



Medidas Relativas



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School

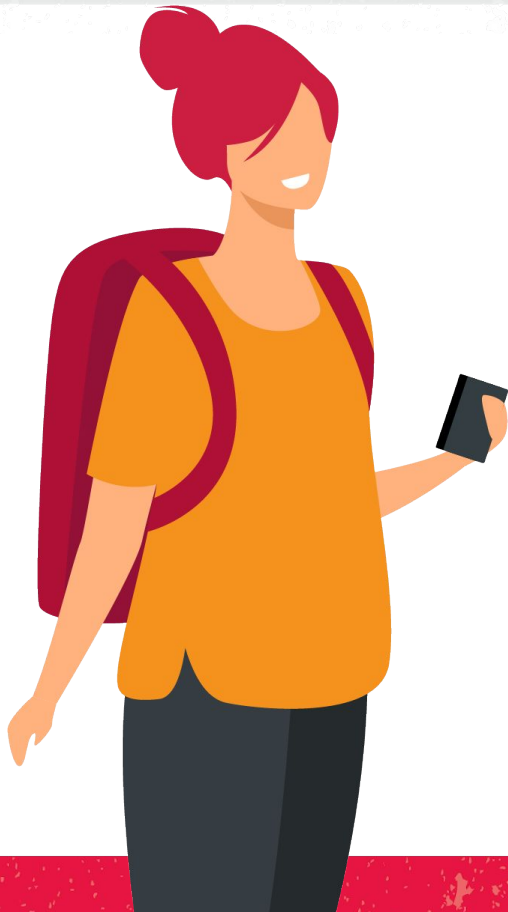


**Tema
da aula:**

- 1** Contexto
- 2** Porcentagens
- 3** em | rm
- 4** vw | vh

1

Contexto



“

As medidas relativas são aquelas que levam em conta todo o contexto da página que são inseridas. Se o contexto da página permitir, as medidas relativas vão se adaptar nesse contexto.

”



O Contexto:

Dependendo das medidas que falamos anteriormente, o número inserido poderá ser:

- O container pai.
- O tamanho da fonte do site.
- O tamanho da fonte do container pai.
- O tamanho do viewport.

Se tomamos as porcentagens como exemplo, podemos decidir que são uma **unidade relativa**, já que **30%** de alguma medida não será a mesma para um container de 2000px de largura, e para um container de 1000px de largura.

2 | Porcentagens



As porcentagens:

Qualquer medida inserida em porcentagem **sempre** estará **relacionada** com a medida do **elemento pai** que a contém.

Se o container **pai** mede **300px** de largura, e atribuirmos a largura de **50%** ao elemento interior(filho), este medirá **150px** (os 50% da largura pai).

Obs: não é recomendado usar porcentagens para a altura de um elemento.

CSS

```
.elementoContainerPai { width: 300px }  
.elementoInteriorFilho { width: 50% } // Será 150px
```

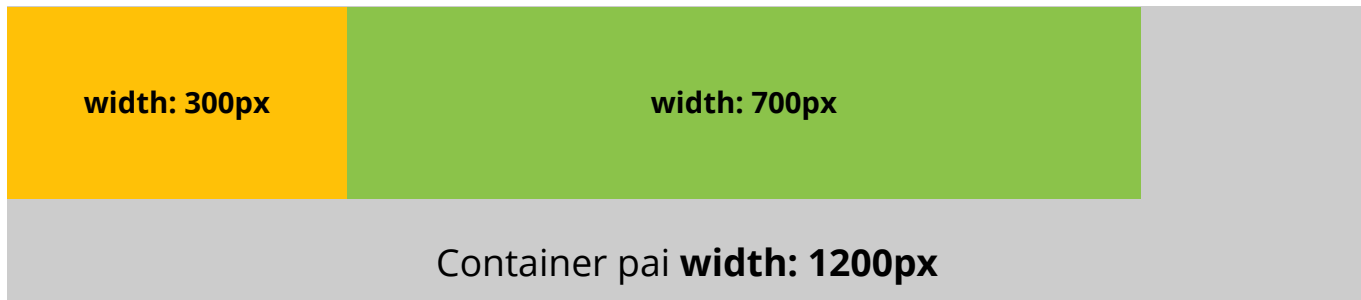


Calculando as porcentagens:

Uma boa forma de calcular as porcentagens se dá por meio da regra de 3 simples:

Para cada elemento do código o cálculo seria:

`elemento filho` * 100 / `elemento pai` = porcentagem





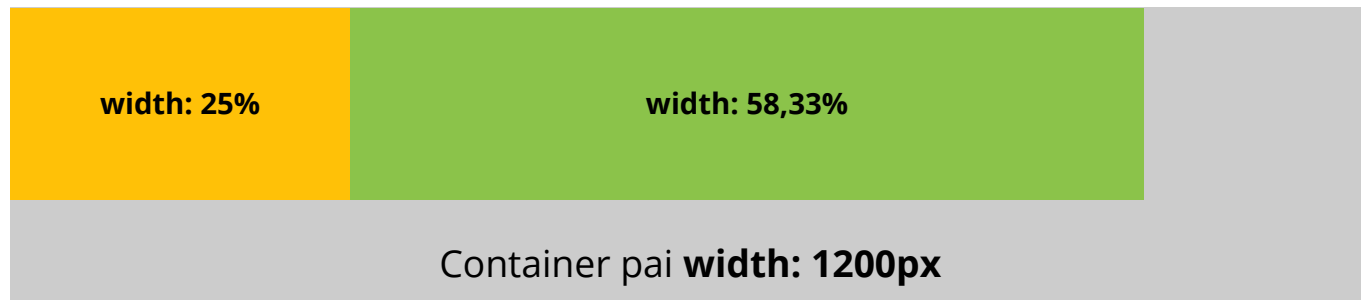
Porcentagens:

Então, se calcularmos cada elemento, este seria o resultado:

$$300 * 100 / 1200 = 25\%$$

$$700 * 100 / 1200 = 58.33\%$$

$$200 * 100 / 1200 = 16.66\%$$



3

em l rem



Como funcionam os em's

Os **em's** são sempre **relativos ao elemento pai**. Tomarão como valor de referência a propriedade **font-size**.

O ponto de referência inicial é o valor do **font-size** do elemento `<html>`, que por padrão é **16px**. O restante dos elementos terão **1em** de **font-size**, que equivale ao mesmo valor que tem o elemento pai.

Se associarmos **1.5em** ao font-size de qualquer elemento, o tamanho resultante será o valor do elemento pai multiplicado pelo valor em em's → **16px * 1.5 = 24px**.

```
css    p { font-size: 1.5em } // 16px * 1.5 = 24px
```



Como funcionam os em's

Se utilizamos em's na propriedade que não tem **font-size**, será utilizado para cálculo o font-size do elemento que estamos modificando.

Geralmente, se utiliza esse tipo de unidade para tudo que está relacionado com a tipografia, mas também podemos aplicar a outras propriedades, como margin e padding para que variem em função dos tamanhos das fontes.

```
css  p {  
    font-size: 20px;  
    line-height: 2em; // 20px * 2 = 40px  
    padding: 1.5em; // 20px * 1.5 = 30px  
}
```



Calculando os em's dos elementos

Por padrão `<html>` terá **16px** de font-size e os demais elementos **1em**.

Porém, todos os elementos do site terão **16px** de font-size.

```
<html> → font-size: 16px
  <body> → font-size: 1em → 16px
    <div> → font-size: 1em → 16px
      <p> → font-size: 1em → 16px
        <strong>Olá Mundo!</strong> → font-size: 1em → 16px
      </p>
    </div>
  </body>
</html>
```



Calculando os em's dos elementos

Se inserimos o font-size da `<div>` em **2em**, essa div terá 2 vezes o tamanho do font-size do seu elemento pai.

Porém, a `<div>` e todos seus elementos filhos terão **32px** de font-size.

```
<html> → font-size: 16px
  <body> → font-size: 1em → 16px
    <div> → font-size: 2em → 16px * 2 → 32px
      <p> → font-size: 1em → 32px
        <strong>Olá Mundo!</strong> → font-size: 1em → 32px
      </p>
    </div>
  </body>
</html>
```



Calculando os em's dos elementos

Então, se inserimos um **font-size** de **1.5em** para a TAG `<p>`, o cálculo terá como base o tamanho de seu elemento pai, a `<div>`: $32px * 1.5 = 48px$.

Portanto, todos os elementos filhos de `<p>` herdarão esse tamanho de fonte.

```
<html> → font-size: 16px
  <body> → font-size: 1em
    <div> → font-size: 2em → 32px
      <p> → font-size: 1.5em → 32px * 1.5 → 48px
        <strong>Olá Mundo!</strong> → font-size: 1em → 48px
      </p>
    </div>
  </body>
</html>
```



Calculando os em's dos elementos

Se agora colocarmos o font-size do elemento `<html>` a **10px**, todos os elementos mudarão de tamanho sem nem mesmo ter modificado diretamente seus font-size.

```
<html> → font-size: 10px
  <body> → font-size: 1em → 10px
    <div> → font-size: 2em → 20px
      <p> → font-size: 1.5em → 30px
        <strong>Olá Mundo!</strong> → font-size: 1em → 30px
      </p>
    </div>
  </body>
</html>
```




Como funcionam os rem's

Como vimos, utilizar os em's pode ser muito complicado.

Os **rem's** funcionam muito parecido com os em's, com a diferença de que sempre usarão de base o tamanho do font-size do elemento `<html>`.

Isso quer dizer que o tamanho expressado em rem's não modificará o tamanho dos elementos filhos, e nem será afetado pelo tamanho do elemento pai.

Geralmente é melhor usar rem's no lugar dos em's já que conservamos as vantagens de uma unidade relativa, além de evitar fazer cálculos complexos e ficarmos preocupados de como eles alterarão os outros elementos.

css

```
p { font-size: 1.5rem }
```



Calculando os rem's dos elementos

Se agora mudamos o font-size do elemento `<html>` para **10px**, todos os elementos mudarão de tamanho sem ter modificado diretamente seu font-size.

```
<html> → font-size: 10px
  <body> → font-size: 1em → 10px
    <div> → font-size: 2em → 20px
      <p> → font-size: 1.5em → 30px
        <strong>Olá Mundo!</strong> → font-size: 1em → 30px
      </p>
    </div>
  </body>
</html>
```

3

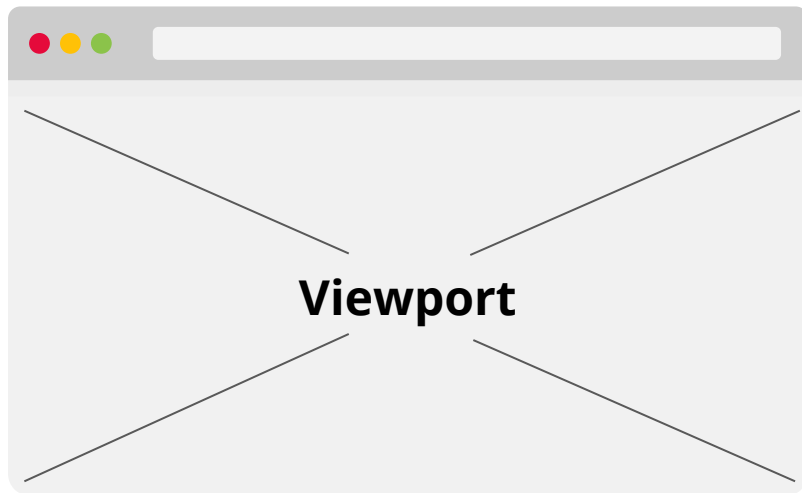
vw | vh



Medidas de Viewport

O viewport é o espaço visível que o navegador tem para mostrar o site.

Isso quer dizer que se pode utilizar medidas relativas neste espaço para poder determinar o tamanho de certos elementos.





Viewport width e height

vw o viewport width é relativo à largura total do viewport.

vh o viewport height é relativo à altura total do viewport.

Ambas medidas estão expressas como porcentagens do total. Isso quer dizer que **50vh** será equivalente à 50% da altura disponível no viewport.

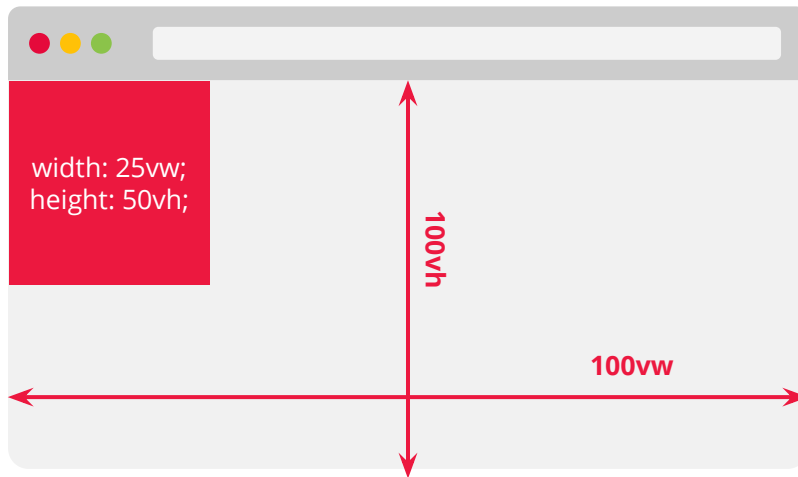
CSS

```
div {  
  width: 25vw; // 25% da largura disponível  
  height: 50vh; // 50% da altura disponível  
}
```



Visualizando o vh e o vw

Qualquer medida expressa em viewport width (vw) ou viewport height (vh) tomará **sempre** como referência o viewport do documento.



DigitalHouse>
Coding School