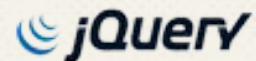




# OFICINA DE DESENVOLVIMENTO WEB

## 2º DIA





Bootstrap

---

# • Bootstrap

Framework HTML, CSS, e JS

---

## Na aula de hoje

---

- Introdução ao Bootstrap;
- Sistema de Grid;
- Atributos de visibilidade;
- Classes variadas;
- Exemplos e aplicação.



Bootstrap

---

# INTRODUÇÃO

Vamos partir do básico!

## O que é Bootstrap?

- Bootstrap é o **framework\* front-end** gratuito mais popular para desenvolvimento web. Ele permite que sejam criados projetos simples e atraentes de forma rápida, simples e fácil.
- Inclui diversos templates de design baseados em HTML e CSS para criação de formulários, botões, tabelas, menus, imagens, carousels e muito outros.
- Um dos principais pontos fortes do Bootstrap é ser um framework **responsivo**.

\* O objetivo de um framework é facilitar o desenvolvimento e manutenção de um projeto utilizando reaproveitamento de código (bibliotecas).



## Porque usar?

- Qualquer pessoa com conhecimento básico de HTML e CSS pode começar a usar o Bootstrap com **facilidade**;
- O CSS **responsivo** do Bootstrap se ajusta aos telefones, tablets e desktops;
- **Compátivel** com todos os browsers modernos;
- Possui vários **plugins** em JavaScript (jQuery) que facilitam o desenvolvimento;
- **Documentação** completa além de muito material para aprendizagem.

- <https://getbootstrap.com/getting-started/>
- <https://www.w3schools.com/bootstrap/>

## Como usar?

Existem duas maneiras de usar o Bootstrap em seu site:

- a. Baixar arquivos fonte em **getbootstrap.com**;
- b. Incluir arquivos do Bootstrap a partir de CDNs\*

\* Se você não deseja baixar e hospedar Bootstrap você mesmo, você pode incluí-lo a partir de um CDN (Content Delivery Network). O MaxCDN fornece suporte ao CDN para CSS e JavaScript do Bootstrap. Você também deve incluir o jQuery.

---

## a. Bootstrap incluído no projeto

---

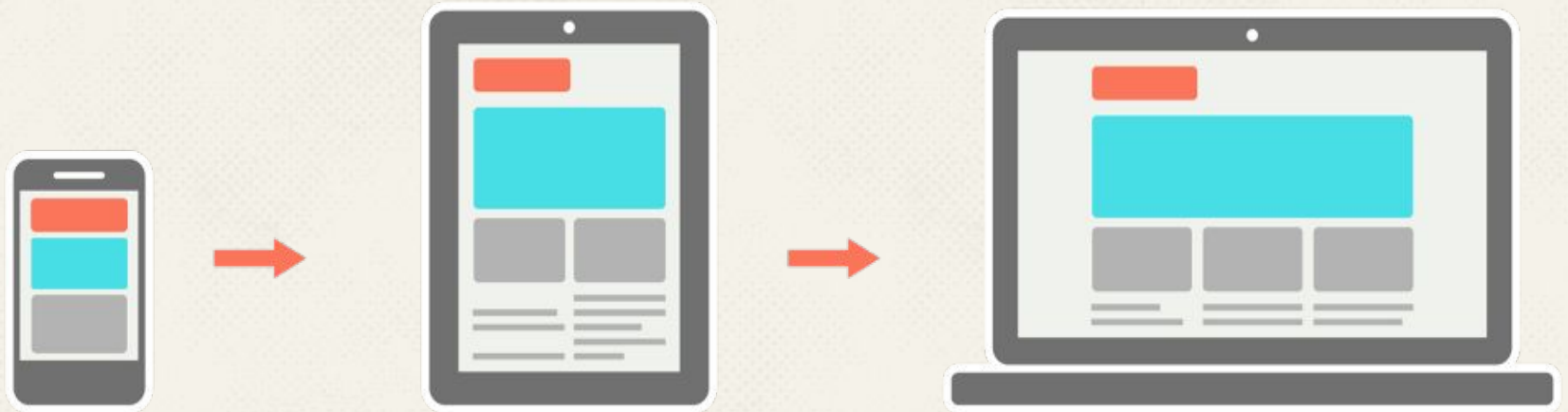
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Oficina dev. Web: Bootstrap</title>
  <link rel="stylesheet" href="_bootstrap/css/bootstrap.min.css">
</head>
<body>

  <script src="_js/jquery.min.js"></script>
  <script src="_bootstrap/js/bootstrap.min.js"></script>
</body>
</html>
```



## b. Usando CDN

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Oficina dev. Web: Bootstrap</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
</head>
<body>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js">
</script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js">
</script>
</body>
</html>
```



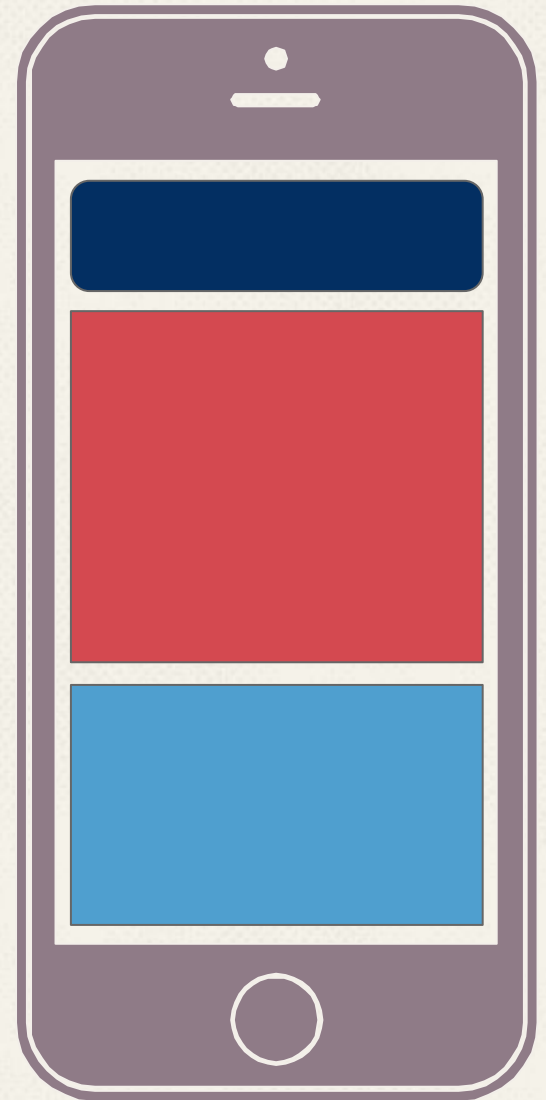
---

# GRID

Entendendo o sistema!

## SISTEMA DE GRID

- Possibilita a criação de um layout responsivo, com base no tamanho de tela do dispositivo utilizado.
- Um website com design responsivo é capaz de se adaptar em qualquer resolução de tela sem a necessidade de criar outro projeto paralelo para tal.



---

## Sistema de Grid do Bootstrap

---

- Três estruturas básicas: Containers, linhas e colunas.

### Containers

Bootstrap requer um elemento que possibilite envolver o conteúdo do site e adote o sistema de grid. O container é um componente que garante que o seu layout vai ficar alinhado corretamente na página.

Existem duas classes de containers disponíveis para uso:

1. `.container`
2. `.container-fluid`

# Containers

1. Use `.container` para conteúdo responsivo com largura limitada.

```
<div class="container">
```

```
...
```

```
</div>
```

2. Use `.container-fluid` para conteúdo responsivo com largura fluída, isto é, toda a largura da janela.

```
<div class="container-fluid">
```

```
...
```

```
</div>
```

Nota: Lembre-se que isso cria um padding a mais, então não deve-se incluir nenhum "container" dentro de outro.



# Linhas

- Uma **row** garante que todas as divs dentro dela apareçam em sua própria linha, fazendo uma divisão horizontal;
- Devem sempre estar dentro de um **container** ou **container-fluid** para um alinhamento e padding adequado.

```
<div class="container">  
  <div class="row">  
    ...  
  </div>  
  
  <div class="row">  
    ...  
  </div>  
  
</div>
```

## Colunas

- O conteúdo deve ser posto dentro de colunas, e apenas colunas podem ser filhas imediatas de uma linha (row).

Em resumo: `.container > .row > .col-*-*`

- O sistema de grade do Bootstrap permite até 12 colunas por linha.
- Para garantir a responsividade as colunas são organizadas na página de acordo com o **tamanho da tela**.

## Opções de grid

- O sistema de grid do Bootstrap possui 4 classes:
  - a) **xs** -> **Ex**tra **s**mall devices (smartphones)
  - b) **sm** -> **S**mall devices (tablets)
  - c) **md** -> **M**edium devices (desktops)
  - d) **lg** -> **L**arge devices (desktops maiores)

< 768px	<b>.col-xs-*</b>
>= 768px	<b>.col-sm-*</b>
>= 992px	<b>.col-md-*</b>
>= 1200px	<b>.col-lg-*</b>

## Estrutura

- Então, para criar o layout desejado:

- a) Crie o container

```
<div class="container">
```

- b) Crie as linhas

```
<div class="row">
```

- c) Adicione o número desejado de colunas para cada linha

```
<div class="col-*-*">
```

Base: `.col-x-y`, onde:

x -> tamanho da tela (classes grid)

y -> número de colunas (máx. 12)

## Exemplos

```
<div class="container">
  <div class="row">
    <div class="col-md-3" style="background-color:red;">col
  </div>

    <div class="col-md-3" style="background-color:blue;">col
  </div>

    <div class="col-md-3" style="background-color:green;">col
  </div>

    <div class="col-md-3" style="background-color:gray;">col
  </div>
  </div>
</div>
```



## Comportamento

- Use definições de tamanho de colunas para diferentes dispositivos.  
`<div class="col-md-3 col-sm-6 col-xs-12"> </div>`
- Observe o comportamento das colunas ao reduzir o tamanho da janela.

## Espaçamento entre colunas

- Usar colunas vazias para organizar colunas horizontalmente dentro `.row` é uma má prática;
- Em vez disso, mova as colunas para a direita usando as classes `.col-x-offset-y`, essas classes aumentam a margem esquerda de uma coluna por `y` colunas.

Exemplo de única coluna (tam. 4) ao meio da tela:

```
<div class="col-md-4 col-md-offset-4"> </div>
```



Bootstrap

---

## **Utilitários responsivos**

Exibição personalizada de conteúdo

## Exibição personalizada

- O Bootstrap fornece classes para mostrar ou ocultar conteúdos de acordo com o tamanho de tela em exibição.

### Hidden

- Esconde o elemento dependendo do tamanho da tela

```
<h1 class="hidden">Nunca aparece</h1>
```

```
<h1 class="hidden-xs bg-danger">Some em celulares</h1>
```

```
<h1 class="hidden-sm bg-info">Some em telas pequenas</h1>
```

```
<h1 class="hidden-md bg-warning">Some em telas medias</h1>
```

```
<h1 class="hidden-lg bg-success">Some em telas largas</h1>
```

- Criar uma coluna que não será exibida em celulares

```
<div class="col-md-6 col-md-offset-3 hidden-xs"> ... </div>
```

# Exibição personalizada

## Visible

- Exibe elemento de acordo com o tamanho da tela.

```
<h1 class="visible-xs-block">Mostra somente em mobile</h1>  
<h1 class="visible-sm-block">Mostra somente em tablet</h1>  
<h1 class="visible-md-block">Mostra somente em monitor medio</h1>  
<h1 class="visible-lg-block">Mostra somente em monitor grande</h1>
```

Estas propriedades são bastante comuns para exibição de conteúdos distintos para acessos via desktops e smartphones.





Bootstrap

---

# **Classes Bootstrap**

Estilos e exemplos



## As classes Bootstrap

- O Bootstrap define diversas classes para os mais variados elementos html.
- Você pode encontrar a listagem em <https://goo.gl/lJ4wPV>

Confira os exemplos encontrados nos códigos dentro da pasta  
'util'.



Bootstrap

---

## **Aplicação**

Dando continuidade ao projeto

**Obrigado!**

*Dúvidas?*

{ aalmeida, fkrein , isakis, jvmeller, rwfazul }  
@inf.ufsm.br