

# Unidades relativas e media-queries

# Mobile first



Responsive Web Design

---

Mobile First Web Design



Estruturar a página **pensando primeiro**  
em um **dispositivo móvel.**

# Unidades de medida relativas



Medidas relacionadas ao container pai direto.

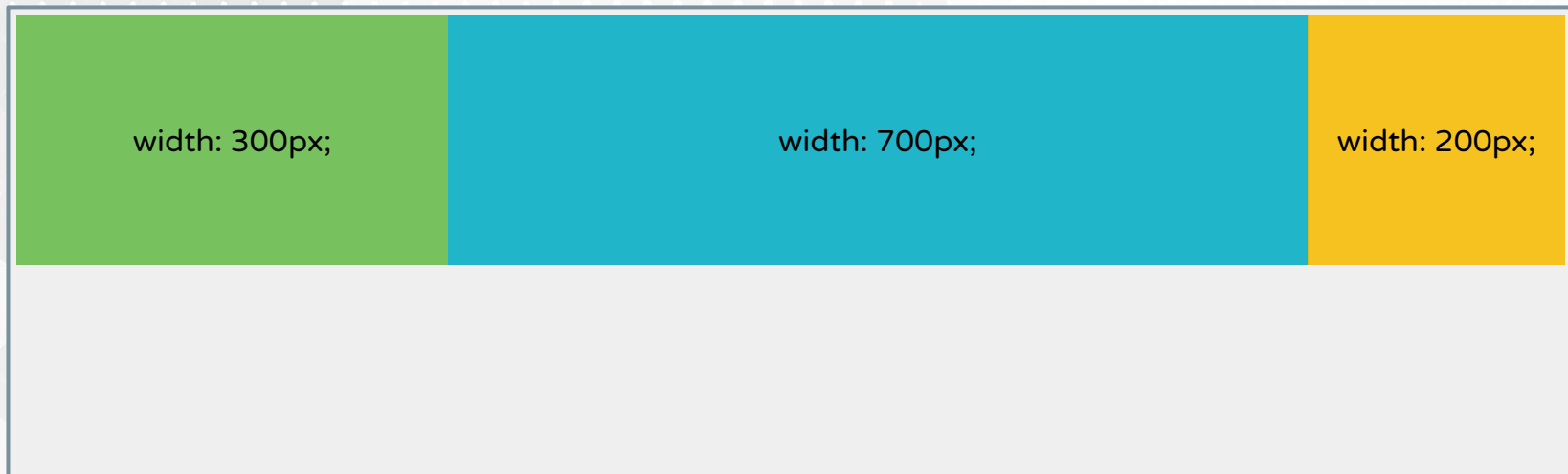
# Porcentagens - %

Uma medida expressa em porcentagem **SEMPRE** está relacionada à medida (no mesmo eixo) do **elemento pai** que a contém.

```
.box{  
  width: 25%;  
}
```

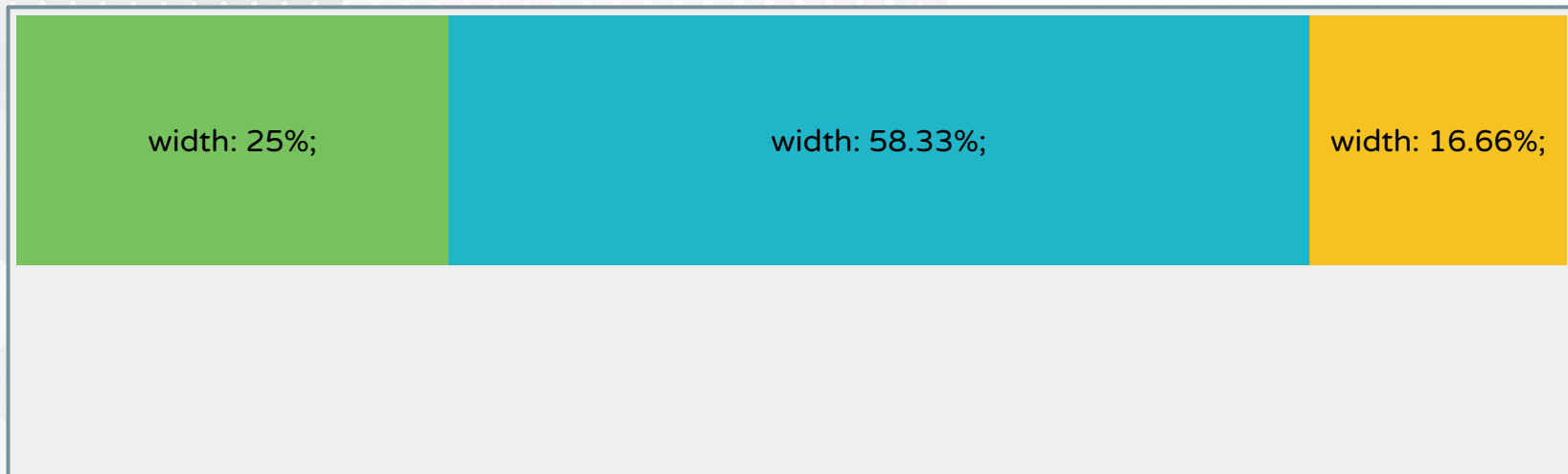
**Geralmente, não usamos porcentagens para a altura.**

# Pixels em percentual



container pai com **1200px** em width

# Pixels em percentual



container pai com **1200px** em width

# Pixels em percentual

width: 25%;  
 $(300/1200)*100$

width: 58.33%;  
 $(700/1200)*100$

width: 16.66%;  
 $(200/1200)$   
 $*100$

Nesse caso, nossa “base” é **1200px**.

# Medida **em**

Os em's são **unidades** de medida que **devem ser usadas** para tudo o que tiver relação com **tipografias**.

**1em** começará sendo igual a **16px** (a menos que configuremos outra coisa)

```
p {  
    font-size: 1em;  
    line-height: 1.5em;  
}
```





```
<body>  
  <div>  
    <p>  
      <strong><strong>  
    </p>  
  </div>  
</body>
```

Diagram illustrating font-size inheritance in HTML:

- `<body>` is assigned `font-size: 16px;`
- `<div>` is assigned `font-size: 1em;`
- `<p>` is assigned `font-size: 1em;`
- `<strong><strong>` is assigned `font-size: 1em;`

`<body>` tem 16px em font-size, os outros elementos, por padrão, vêm com 1em em font-size.

Portanto, os demais elementos têm 16px de font-size.



```
<body>  
  <div>  
    <p>  
      <strong><strong>  
    </p>  
  </div>  
</body>
```

Diagram illustrating font-size inheritance:

- `<body>` is linked to `font-size: 16px;` (green arrow).
- `<div>` is linked to `font-size: 2em;` (red arrow).
- `<p>` is linked to `font-size: 1em;` (green arrow).
- `<strong><strong>` is linked to `font-size: 1em;` (green arrow).

Se alteramos o font-size de `<div>` para 2em, ele terá 2 vezes o tamanho de font-size do pai.

`<div>` agora tem 32px de font-size.

Portanto, agora, os filhos de `<div>` mudaram porque alteramos o pai.

Os filhos de `<div>` ficaram com 32px de font-size.



```
<body>  
  <div>  
    <p>  
      <strong><strong>  
    </p>  
  </div>  
</body>
```

Diagram illustrating font-size inheritance in HTML:

- `<body>` has a font-size of 16px (indicated by a green arrow).
- `<div>` has a font-size of 2em (indicated by a red arrow).
- `<p>` has a font-size of 1.5em (indicated by a red arrow).
- `<strong>` has a font-size of 1em (indicated by a green arrow).

`<div>` tem 32px de font-size.

Se definimos para `<p>` o font-size: 1.5em, `<p>` terá 1,5 vezes o tamanho do font-size do pai.

`<p>` agora tem 48px em font-size.

`<strong>` tem 1 vez o font-size do pai.

`<p>` tem 48px em font-size.



```
<body>  
  <div>  
    <p>  
      <strong><strong>  
    </p>  
  </div>  
</body>
```

Diagram illustrating font-size inheritance:

- `<body>` is linked to `font-size: 10px;` (red arrow).
- `<div>` is linked to `font-size: 2em;` (red arrow).
- `<p>` is linked to `font-size: 1.5em;` (red arrow).
- `<strong>` is linked to `font-size: 1em;` (green arrow).

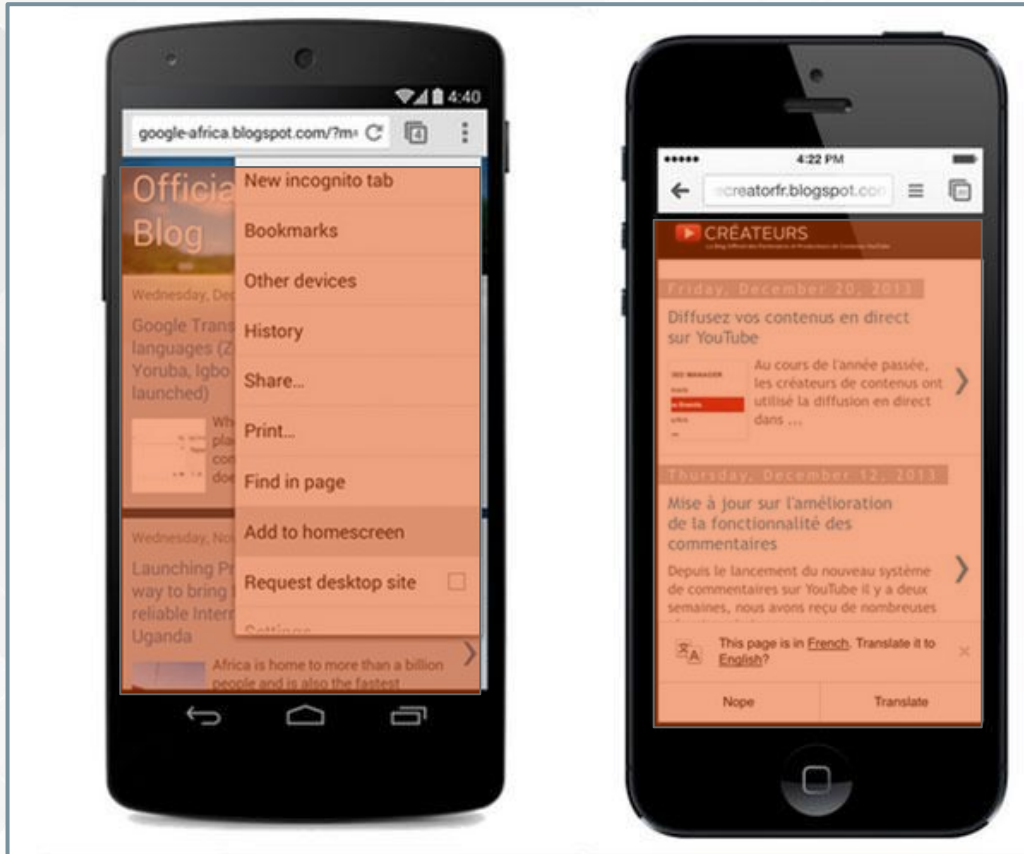
Agora, se a base do documento (`<body>`) for alterada, com 10px de font-size, todos os elementos mudarão de tamanho, mesmo sem que o font-size deles seja alterado.

`<div>` = 20px de font-size

`<p>` = 30px de font-size

`<strong>` = 30px de font-size

# O Viewport



# Tag para RWD

A tag **<meta> viewport** dá instruções ao navegador sobre **como dimensionar** e escalar a página web durante o carregamento.

```
<meta name="viewport"  
      content="width=device-width, initial-scale=1">
```

Esta é a estrutura **básica** desta tag, às vezes ela pode ter mais informações.  
Saiba mais: [Responsive Web Design](#)




# Viewport Measures - vw / vh

Toda medida expressa em **vw/vh** terá **SEMPRE** como eixo de referência o **viewport** do documento.

```
.box {  
    width: 25vw;  
    height: 50vh;  
}
```

Viewport faz referência à “caixa visível” de conteúdo dentro de um navegador.



A diagram illustrating a viewport layout. It features a large gray rectangle representing the viewport. Inside the top-left corner of this rectangle is a smaller blue rectangle. The blue rectangle is labeled with its dimensions: 'width: 30vw;' and 'height: 50vh;'. The gray rectangle is labeled with its dimensions: 'viewport width de 320px' at the bottom and 'viewport height de 450px' on the right side. The background of the slide has a light gray geometric pattern of squares and triangles, with decorative clusters of red, white, and blue triangles in the corners.

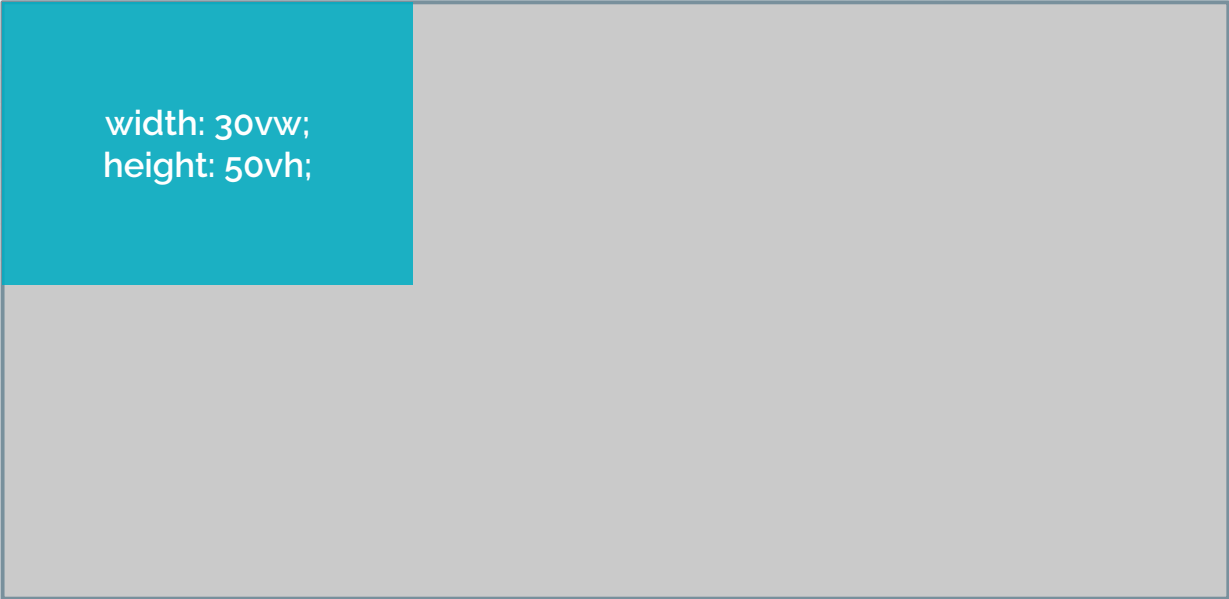
width: 30vw;  
height: 50vh;

viewport height de 450px

Neste caso, a caixa terá 0,3 vezes a largura do viewport (**96px**) e 0,5 vezes a altura do viewport (**225px**)

viewport width de 320px





A diagram showing a large gray rectangle representing a viewport. In the top-left corner of this rectangle is a smaller teal rectangle. Inside the teal rectangle, the text 'width: 30vw; height: 50vh;' is written in white. To the right of the gray rectangle, the text 'viewport height de 250px' is written vertically in red. Below the gray rectangle, the text 'viewport width de 660px' is written in red. The background of the slide features a light gray grid with a pattern of small dots and larger squares, and decorative geometric patterns of red, white, and blue triangles in the corners.

```
width: 30vw;  
height: 50vh;
```

viewport height de 250px

viewport width de 660px

Neste caso, a caixa terá 0,3 vezes a largura do viewport (**198px**) e 0,5 vezes a altura do viewport (**125px**)

# Media-queries

Conjunto de regras CSS que permitem reorganizar o conteúdo de acordo com as condições de visualização do documento.

Sempre devem ser escritos no final da folha de CSS.

With Breakpoints



Without Breakpoints



# Media-queries

para Mobile **First**

```
@media (min-width: 460px) {  
    /* regras de CSS */  
}
```

Especificar a **min-width** é como dizer: “se o mínimo é Npx de largura, aplicar isto” ou “deste ponto para cima”.

# Media-queries

para Mobile **Last**

```
@media (max-width: 960px) {  
    /* regras de CSS */  
}
```

Especificar a **max-width** é como dizer: “se o máximo é Npx de largura, aplicar isto” ou “deste ponto para baixo”.

# Media-queries

definindo a **orientação**

```
@media (max-width: 768px) and (orientation: portrait){  
  /* regras de CSS */  
}
```

Especificar **a orientação (portrait ou landscape)**, é como dizer “se a largura máxima é Npx e o dispositivo está em posição vertical/horizontal, aplicar isto”.

# Bootstrap - Framework para layouts



# Bootstrap - Container

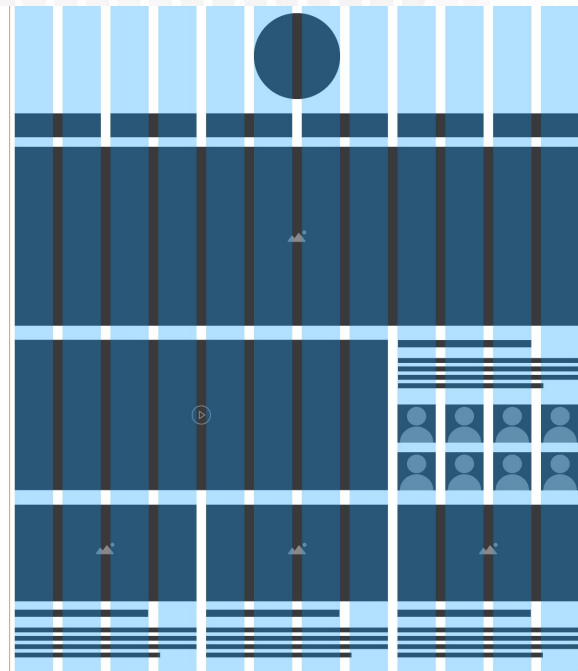
```
<div class="container">
```

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

Viewport	Sigla	largura .container
Mobile (<576px)	.col-	100%
Tablets (≥576px)	.col-sm	540px
Desktop (≥768px)	.col-md	720px
Large Desktop (≥992px)	.col-lg	960px
Extra Large (≥1200px)	.col-xl	1140px

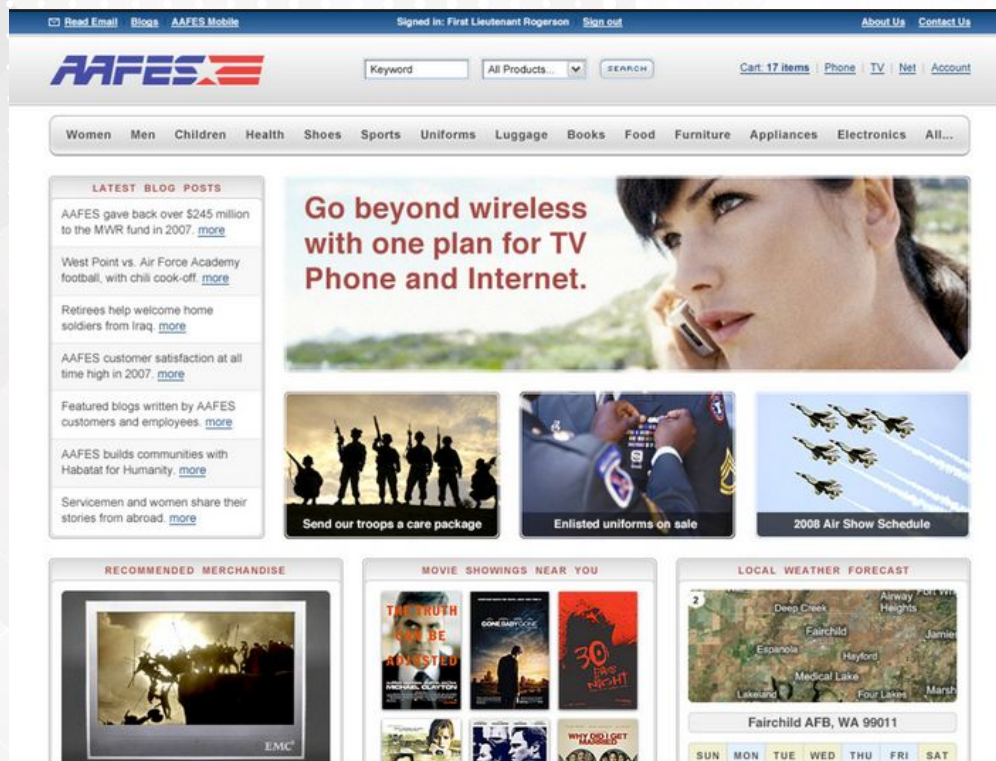
# Bootstrap - Grid

Grid, ou grade em português, é a maneira que o bootstrap divide sua tela.

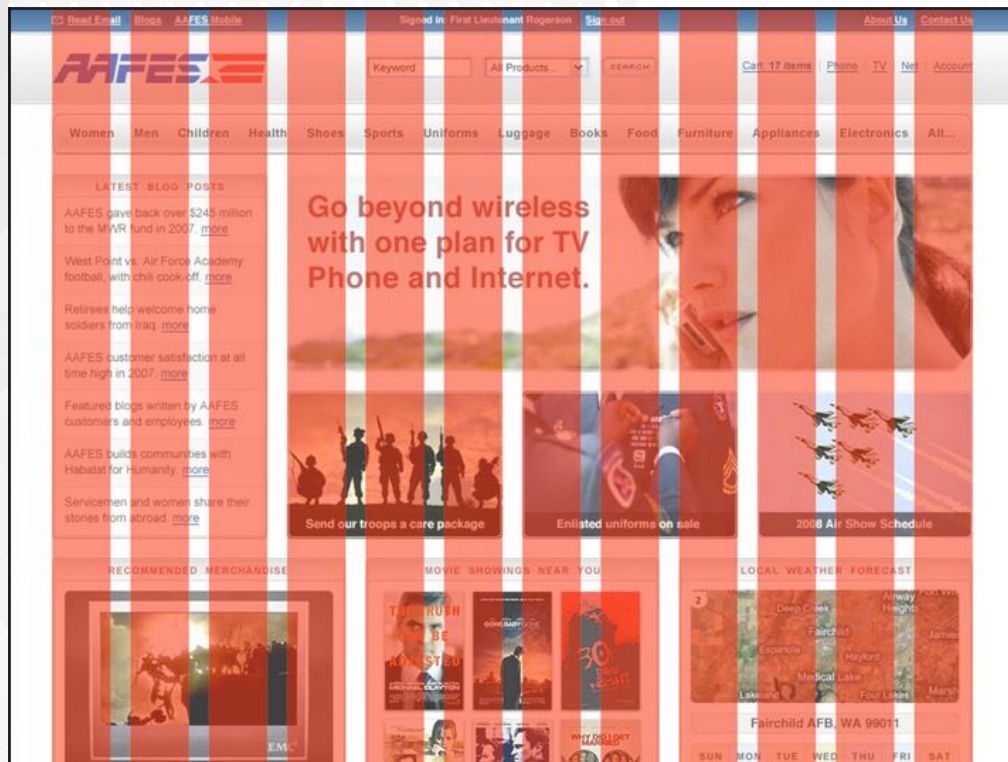




# Bootstrap - Grid



# Bootstrap - Grid



# Bootstrap - Grid básica

```
<div class="row">  
  <div class="col-lg-8">  
    ...  
  </div>  
  <div class="col-lg-4">  
    ...  
  </div>  
</div>
```

# Bootstrap - Grid básica

Grade básica para diferentes viewports.

```
<div class="row">  
  <div class="col-lg-8 col-md-6 col-12">  
    ...  
  </div>  
  <div class="col-lg-4 col-md-6 col-12">  
    ...  
  </div>  
</div>
```

# Bootstrap - Grid aninhadas

```
<div class="row">  
  <div class="col-lg-8 col-md-6 col-12">  
    <div class="row">  
      <div class="col-lg-6">...</div>  
      <div class="col-lg-6">...</div>  
    </div>  
  </div>  
</div>
```



# Bootstrap - Instalação

## Opção 1 - Baixar código

Download:

<http://getbootstrap.com/getting-started/#download>

Permite a personalização:

<http://getbootstrap.com/customize/>

# Bootstrap - Instalação

## Opção 2 - Importar código com CDN

Usando CDN

<http://bootstrapcdn.com>

```
<link href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/  
css/bootstrap.min.css" rel="stylesheet">
```

# OUTROS FRAMEWORKS

Zurb Foundation

<http://foundation.zurb.com>

BassCss

<http://www.basscss.com/>

Tachyons

<http://tachyons.io>

Materilize

<https://materializecss.com/>

Material Design (Google)

<https://material.io/design/>

Semantic UI

<http://semantic-ui.com/>



Obrigada!

