=Lato" rel="stylesheet">
8 </head>

```

Dessa forma teremos acesso a essa fonte durante a confecção de nossa página.

Passo 41 – inserindo e estilizando o <main> e aprendendo sobre o canal alpha

Vamos definir agora como será disposto nosso conteúdo principal, vamos então colocar o <main> dentro do nosso <body> para que seja possível o navegador entender que será nosso conteúdo principal e além disso, que terá um estilo próprio.

```

9 <body>
10   <main>
11     ...
12   </main>
13 </body>
14 </html>

```

Apenas o ato de abrir e fechar a tag já é o suficiente para que possamos trabalhar o CSS dela, então vamos lá. No nosso arquivo CSS digite o seguinte código:

```

7 main {
8   background-color: rgba(0, 0, 0, 0.6);
9   padding: 5px 40px;
10  margin-top: 20px;
11  border-radius: 10px;
12 }

```

Vamos agora entender cada coisinha que será feita nesta estilização, vamos lá... Como seletor temos a tag main, ou seja, tudo o que estiver dentro da tag main receberá estes estilos que são:

```

8 background-color: rgba(0, 0, 0, 0.6);

```

Nós já aprendemos como escrever cores em RGB, mas desta vez colocamos RGBA, isso significa que o canal alpha está presente. Daí você me pergunta o que é canal alpha e a resposta é simples, canal alpha é aonde se encontra a **transparência**.

Ou seja, quando colocamos ali a cor RGB junto com o canal alpha estamos definindo o grau de transparência que terá o objeto. O valor dessa transparência é medido na variação entre 0 e 1, sendo 0 totalmente visível e 1 totalmente invisível.

9 `padding: 5px 40px;`

O padding está aí para nos ajudar com o espaçamento entre o conteúdo e as bordas, diferentemente da margem que dá um espaçamento externo, o padding dá um espaçamento interno.

10 `margin-top: 20px;`

O margin-top cria uma margem com o topo do site, nós colocamos em pixels, mas também pode ser colocado em porcentagem sem problemas.

11 `border-radius: 10px;`

E por fim temos o border-radius que define o arredondamento das esquinas do main utilizando como referência o raio das bordas, veja a imagem a seguir:



imagem 4 - Funcionamento do border-radius (Fonte: (mdnwebdocs, 2019))

ao olharmos no navegador poderemos notar que logo em cima da pagina apareceu uma linha transparente, agora nós vamos colocar mais conteúdo dentro dessa linha.

Passo 42 - trabalhando com containers DIV e classificação de tag

Vamos agora dividir nosso conteúdo em pequenos containers, que como o nome já sugere, são espaços separados onde o conteúdo ficará guardado, sendo assim, vamos digitar o seguinte código na nossa página index:

```
10 <main>
11   <div class="logo">
12     
13   </div>
14 </main>
```

Essa div que colocamos foi classificada como “logo”, o nome logo foi dado por nós, se colocássemos pintinho amarelinho iria funcionar da mesma forma. Dentro dessa div colocaremos a imagem referente ao logo da nossa pagina e então editaremos seu estilo no CSS

```

14  .logo {
15      margin: 0;
16      text-align: center;
17  }

```

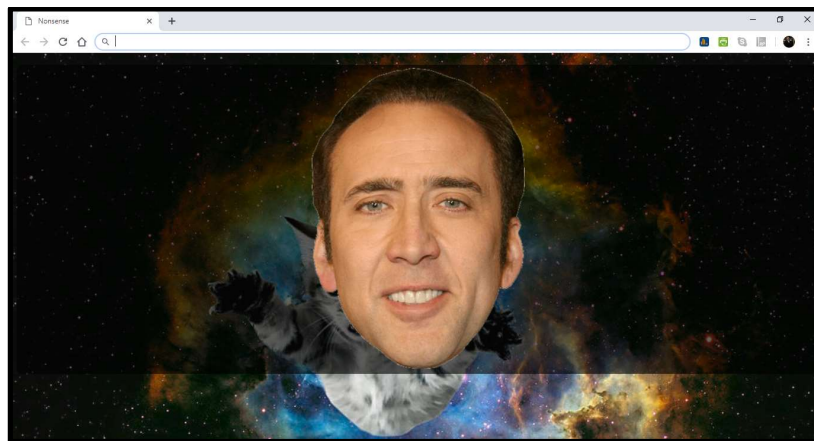
Note que dessa vez ao invés de usar como seletor a tag que vamos modificar, nós utilizamos a classe que criamos, pois na mesma página teremos varias div que podem ou não ser diferentes, sendo assim, o mais prudente é colocar a classe como seletor.

```

15      margin: 0;
16      text-align: center;

```

O margin define o espaçamento entre os elementos, enquanto o text-align faz o alinhamento do texto que no caso queremos que fique no centro, mesmo que nós tenhamos colocado uma imagem e não um texto, o alinhamento será feito da mesma maneira.



Passo 43 – estilizando tags de titulo

Vamos agora voltar a trabalhar com títulos, mudando suas propriedades com classes que nós já aprendemos a fazer, insira o seguinte código na pagina index:

```

10  <main>
11      <div class="logo">
12          
13      </div>
14      <h1 class="manchete">O sentido não esta presente!!!</h1>
15      <h2 class="headline">Jacaré no seco anda??</h2>
16  </main>

```

Podemos colocar essas classificações com as mesmas características, então nós só iremos adicionar mais classes na mesma declaração no css.

```

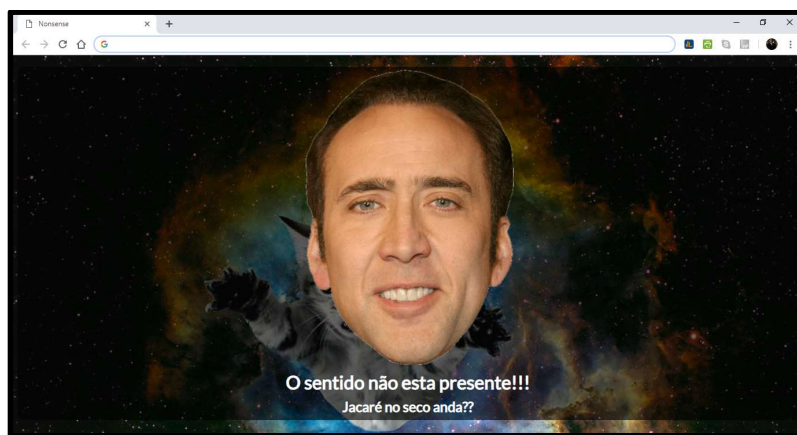
14  .logo, .manchete, .headline {
15      margin: 0;
16      text-align: center;
17  }

```

Mas note que no navegador é possível perceber que a fonte na cor preta fica praticamente ilegível pois nosso fundo já tem tons escuros, sendo assim, precisamos mudar essa fonte. Como o problema é generalizado para todas as classes, o mais sensato é colocarmos a estilização dentro do body para que todas as alterações abaixo recebam os estilos herdados dela.

```
1 body {
2   background-image: url("../imagens/bg2.jpg");
3   background-repeat: repeat;
4   background-size: 100%;
5   font-family: 'Lato';
6   line-height: 1.5;
7   color: white;
8 }
```

Note que estamos usando agora a fonte que baixamos com o elemento font-family. Também utilizamos o line-height para definir a altura da linha e por fim o color para pintarmos nossa fonte de branco.



Passo 44 – inserindo vídeo na nossa pagina

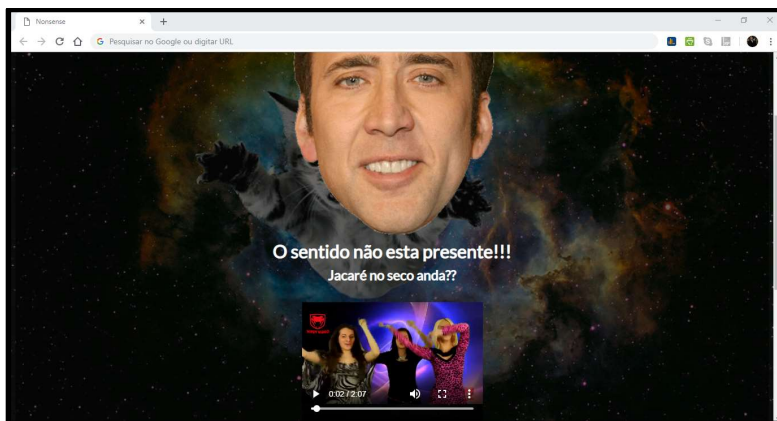
Chegou o momento que muitos estavam esperando! Vamos colocar vídeos na nossa pagina html, para isso vamos inserir o seguinte código na página index:

```
16 <div class="video">
17   <video width="320" height="240" controls="controls">
18     <source src="videos/movie.mp4" type="video/mp4" />
19   </video>
20 </div>
```

Assim como a tag img busca uma imagem, a tag source busca um recurso que no caso é um vídeo. Declaramos no CSS separadamente das outras de preferência pois pode ser que seja necessário aplicar algumas mudanças com o tempo, então vamos deixa-la com as mesmas características do logo mas em uma declaração diferente.

```
22 .video {
23   margin: 0;
24   text-align: center;
25 }
```


O resultado disso é nosso site agora com o recurso de videoo



Passo 45 – Estilizando a tag <article>, fazendo um primeiro artigo, e utilizando a tag <hr>

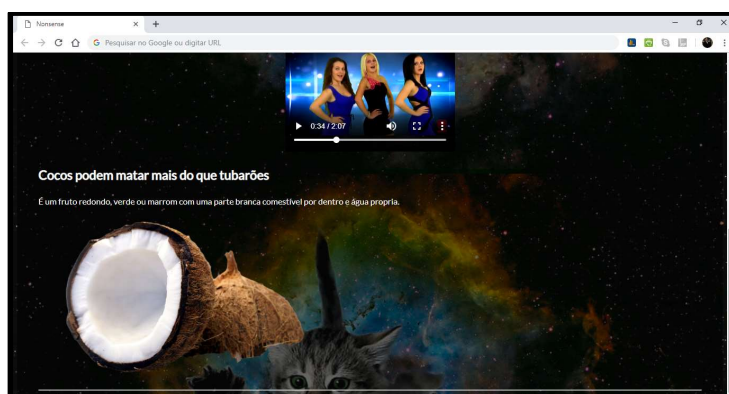
Agora vamos fazer o nosso primeiro artigo do site, contendo parágrafos e imagem, para isso insira o seguinte código na página index:

```

21 <article>
22   <h1>Cocos podem matar mais do que tubarões</h1>
23   <p>É um fruto redondo, verde ou marrom com uma parte branca
24   comestível por dentro e água propria.</p>
25   <figure>
26     
27   </figure>
28 </article>
29 <hr>

```

Mesmo sem colocar nenhum estilo na tag article, podemos ver que ela já esta herdando estilos do body, utilizamos também a tag <hr> que faz uma linha entre um conteúdo e outro.



Para mudar isso precisamos sobrepor os estilos

```

27 article {
28     text-align: center;
29     font-style: normal;
30     font-size: 20px;
31     color: rgb(0,255,0);
32 }

```

Nós alinhamos o texto no centro, mudamos o estilo da fonte, deixamos o tamanho da fonte como 20 pixels além de deixarmos a fonte na cor verde.

Passo 46 – Trabalhando com estilização de tabelas

Nós já aprendemos o básico de tabelas, entretanto, vamos aprender a estiliza-las com o CSS e também utilizar novas tags de tabela, insira o seguinte código na página index:

```

30 <div class="datas">
31     <h2>Datas Comemorativas Aleatorias</h2>
32     <table>
33         <thead>
34             <tr>
35                 <th>Acontecimento</th>
36                 <th>Data</th>
37             </tr>
38         </thead>
39         <tbody>
40             <tr>
41                 <td>Dia Mundial do Animal de Laboratório</td>
42                 <td>24/04</td>
43             </tr>
44             <tr>
45                 <td>Dia da Levitação</td>
46                 <td>16/12</td>
47             </tr>
48             <tr>
49                 <td>Dia da Anestesia Geral</td>
50                 <td>30/03</td>
51             </tr>
52             <tr>
53                 <td>Dia da Baiana do Acarajé</td>
54                 <td>25/11</td>
55             </tr>
56         </tbody>
57     </table>
58 </div>

```

As tags novas que vimos nesse código são <thead> e <tbody> que fazem funcionar para separar o que é o cabeçalho da tabela e o que é o corpo da tabela, uma didática fundamental para o navegador.

Vamos agora estilizar separadamente a div e a table que se encontra dentro da div, insira o seguinte código no seu arquivo CSS:

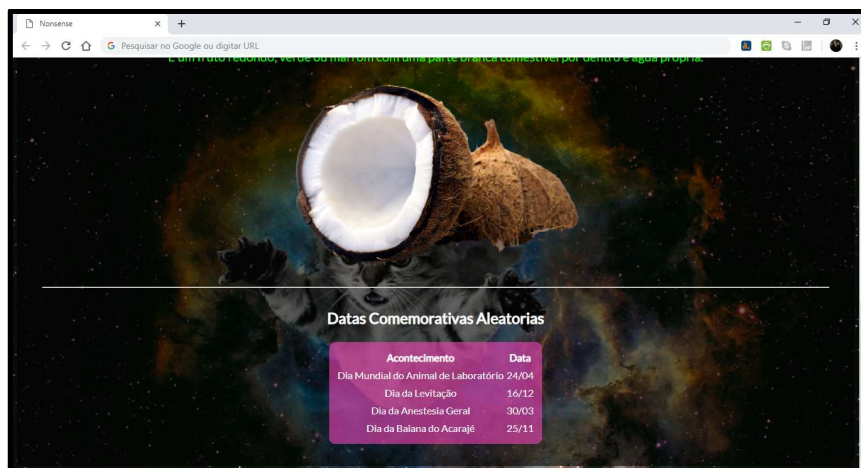
```

34  .datas {
35      margin: 30px 0;
36      text-align: center;
37      width: 100%;
38  }
39  .datas table {
40      background: rgba(244, 66, 182, 0.6);
41      border-radius: 10px;
42      padding: 10px;
43      margin: 0 auto;
44  }

```

O primeiro trecho do CSS diz respeito a div, que terá margem de 30 pixels, texto alinhado no centro e largura de 100%

Já a segunda parte do CSS diz respeito a tabela que se encontra dentro da div, que terá um background mais em um tom de rosa e também terá um nível de transparência, por isso utilizamos o RGBA, de resto a configuração é parecida com a da transparência que fizemos no body, utilizando padding e border-radius.



Exercício 5 – Hackeando o Google

Utilize CSS para Hackear o site do google e alterar todo o seu layout

Exercício 6 – Meu Site Plus Advanced

Agora que você aprendeu a dominar o CSS, crie um novo projeto do zero ou continue o projeto que você estava fazendo e coloque todos os elementos do capítulo 4 em prática.