

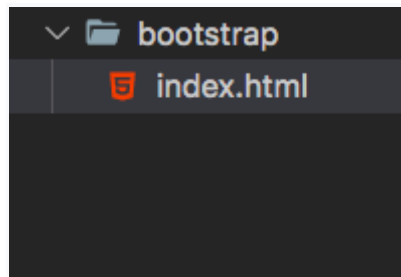
Bootstrap

Conceito

Kit de ferramentas de front-end (HTML, CSS e JS) poderoso, extensível e repleto de recursos. Crie e personalize com Sass, utilize sistemas e componentes de grade pré-construídos e dê vida a projetos com poderosos plugins JavaScript.

Instalação

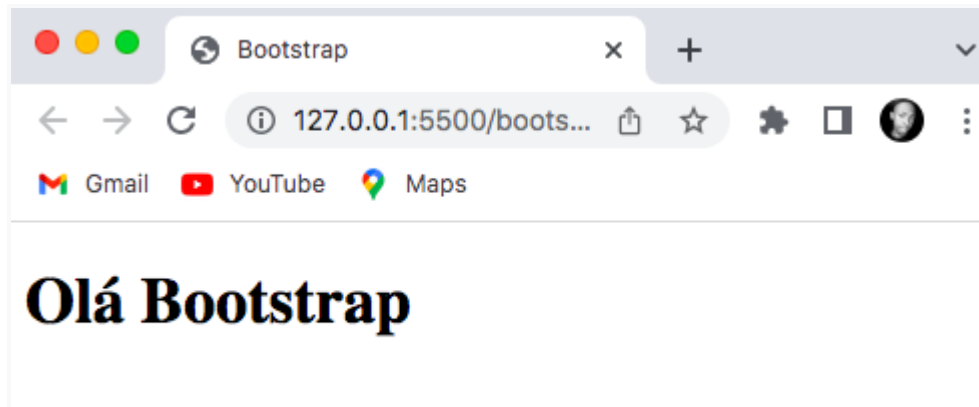
Antes de partirmos para a instalação vamos primeiro criar uma pasta chamada bootstrap e dentro desta pasta vamos criar o arquivo index.html.



Dentro de index.html vamos criar a estrutura básica HTML. No documento criado na imagem a baixo temos um documento simples onde temos no **title** “Bootstrap” e um **h1** “Olá Bootstrap”.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Bootstrap</title>
8  </head>
9  <body>
10     <h1>Olá Bootstrap</h1>
11 </body>
12 </html>
```

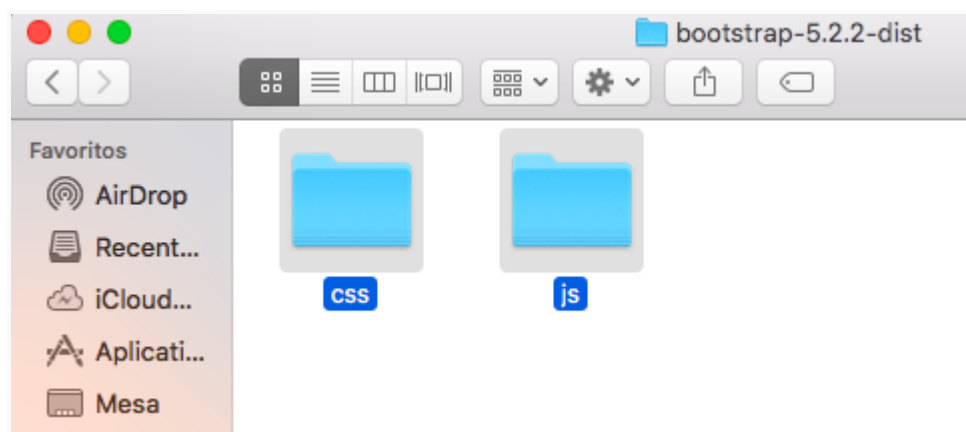
Resultado:



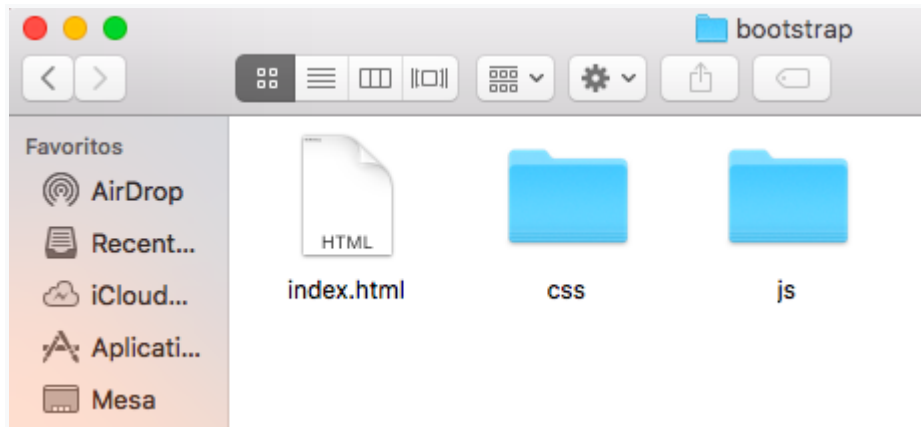
Até ao momento nada de novo aqui. Vamos começar por visitar o site oficial do Bootstrap <https://getbootstrap.com/> e clicamos no link de download e seremos direcionados para página de download e na página de download baixamos o **Compiled CSS and JS** (Para as outras opções siga a documentação <https://getbootstrap.com/docs/5.2/getting-started/download/>).

A seguir, dentro da pasta **bootstrap** que acabamos de baixar, copiamos as pastas css e js e colamos no nas pasta bootstrap que criamos as pouquinho como mostram os passos a seguir.

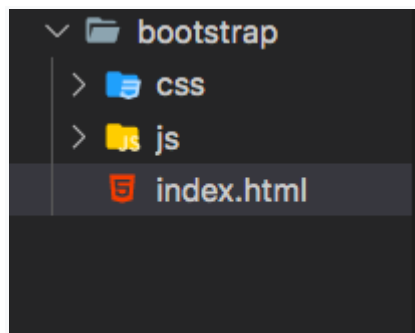
Passo: 1



Passo: 2



No VS Code:

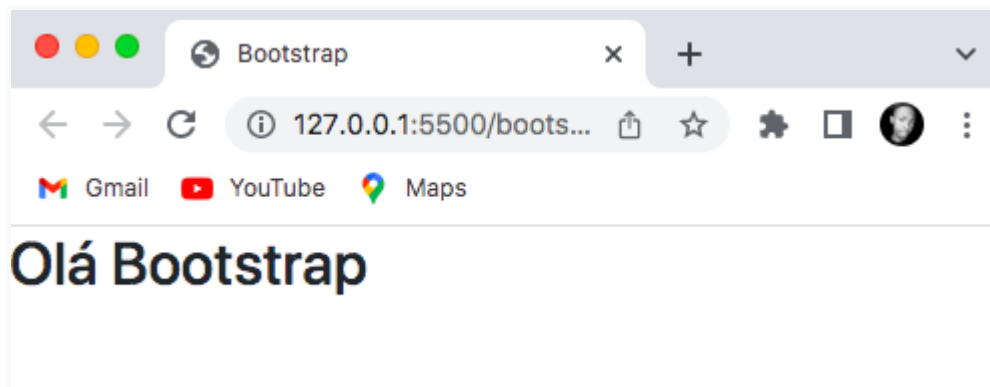


Dentro da pasta CSS encontramos os arquivos do bootstrap responsáveis por conterem as classes que vamos usar na nossa página. E na pasta js nós encontramos os arquivos JavaScript responsáveis pelas animações do bootstrap como carrousel, dropdown, entre outros.

Vamos começar então a ver como importa o bootstrap na página HTML. Em index.html importamos usando a tag link vamos importar bootstrap.css como mostra a imagem.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet" href="./css/bootstrap.css">
8   <title>Bootstrap</title>
9 </head>
10 <body>
11   <h1>Olá Bootstrap</h1>
12 </body>
13 </html>
```

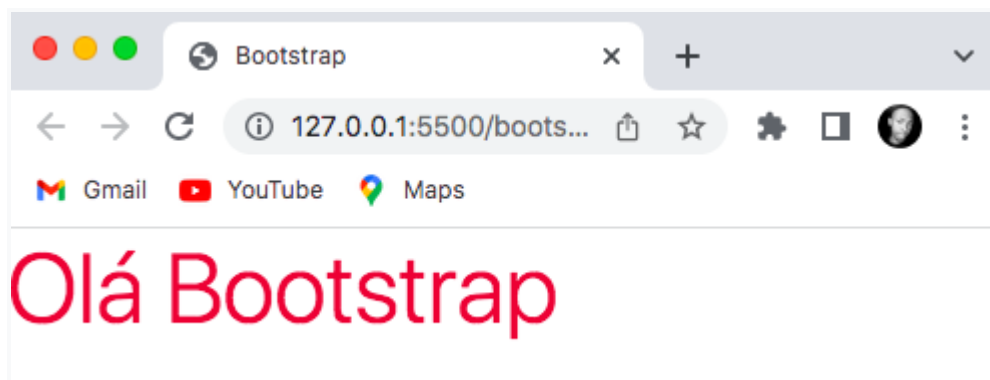
Com isso se dermos uma olhada no navegador a fonte muda para a fonte padrão do bootstrap, o que significa que temos o bootstrap instalado em nossa página:



Podemos adicionar uma classes aos nossos elementos para ver ver a funcionar

```
<h1 class="display-1 text-danger">Olá Bootstrap</h1>
```

Resultado:



Ao adicionar a class **display-1** aumentamos o tamanho do e a class **text-danger** torna o texto vermelho.

Também podemos incluir o javascript:

```
10 <body>
11   <h1 class="display-1 text-danger">Olá Bootstrap</h1>
12
13   <script src="./js/bootstrap.js"></script>
14 </body>
```

Agora que temos o bootstrap configuramos vamos explorar.

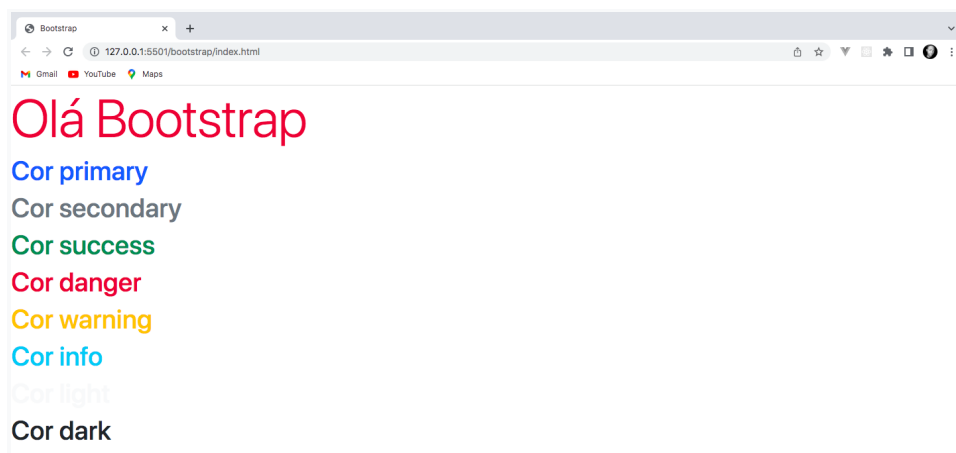
Cores do Bootstrap

O bootstrap suporta um extenso sistema de cores que permite estilizar os componentes. Isto habilita uma customização mais compreensiva e extensiva do para qualquer tipo de projecto. As cores de tema do Bootstrap são: **primary**, **secondary**, **success**, **danger**, **warning**, **info**, **light**, **dark**.

Vamos aplicar as diferentes cores na página que estamos a trabalhar, a princípio nós já aplicamos uma cor no texto que é a cor vermelha utilizando a class **.text-danger**, vamos agora aplicar outras cores para ver como fica.

```
bootstrap > index.html > html > body
 8  <title>Bootstrap</title>
 9  </head>
10  <body>
11    <h1 class="display-1 text-danger">Olá Bootstrap</h1>
12
13    <h1 class="text-primary">Cor primary</h1>
14    <h1 class="text-secondary">Cor secondary</h1>
15    <h1 class="text-success">Cor success</h1>
16    <h1 class="text-danger">Cor danger</h1>
17    <h1 class="text-warning">Cor warning</h1>
18    <h1 class="text-info">Cor info</h1>
19    <h1 class="text-light">Cor light</h1>
20    <h1 class="text-dark">Cor dark</h1>
21
22    <script src="./js/bootstrap.js"></script>
23  </body>
24  </html>
25
```

Como vemos na imagem acima nós adicionamos h1 com a class **text-nome_da_cor** fazendo isso nós adicionamos cor aos texto conforme mostra o resultado no navegador.



Assim como aplicamos cor no texto as cores podem ser aplicadas em qualquer componente do bootstrap como vamos ter a oportunidade de ver.

Buttons

O Bootstrap inclui vários estilos predefinidos de botões, cada um deles servindo o seu propósito semântica, com alguns extras adicionados para maior controle. Na nossa página de depois dos textos vamos adicionar 1 **<button>** como vemos na imagem.

```
bootstrap > index.html > html > body
15 <h1 class="text-success">Cor success</h1>
16 <h1 class="text-danger">Cor danger</h1>
17 <h1 class="text-warning">Cor warning</h1>
18 <h1 class="text-info">Cor info</h1>
19 <h1 class="text-light">Cor light</h1>
20 <h1 class="text-dark">Cor dark</h1>
21
22 <!-- Buttons -->
23 <button class="btn btn-primary">Primary</button>
24
25 <script src="./js/bootstrap.js"></script>
26 </body>
27 </html>
28
```

Este **<button>** recebe a class **btn** que define a estrutura de um botão em bootstrap e depois recebe a class **btn-primary** que define a cor do botão. Como resultado vemos a seguir.

Cor danger

Cor warning

Cor info

Cor light

Cor dark

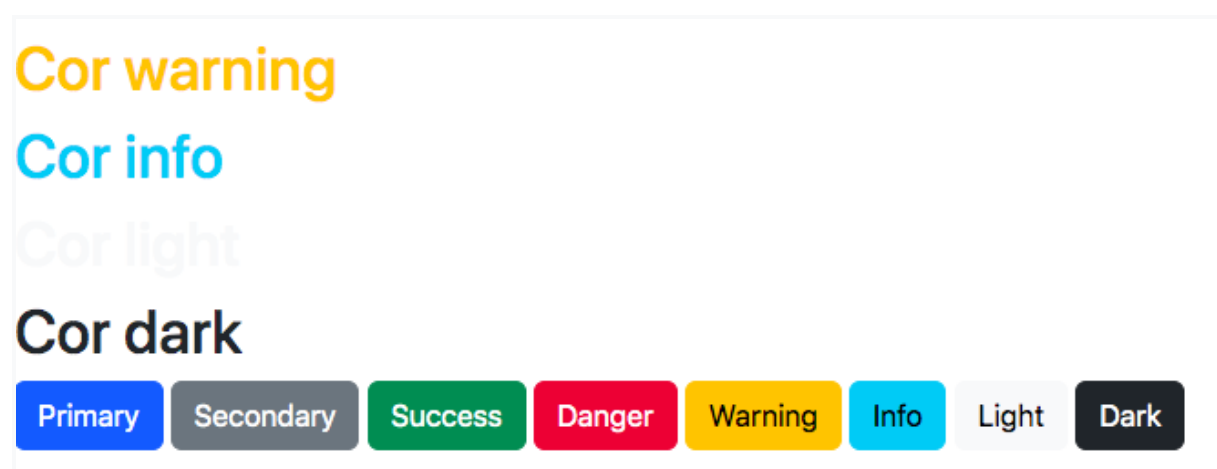
Primary

Agora podemos adicionar outros buttons à nossa página usando as cores que já aprendemos.

```
bootstrap > index.html > html > body
15 <h1 class="text-success">Cor success</h1>
16 <h1 class="text-danger">Cor danger</h1>
17 <h1 class="text-warning">Cor warning</h1>
18 <h1 class="text-info">Cor info</h1>
19 <h1 class="text-light">Cor light</h1>
20 <h1 class="text-dark">Cor dark</h1>
21
22 <!-- Buttons -->
23 <button class="btn btn-primary">Primary</button>
24
25 <button class="btn btn-secondary">Secondary</button>
26 <button class="btn btn-success">Success</button>
27 <button class="btn btn-danger">Danger</button>
28 <button class="btn btn-warning">Warning</button>
29 <button class="btn btn-info">Info</button>
30 <button class="btn btn-light">Light</button>
31 <button class="btn btn-dark">Dark</button>
32
33 <script src="./js/bootstrap.js"></script>
34 </body>
35 </html>
```

Na imagem como podemos ver, adicionamos outros buttons contendo as classes das diferentes cores como **btn-secondary**, **btn-danger**, **btn-info**....

Resultado:



Alerts

Os são componentes que fornecem mensagens de feedback contextuais para ações típicas do usuário, como no caso de um usuário que submete informação por meio de um formulário.

```
bootstrap > index.html > ...
30 <button class="btn btn-light">Light</button>
31 <button class="btn btn-dark">Dark</button>
32
33 <!-- Alerts -->
34 <div class="alert alert-success mt-5" role="alert">
35 |   Alert success. Ex: Cadastro realizado com sucesso!
36 </div>
37 <div class="alert alert-danger" role="alert">
38 |   Alert danger. Ex: Falha ao castrar preencha todos os campos!
39 </div>
40
41 <script src="./js/bootstrap.js"></script>
42 </body>
43 </html>
44
```

No código acima primeiro adicionamos uma `<div>` com as class **alert alert-success** para indicar que uma operação do usuário foi bem sucedida, e outra div com as class **alert alert-danger** para indicar que houve um erro na operação. E note que usamos a class **mt-5 (margin-top)** na primeira div, esta é uma class do bootstrap para margin-top que vai de 0 a 5 (0 sem margin e 5 valor máximo). Assim como para também temos **mb (margin-bottom)**, **pt, pb (padding-top, paddin-bottm)**, para saber mais visite <https://getbootstrap.com/docs/5.2/layout/utilities/>. Com isso temos no navegador:

Alert success. Ex: Cadastro realizado com sucesso!

Alert danger. Ex: Falha ao castrar preencha todos os campos!

Sistema de Grid

O sistema de grid (grade) usa flexbox para criar layouts de todas as formas e tamanhos graças a um sistema de doze colunas, seis camadas responsivas padrão, variáveis e dezenas de classes predefinidas.

Como Funciona

Para criar o sistema de Grid nós primeiro criamos um elemento com a class **.container** que vai ser o pai, depois criamos os filhos que são **.row** (linha) e dentro delas colocamos as **.col** (colunas).

Exemplo:

```
bootstrap > index.html > html > body > div.alert.alert-danger
38 | Alert danger. Ex: Falha ao castrar preencha todos os campos!
39 | </div>
40 |
41 | <!-- Sistema de Grid -->
42 | <div class="container mb-5 text-center">
43 |   <div class="row">
44 |     <div class="col border">
45 |       Column
46 |     </div>
47 |     <div class="col border">
48 |       Column
49 |     </div>
50 |     <div class="col border">
51 |       Column
52 |     </div>
53 |   </div>
54 | </div>
55 |
56 | <script src="./js/bootstrap.js"></script>
57 | </body>
58 | </html>
59 |
```

O exemplo acima cria três colunas de largura igual em todos os dispositivos e viewports usando classes de grade predefinidas. Essas colunas são centralizadas na página com o **container** pai. Em cada coluna usamos a class **border** só para conseguir visualizar as nossas colunas como vemos abaixo.

Column	Column	Column
--------	--------	--------

O sistema de grade do Bootstrap pode se adaptar em todos os seis breakpoints (pontos de interrupção) e em qualquer que você personalizar. As seis camadas de grade padrão são as seguintes:

- Extra small (xs)
- Small (sm)
- Medium (md)
- Large (lg)
- Extra large (xl)
- Extra extra large (xxl)

	xs <576px	sm ≥576px	md ≥768px	lg ≥992px	xl ≥1200px	xxl ≥1400px
Container max-width	None (auto)	540px	720px	960px	1140px	1320px
Class prefix	<code>.col-</code>	<code>.col-sm-</code>	<code>.col-md-</code>	<code>.col-lg-</code>	<code>.col-xl-</code>	<code>.col-xxl-</code>
# of columns	12					
Gutter width	1.5rem (.75rem on left and right)					
Custom gutters	Yes					
Nestable	Yes					
Column ordering	Yes					

Exemplo com largura de coluna diferente:

O layout automático para colunas de grade flexbox também significa que você pode definir a largura de uma coluna nos diferentes breakpoints e fazer com que as colunas irmãs sejam redimensionadas automaticamente em torno dela . Como se pode ver nas seguintes imagens:

```
strap > index.html > html > body > div.container.text-center.my-5 >
56      <!-- Com largura de coluna diferente -->
57      <div class="container text-center my-5">
58          <div class="row justify-content-md-center border">
59              <div class="col col-lg-2 border">
60                  1 de 3
61              </div>
62              <div class="col-md-auto border">
63                  Largura Variável
64              </div>
65              <div class="col col-lg-2 border">
66                  3 de 3
67              </div>
68          </div>
69          <div class="row border">
70              <div class="col border">
71                  1 de 3
72              </div>
73              <div class="col-md-auto border">
74                  Largura Variável
75              </div>
76              <div class="col col-lg-2 border">
77                  3 de 3
78              </div>
79          </div>
80      </div>
81
82      <script src="./js/bootstrap.js"></script>
83  </body>
84  </html>
```

Largura: **lg**

	1 de 3	Largura Variável	3 de 3	
	1 de 3		Largura Variável	3 de 3

Largura: **md**

1 de 3	Largura Variável	3 de 3
1 de 3	Largura Variável	3 de 3

Largura: **sm**

1 de 3
Largura Variável
3 de 3
1 de 3
Largura Variável
3 de 3

Cards

Um card (cartão) é um contêiner de conteúdo flexível e extensível. Inclui opções para cabeçalhos e rodapés, uma ampla variedade de conteúdo, cores de fundo contextuais e opções de exibição poderosas.

Exemplo:

Para criarmos um card vamos precisar de uma imagem, pegue uma imagem e cole na pasta que estamos a trabalhar.

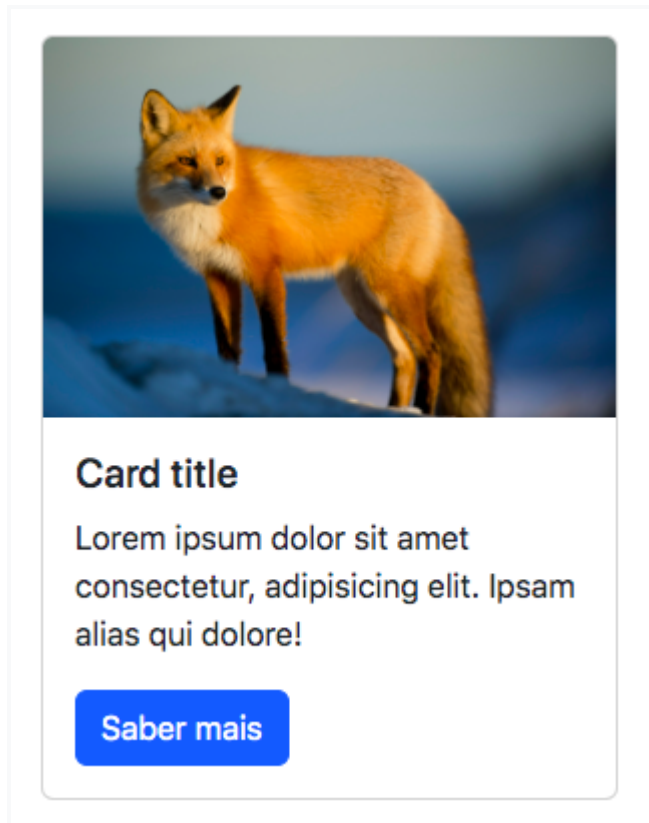
No meu eu caso peguei a imagem de uma raposa do site **Unsplash** <https://unsplash.com/>

```
bootstrap > index.html > ...
81
82 <!-- Cards -->
83 <div class="card" style="width: 18rem;">
84   
85   <div class="card-body">
86     <h5 class="card-title">Card title</h5>
87     <p class="card-text">
88       Lorem ipsum dolor sit amet consectetur, adipisicing elit. Ipsam alias qui dolore!
89     </p>
90     <a href="#" class="btn btn-primary">Saber mais</a>
91   </div>
92 </div>
93
94 <script src="./js/bootstrap.js"></script>
95 </body>
96 </html>
```

No código acima usamos uma `<div>` com a class **card**, a seguir utilizamos o atributo **style** e demos uma largura de **18rem** para o nosso card, como filho do card usamos uma `` com a class **card-img-top** e definimos o corpo do card através da

`<div>` com a class **card-body** onde como filho usamos um `<h1>` com a class o **card-title**, um `<p>` com a class **card-text** e finalmente um button com as classes **btn btn-primary**.

Resultado:



Forms

O bootstrap possui classes que permitem criar e estilizar os elementos do formulário bem como criar estrutura de layouts variados.

Form controls

A class form-control dá aos as tags **input** e **textarea** um estilo customizado, tamanho, estilo de focos, e muito mais.

Exemplo:

```

bootstrap > index.html > ...
93
94 <!-- Forms -->
95 <form class="my-5 p-4">
96   <div class="mb-3">
97     <label for="email" class="form-label">Email</label>
98     <input type="email" class="form-control" id="email" placeholder="Seu endereço de email">
99   </div>
100  <div class="mb-3">
101    <label for="mensagem" class="form-label">Mensagem</label>
102    <textarea class="form-control" id="mensagem" rows="3"></textarea>
103  </div>
104  <button class="btn btn-primary">Enviar</button>
105 </form>
106
107 <script src="./js/bootstrap.js"></script>
108 </body>
109 </html>
110

```

No código acima se criou um **<form>**, dentro criou-se **<div>**'s onde se encontram a label com a class **form-label** e **<input>** (e **<textarea>** no segundo) relacionado a essa label com a class **form-control** que faz com que tenham o estilo de um formulário bootstrap. Ainda dentro deste formulário encontramos um **<button>** com as class **btn btn-primary**.

Resultado:

Email

Seu endereço de email

Mensagem

Enviar

Podemos criar ainda layouts diversos de formulários e adicionar mais campos com podemos ver a seguir:

```

bootstrap > index.html > html > body
94  <!-- Forms -->
95  <form class="my-5 p-4 row g-3">
96    <div class="col-md-6">
97      <label for="nome" class="form-label">Nome</label>
98      <input type="text" class="form-control" id="nome" placeholder="Seu nome">
99    </div>
100   <div class="col-md-6">
101     <label for="sobrenome" class="form-label">Sobre Nome</label>
102     <input type="text" class="form-control" id="sobrenome" placeholder="Seu endereço sobre nome">
103   </div>
104   <div class="col-md-8">
105     <label for="email" class="form-label">Email</label>
106     <input type="email" class="form-control" id="email" placeholder="Seu endereço de email">
107   </div>
108   <div class="col-md-4">
109     <label for="cidade" class="form-label">Cidade</label>
110     <select id="cidade" class="form-select">
111       <option selected>Selecione</option>
112       <option>Luanda</option>
113       <option>Benguela</option>
114       <option>Kwanza Sul</option>
115     </select>
116   </div>
117   <div class="col-12">
118     <label for="mensagem" class="form-label">Mensagem</label>
119     <textarea class="form-control" id="mensagem" rows="3"></textarea>
120   </div>
121   <div class="col-12">
122     <button class="btn btn-primary">Enviar</button>
123   </div>
124 </form>

```

Neste código as diferenças com o anterior é que agora no **<form>** estamos a usar a class **row** com isso podemos alinhar elementos na mesma linha, e a class **g-3** atribui espaço entre os **<input>s** na mesma linha. Também removemos as class **mb-3** nas **<div>s** que seguram os elementos e atribuímos a class **col-md-6** para a **<div>** do nome e do sobrenome o que significa que os elementos ali no tamanho médio vão estar na mesma linha e ocupam a cada um ocupa o mesmo tamanho. Para as **<div>** com a class **12** vão ocupar uma linha. E também usamos um **<select>** com a class **form-select**.

Nome

Seu nome

Sobre Nome

Seu endereço sobre nome

Email

Seu endereço de email

Cidade

Selecione

Mensagem

Enviar

Navbar

É um poderoso e responsivo cabeçalho de navegação. Inclui suporte para branding, navegação e muito mais, incluindo suporte para nosso plug-in de recolhimento.

Exemplo para criar um Navbar vamos criar a partir do modelo apresentado na página do bootstrap <https://getbootstrap.com/docs/5.2/components/navbar/>. Podemos pegar a estrutura que encontramos nessa página e colamos no topo da nossa página e ou seja depois do **<body>** como vemos na imagem.

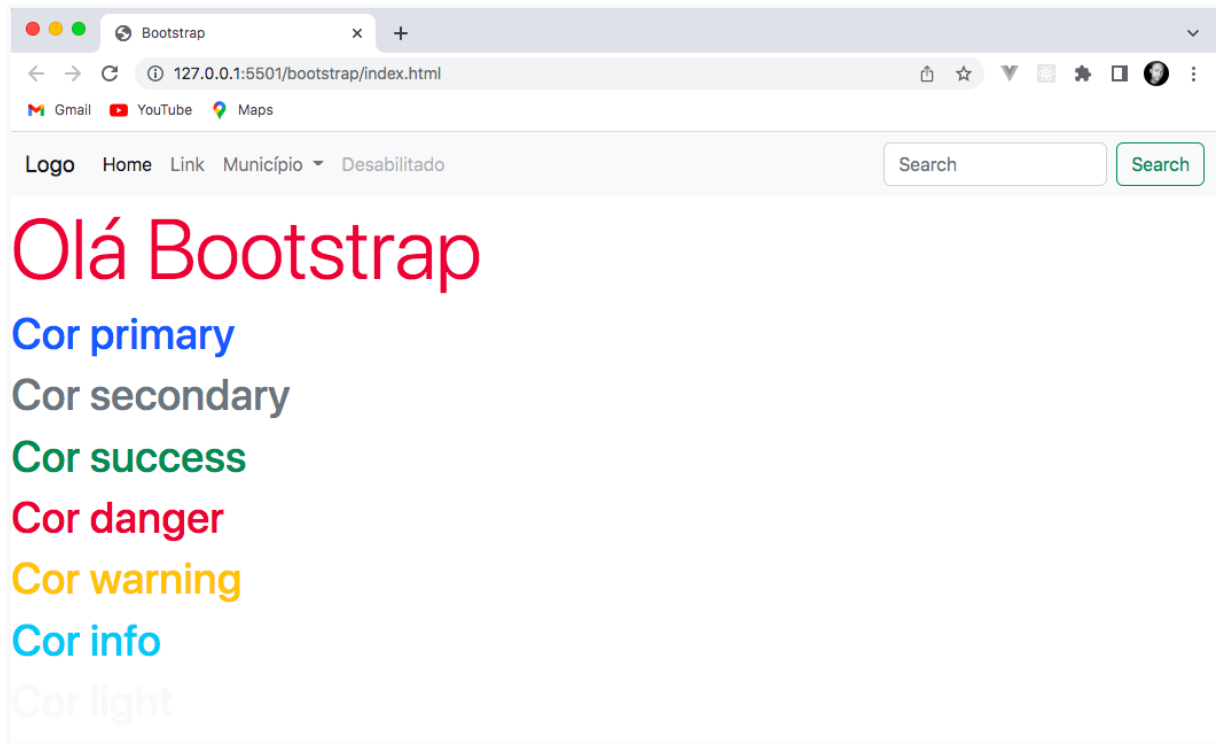

```

10 <body>
11   <nav class="navbar navbar-expand-lg bg-light">
12     <div class="container-fluid">
13       <a class="navbar-brand" href="#">Logo</a>
14       <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navba
15       <span class="navbar-toggler-icon"></span>
16     </button>
17     <div class="collapse navbar-collapse" id="navbarSupportedContent">
18       <ul class="navbar-nav me-auto mb-2 mb-lg-0">
19         <li class="nav-item">
20           <a class="nav-link active" aria-current="page" href="#">Home</a>
21         </li>
22         <li class="nav-item">
23           <a class="nav-link" href="#">Link</a>
24         </li>
25         <li class="nav-item dropdown">
26           <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria
27             Município
28           </a>
29           <ul class="dropdown-menu">
30             <li><a class="dropdown-item" href="#">Cacuaco</a></li>
31             <li><a class="dropdown-item" href="#">Viana</a></li>
32             <li><hr class="dropdown-divider"></li>
33             <li><a class="dropdown-item" href="#">Outro Link</a></li>
34           </ul>
35         </li>
36         <li class="nav-item">
37           <a class="nav-link disabled">Desabilitado</a>
38         </li>
39       </ul>
40       <form class="d-flex" role="search">
41         <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">
42         <button class="btn btn-outline-success" type="submit">Search</button>
43       </form>
44     </div>
45   </div>
46 </nav>

```

Nesta imagem temos uma estrutura básica de uma navbar onde primeiro criada através de uma tag **<nav>** com a class **navbar** que define a estrutura de uma navegação em bootstrap, definimos que temos uma navegação estendida através da class **navbar-expand-lg**, **bg-light** é diz que a cor de fundo da nossa navbar é o cinzento claro. A class **navbar-nav** define a estrutura de uma lista da navegação, **nav-item** define os itens da nossa navegação, **nav-link** define os links de uma navbar, vemos ali um **<button>** com a class **navbar-toggler** para abrir e fechar o os elementos ou os filhos da navegação no tamanho mobile. Também temos um **<button>** com a class **dropdown** este será responsável por mostrar os elementos da **** com a class dropdown-menu onde no nosso caso colocamos os nossos municípios. Depois criamos ali um form contendo o um **<input>** com as class form-control e um **<button>** com a class **btn btn-success-outline** que significa que o nosso button será realçado apenas com uma borda.

Resultado:



Carousel

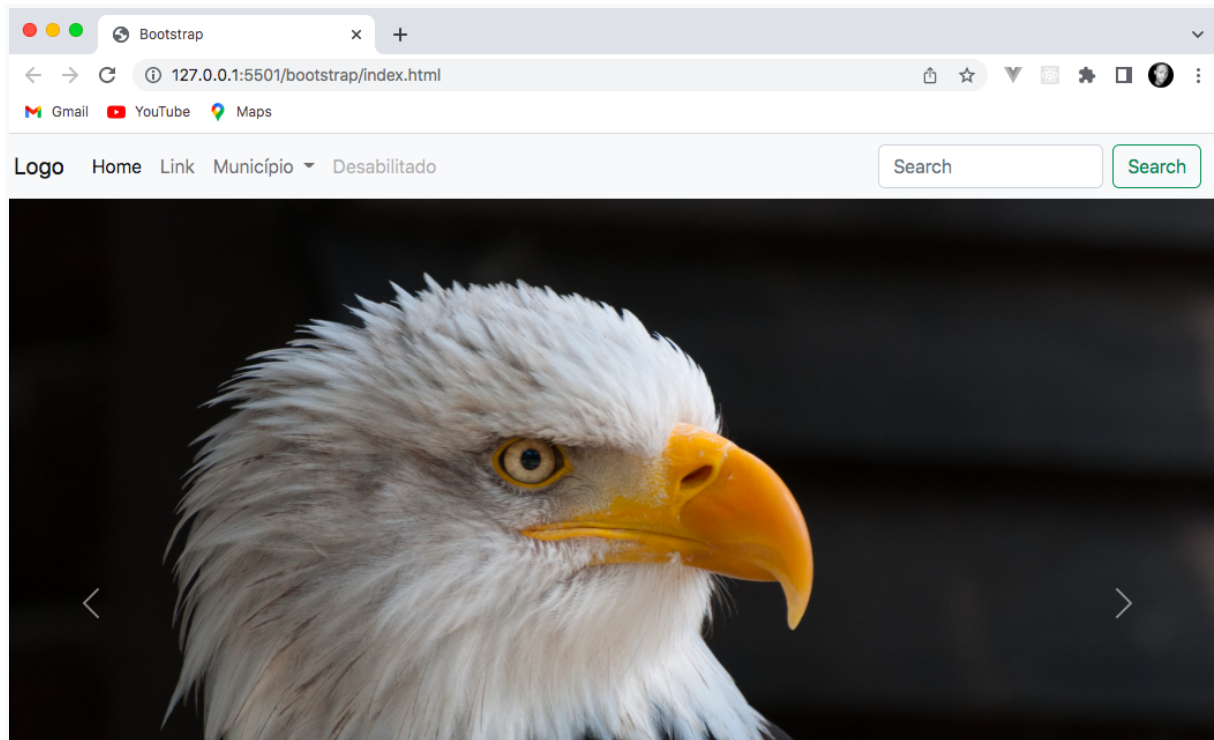
Um componente de apresentação de slides para percorrer elementos - imagens ou slides de texto - como um carrossel.

```
49 <!-- Carousel -->
50 <div id="carouselExampleControls" class="carousel slide" data-bs-ride="carousel">
51   <div class="carousel-inner">
52     <div class="carousel-item active">
53       
54     </div>
55     <div class="carousel-item">
56       
57     </div>
58     <div class="carousel-item">
59       
60     </div>
61   </div>
62   <button class="carousel-control-prev" type="button" data-bs-target="#carouselExampleControls">
63     <span class="carousel-control-prev-icon" aria-hidden="true"></span>
64     <span class="visually-hidden">Previous</span>
65   </button>
66   <button class="carousel-control-next" type="button" data-bs-target="#carouselExampleControls">
67     <span class="carousel-control-next-icon" aria-hidden="true"></span>
68     <span class="visually-hidden">Next</span>
69   </button>
70 </div>
```

No código acima temos nós criamos uma `<div>` com uma class carousel, dentro temos o **inner** onde temos os itens do carousel que são os **carousel-item** dentro

colocamos as nossas imagens onde no exemplo colocamos imagens de animais renomeadas de 1 a 3.

Resultado:



Links Úteis

Colors:

<https://getbootstrap.com/docs/5.2/customize/color/>

Button:

<https://getbootstrap.com/docs/5.2/components/buttons/>

Alerts

<https://getbootstrap.com/docs/5.2/components/alerts/>

Grid

<https://getbootstrap.com/docs/5.2/layout/grid/>

Card

<https://getbootstrap.com/docs/5.2/components/card/>

Forms

<https://getbootstrap.com/docs/5.2/forms/overview/>

Navbar

<https://getbootstrap.com/docs/5.2/components/navbar/>

Carousel

<https://getbootstrap.com/docs/5.2/components/carousel/>