

# Unidades relativas e media-queries

**O que aprendemos na  
aula anterior?**

Flexbox



# 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.

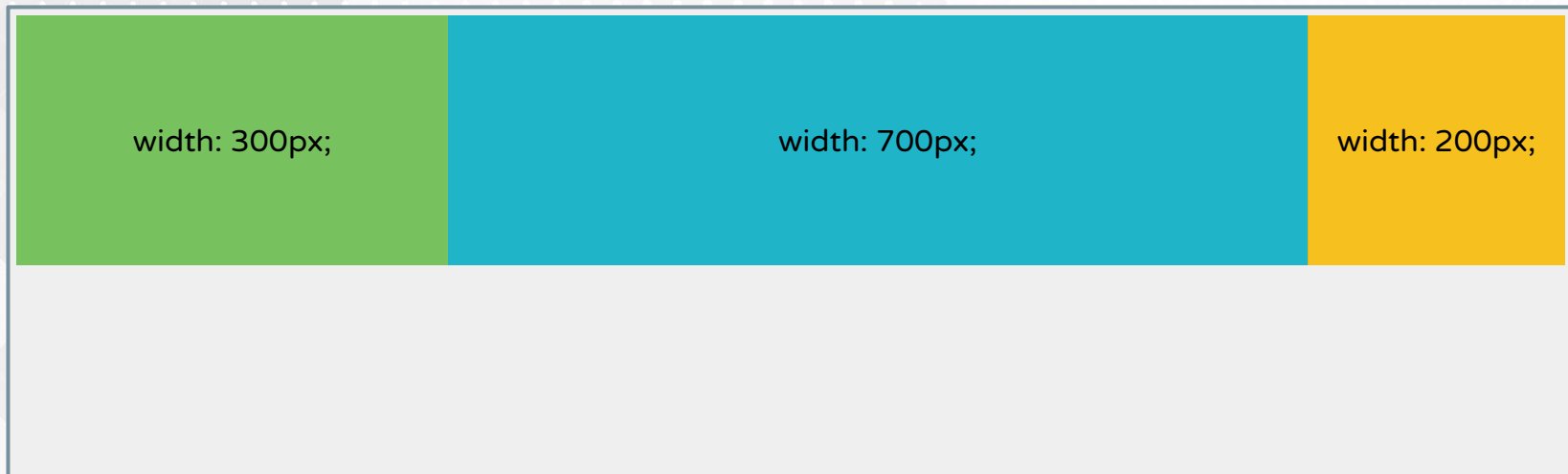
# Porcentagens - %

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

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

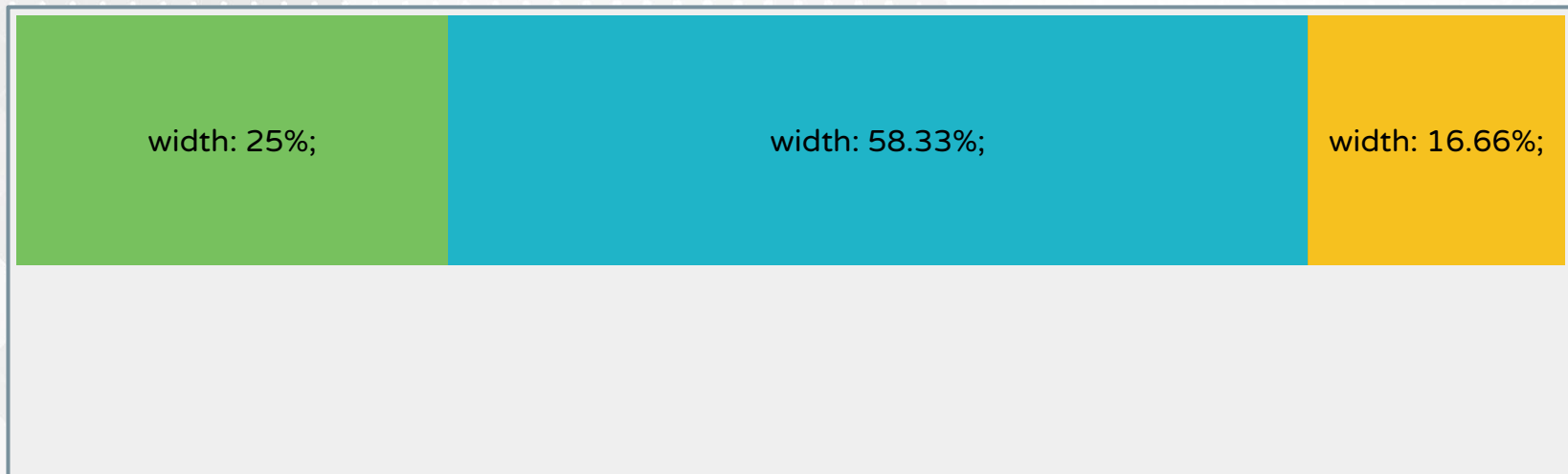
**Geralmente, não usamos porcentagens para a altura.**

# 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>
```

font-size: 16px;

font-size: 1em;

font-size: 1em;

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>
```

font-size: 16px;

font-size: 2em;

font-size: 1em;


font-size: 1em;

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>
```

font-size: 16px;

font-size: 2em;

font-size: 1.5em;

font-size: 1em;

<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>
```

font-size: 10px;

font-size: 2em;

font-size: 1.5em;

font-size: 1em;

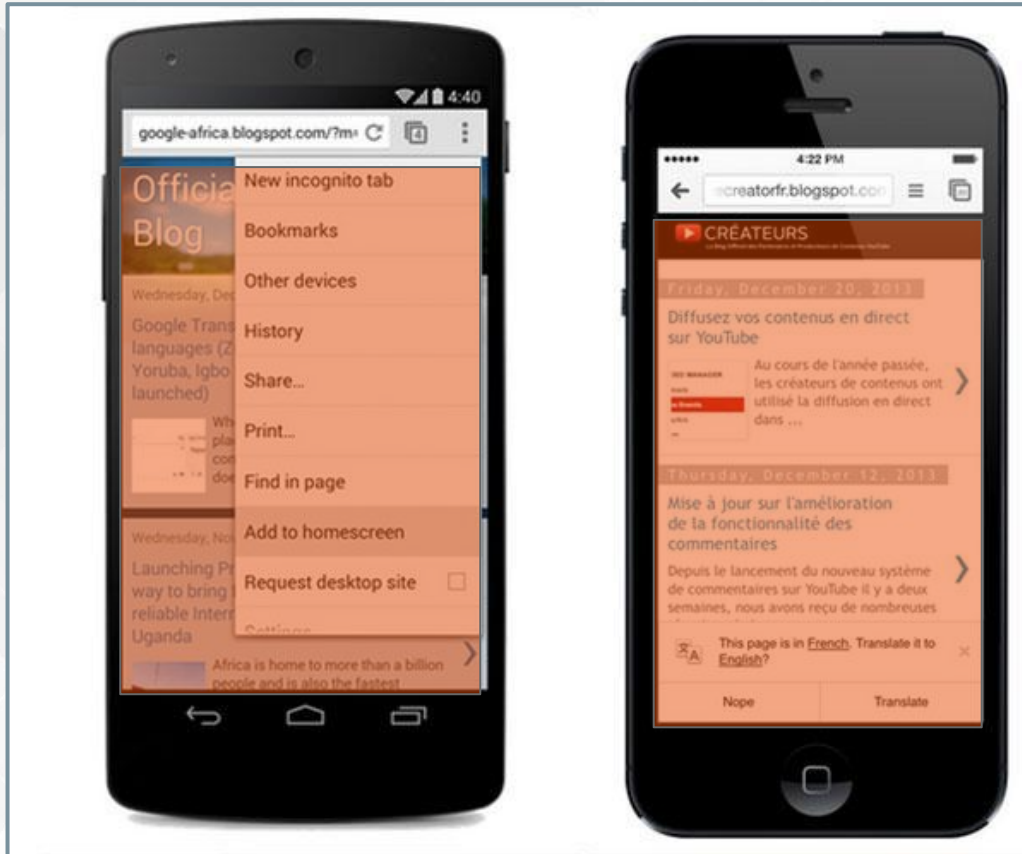
Agora, se a base do documento (<body>) for alterada, com 10px de font-size, todos os elementos mudarão de tamanho, mesmo sem que o font-size deles seja alterado.

<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.



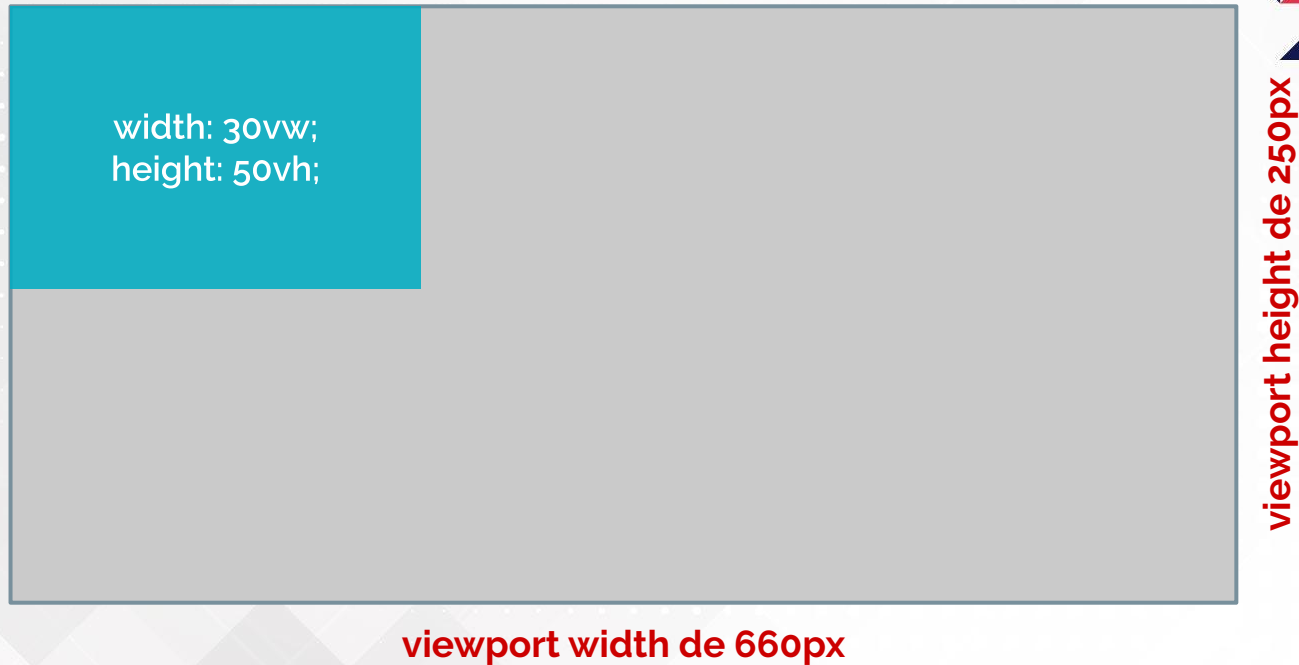


width: 30vw;  
height: 50vh;

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

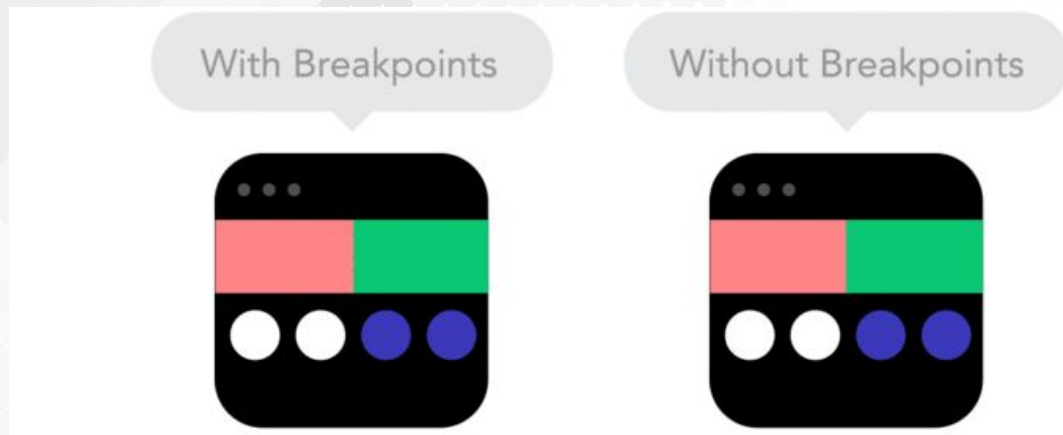


Neste caso, a caixa terá 0,3 vezes a largura do viewport (**198px**) e 0,5 vezes a altura do viewport (**125px**)

# 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.



# 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”.

# Na próxima aula

Visualização e  
pseudo-elementos

