

Responsividade

- Texto
- Media Queries
- GridView
- Flexbox



Responsividade

$1vw = 1\%$ da largura do viewport

$1vh = 1\%$ da altura do viewport

$1vmin = 1vw$ ou $1vh$, o que for menor

$1vmax = 1vw$ or $1vh$, o que for maior

Responsividade

```
<!DOCTYPE html>
<html>
<head>
<title>Teste de HTML</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
<h1>Fontes com diferentes tamanhos</h1>
<p>Parágrafo padrão</p>
<p style="font-size:12px;">Parágrafo com 12 px</p>
<p style="font-size: calc(12px + 1.5vw);">Parágrafo com 12px + 1.5vw</p>
<p style="font-size: calc(12px + 1.5vh);">Parágrafo com 12px + 1.5vh</p>
<p style="font-size:80%;">Parágrafo com fonte 80%</p>
<p style="font-size:100%;">Parágrafo com fonte 100%</p>
<p style="font-size:120%;">Parágrafo com fonte 120%</p>
<p style="font-size:1.5em;">Parágrafo com fonte 1.5em</p>
</body>
</html>
```

Responsividade

Fontes com diferentes tamanhos

Parágrafo padrão

Parágrafo com 12 px

Parágrafo com 12px + 1.5vw

Parágrafo com 12px + 1.5vh

Parágrafo com fonte 80%

Parágrafo com fonte 100%

Parágrafo com fonte 120%

Parágrafo com fonte 1.5em

Responsividade

Media Query

- *Media query* é uma técnica CSS introduzida no CSS3.
- A regra @media é usada para incluir um bloco de propriedades CSS somente se uma determinada condição for verdadeira.

Responsividade

Media Query (consulta de Mídia)

Pode ser usada, por exemplo, para verificar:

- largura e altura do dispositivo
- orientação (paisagem ou retrato?)
- resolução



Responsividade

Media Queries

- Além de redefinir os tamanhos das imagens e textos, é possível definir diferentes estilos de páginas para navegadores de tamanhos diferentes.
- MQ é uma técnica introduzida em CSS3 que usa **@media** para definir uma propriedade quando uma determinada condição é verdadeira.

```
<style>
body {
    background-color: lightgreen;
}
@media (max-width: 800px) {
    body {
        background-color: lightblue;
    }
}
</style>
```

Responsividade

Media Query (consulta de Mídia)

```
body {  
  background-color: lightblue;  
}  
@media screen and (min-width: 400px) {  
  body {  
    background-color: lightgreen;  
  }  
}  
@media screen and (min-width: 800px) {  
  body {  
    background-color: lavender;  
  }  
}
```


Responsividade

Media Queries

```
body { font-size: 14px; }
```

```
@media (min-width:600px) { body { font-size: 16px; } }
```

```
@media (min-width:800px) { body { font-size: 18px; } }
```

```
@media (min-width:1000px) { body { font-size: 20px; } }
```



Responsividade

Retrato X Paisagem

```
<style>
body {
    background-color: lightgreen;
}

@media (orientation: landscape) {
    body {
        background-color: lightblue;
    }
}
</style>
```

Responsividade

CSS Syntax

```
@media not|only mediatype and (mediafeature and|or|not mediafeature) {  
    CSS-Code;  
}
```

meaning of the **not**, **only** and **and** keywords:

not: The not keyword reverts the meaning of an entire media query.

only: The only keyword prevents older browsers that do not support media queries with media features from applying the specified styles. **It has no effect on modern browsers.**

and: The and keyword combines a media feature with a media type or other media features.

They are all optional. However, if you use **not** or **only**, you must also specify a media type.

https://www.w3schools.com/cssref/css3_pr_mediaquery.asp

Responsividade

Media Queries

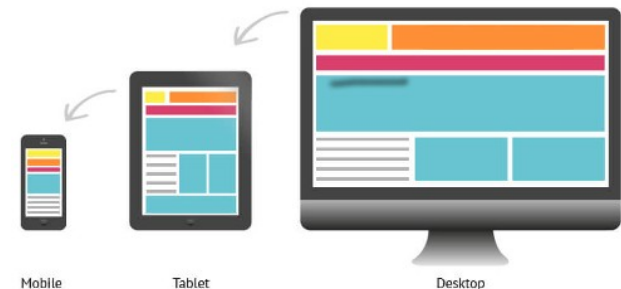
- Exemplos:

1. https://www.w3schools.com/css/tryit.asp?filename=tryresponsive_mediaquery

2. https://www.w3schools.com/html/tryit.asp?filename=tryhtml_responsive_media_query

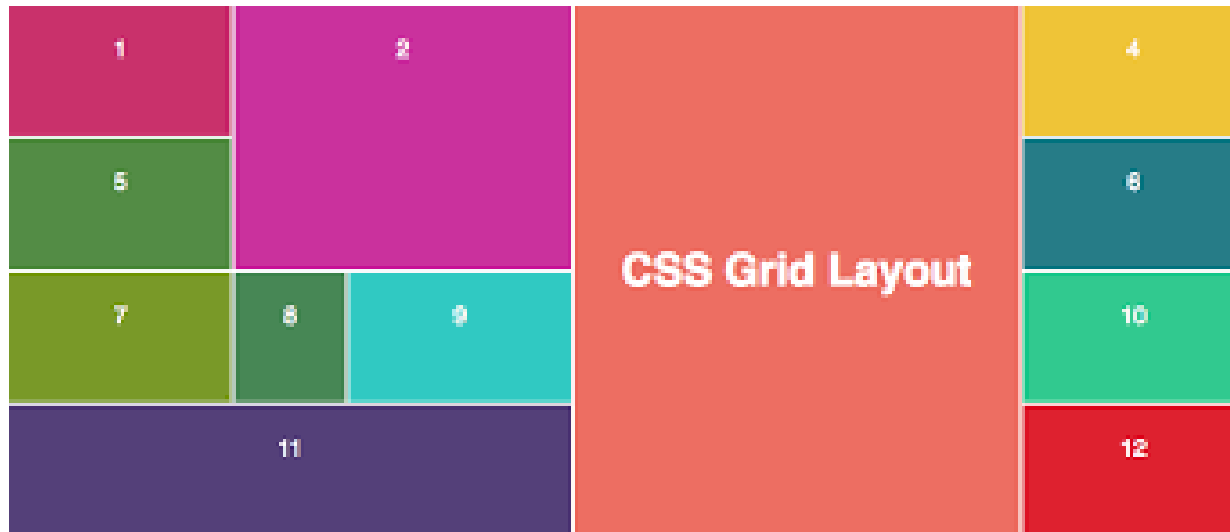
3. https://www.w3schools.com/css/tryit.asp?filename=tryresponsive_breakpoints

4. https://www.w3schools.com/css/css3_mediaqueries_ex.asp



Responsividade

GridView



Veja https://www.w3schools.com/css/css_grid.asp

Responsividade

```
<!DOCTYPE html>
<html>
<head>
<title>Teste de HTML</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
.blocao {
  display: grid;
  grid-template-columns: 200px 300px 400px;
  grid-gap: 20px;
  background-color: green;
}
.bloco {
  color: black; border-radius: 10px;
  padding: 20px; font-size: 150%;
}
</style>
</head>
```

Responsividade

```
<body>
<div class="blocao">
  <div class="bloco" style="background-color: blue;">Um</div>
  <div class="bloco" style="background-color: yellow;">Dois</div>
  <div class="bloco" style="background-color: salmon;">Três</div>
  <div class="bloco" style="background-color: red;">Quatro</div>
  <div class="bloco" style="background-color: orange;">Cinco</div>
  <div class="bloco" style="background-color: violet;">Seis</div>
</div>
</body>
</html>
```



Responsividade

```
<!DOCTYPE html>
<html>
<head>
<title>Teste de HTML</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
html { font-size: 22px; }
body { padding: 5px; }
.bloco {
  color: black; padding: 10px; height: 100px; font-weight: bold;
}
.blocos {
  max-width: 1200px; margin: 0 auto;
  display: grid; gap: 10px;
  grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
}
</style>
</head>
```

A unidade fr representa uma fração do espaço disponível no container do grid.

Responsividade

```
<body>
<div class="blocos">
  <div class="bloco" style="background-color: blue;">Um</div>
  <div class="bloco" style="background-color: yellow;">Dois</div>
  <div class="bloco" style="background-color: red;">Três</div>
  <div class="bloco" style="background-color: green;">Quatro</div>
  <div class="bloco" style="background-color: chocolate;">Cinco</div>
  <div class="bloco" style="background-color: orange;">Seis</div>
  <div class="bloco" style="background-color: olive;">Sete</div>
  <div class="bloco" style="background-color: violet;">Oito</div>
  <div class="bloco" style="background-color: salmon;">Nove</div>
  <div class="bloco" style="background-color: LightGreen;">Dez</div>
  <div class="bloco" style="background-color: DeepPink;">Onze</div>
  <div class="bloco" style="background-color: grey;">Doze</div>
</div>

</body>
</html>
```

Responsividade

Um	Dois	Três
Quatro	Cinco	Seis
Sete	Oito	Nove
Dez	Onze	Doze

Responsividade



Flexbox é um modelo de layout da Web CSS3

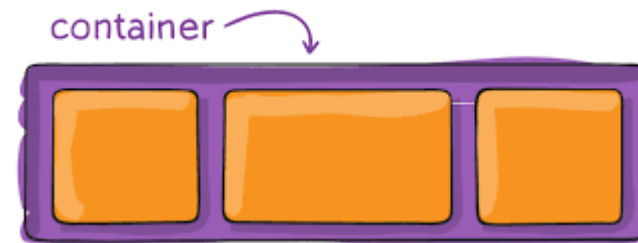
Responsividade

Características

- Permite a distribuição de espaços entre itens de uma interface.
- É um layout de uma dimensão (linha ou coluna). Um modelo de duas dimensões: *Grid Layout*.
- Permite definir layout flexível sem usar *float* ou *positioning* (*Media Queries*).
- Uma área de um documento usando o flexbox é chamada de *flex container*.
- Os itens a serem organizados são chamados de *flex itens*.

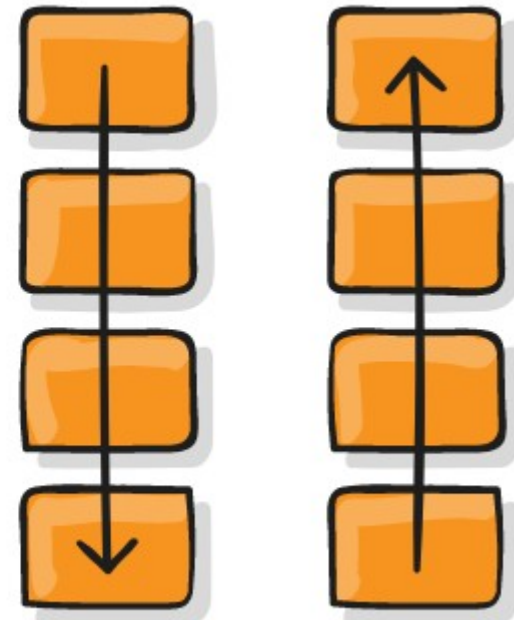
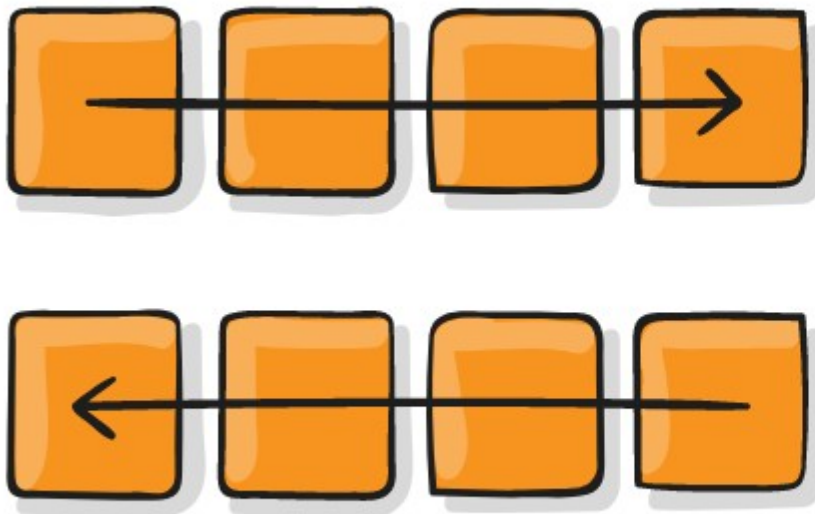
Responsividade

- Propriedades:
 - ♦ flex-direction
 - ♦ flex-wrap
 - ♦ flex-flow
 - ♦ justify-content
 - ♦ align-items
 - ♦ align-content



Responsividade

Flex-direction – especifica a direção dos itens



Responsividade

Flex-direction

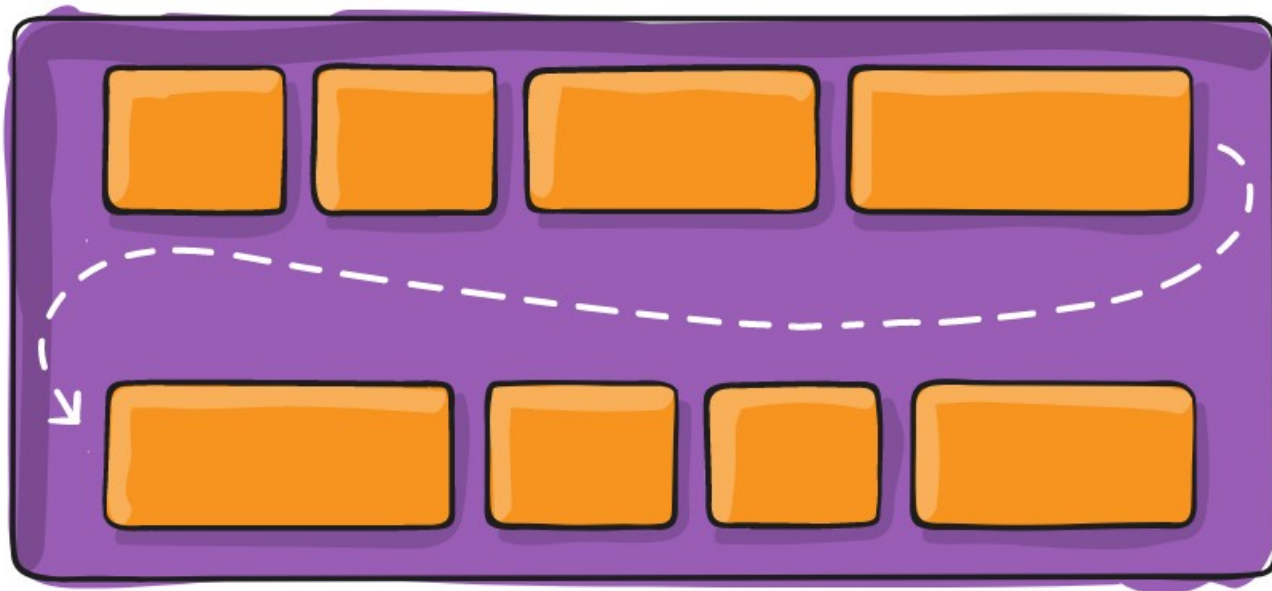
Value	Description
row	Default value. The flexible items are displayed horizontally, as a row
row-reverse	Same as row, but in reverse order
column	The flexible items are displayed vertically, as a column
column-reverse	Same as column, but in reverse order
initial	Sets this property to its default value. Read about <i>initial</i>
inherit	Inherits this property from its parent element. Read about <i>inherit</i>

Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_flex-direction&preval=row

Responsividade

Flex-wrap – especifica se os itens devem quebrar ou não



Por padrão, os *flex items* vão todos tentar se encaixar em uma só linha.

Responsividade

Flex-wrap

Value	Description
nowrap	Default value. Specifies that the flexible items will not wrap
wrap	Specifies that the flexible items will wrap if necessary
wrap-reverse	Specifies that the flexible items will wrap, if necessary, in reverse order
initial	Sets this property to its default value. Read about <i>initial</i>
inherit	Inherits this property from its parent element. Read about <i>inherit</i>

Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_flex-wrap&preval=wrap

Responsividade

Flex-flow

- É uma forma abreviada para flex-direction e flex-wrap.
- Valor padrão: row nowrap

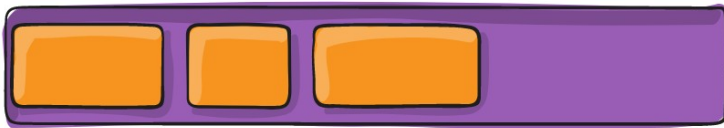
Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_flex-flow

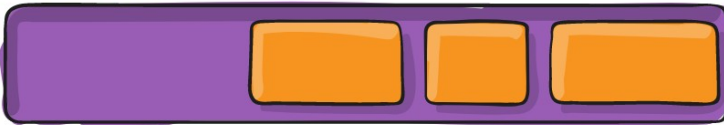
Responsividade

Justify-content – alinha os itens no contêiner

flex-start



flex-end



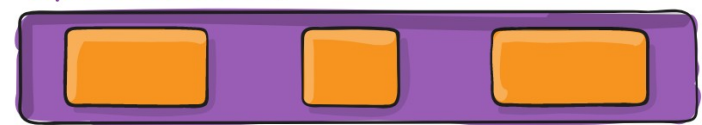
center



space-between



space-around



space-evenly



Responsividade

Justify-content

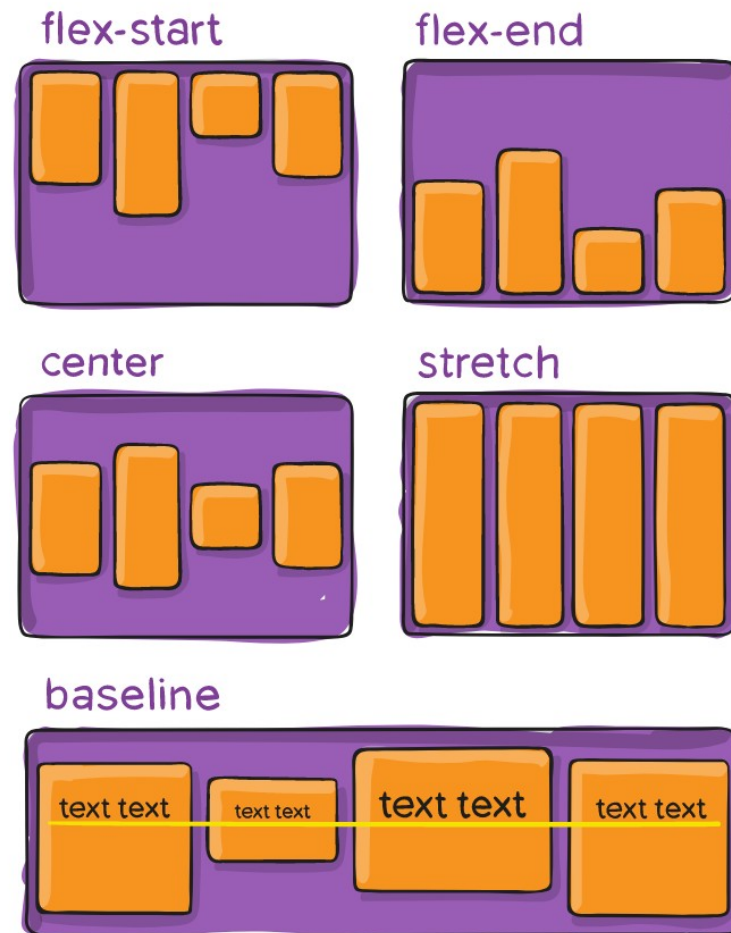
Value	Description
flex-start	Default value. Items are positioned at the beginning of the container
flex-end	Items are positioned at the end of the container
center	Items are positioned at the center of the container
space-between	Items are positioned with space between the lines
space-around	Items are positioned with space before, between, and after the lines
initial	Sets this property to its default value. Read about <i>initial</i>
inherit	Inherits this property from its parent element. Read about <i>inherit</i>

Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_justify-content&preval=flex-start

Responsividade

Align-items – especifica o alinhamento dos itens dentro do contêiner



Responsividade

Align-items

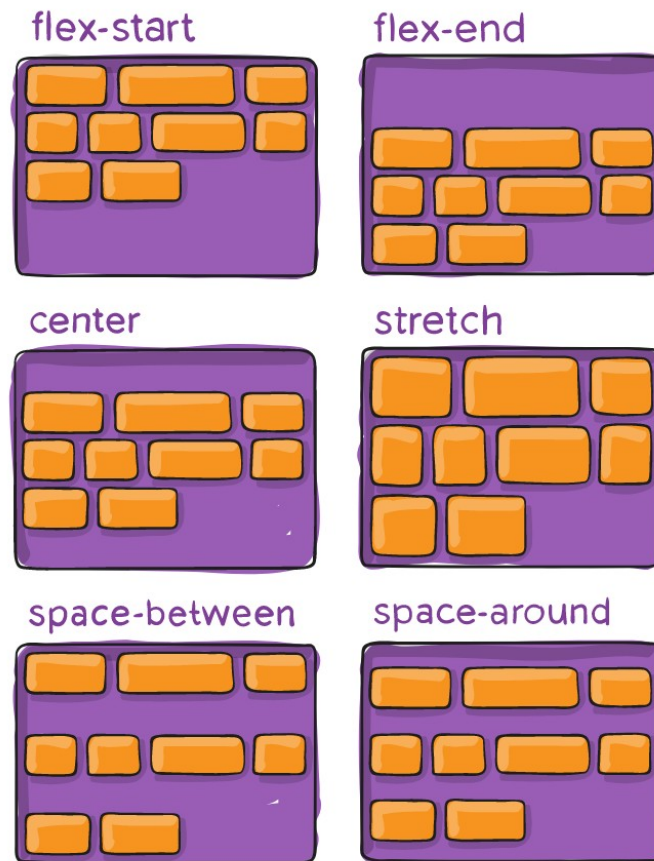
Value	Description
stretch	Default. Items are stretched to fit the container
center	Items are positioned at the center of the container
flex-start	Items are positioned at the beginning of the container
flex-end	Items are positioned at the end of the container
baseline	Items are positioned at the baseline of the container
initial	Sets this property to its default value. Read about initial
inherit	Inherits this property from its parent element. Read about inherit

Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_align-items&preval=stretch

Responsividade

Align-content – modifica o comportamento da propriedade flex-wrap. É similar à propriedade *align-items*, só que alinha as linhas.



Responsividade

Align-content

Value	Description
stretch	Default value. Lines stretch to take up the remaining space
center	Lines are packed toward the center of the flex container
flex-start	Lines are packed toward the start of the flex container
flex-end	Lines are packed toward the end of the flex container
space-between	Lines are evenly distributed in the flex container
space-around	Lines are evenly distributed in the flex container, with half-size spaces on either end
initial	Sets this property to its default value. Read about <i>initial</i>
inherit	Inherits this property from its parent element. Read about <i>inherit</i>

Exemplo:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_align-content&preval=stretch

Responsividade

```
<!DOCTYPE html>
<html>
<head>
<title>FlexBox</title>
<meta charset="UTF-8">
</head>
<style>
.box {
  display: flex;
}

div {
  padding: 10px;
  border: 2px red solid;
}
</style>
```

```
<body>

<h1>Teste de Flexbox</h1>

<div class="box">
  <div>A</div>
  <div>B</div>
  <div>C
    <br>C.1
    <br>C.2
    <br>C.3
  </div>
</div>

</body>
</html>
```

Responsividade

Teste de Flexbox

A	B	C
		C.1
		C.2
		C.3

Responsividade

```
.box {
  display: flex;
  flex-direction: row-reverse;
}
```

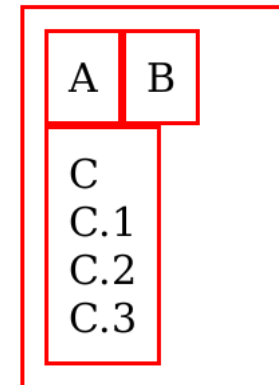
Teste de Flexbox



Responsividade

```
.box {
  display: flex;
  flex-wrap: wrap;
}
```

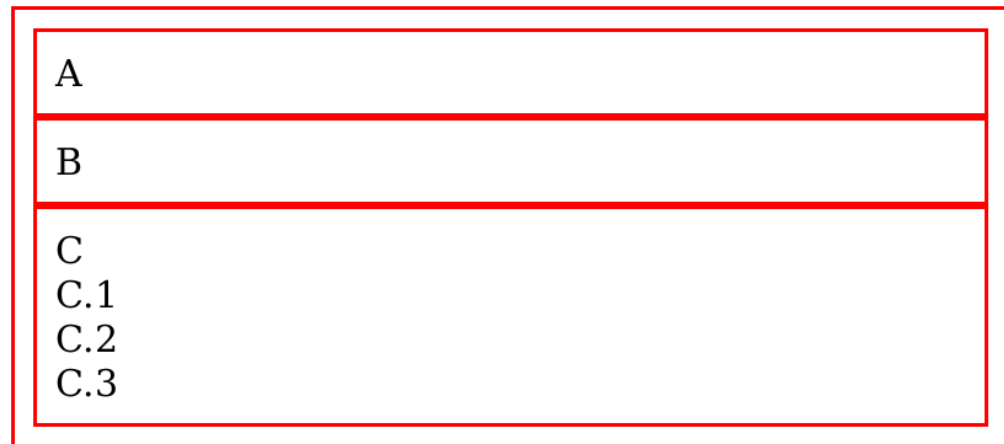
Teste de Flexbo



Responsividade

Teste de Flexbox

```
.box {  
  display: flex;  
  flex-direction: column;  
}
```



Responsividade

<https://www.alura.com.br/artigos/css-guia-do-flexbox>



Responsividade

<https://flexboxfroggy.com/>



Responsividade

- **Exemplo 1**

https://www.w3schools.com/css/tryit.asp?filename=trycss3_flexbox_responsive

- **Exemplo 2**

https://www.w3schools.com/css/tryit.asp?filename=trycss_mediaqueries_flex

- **Exemplo 3**

https://www.w3schools.com/css/tryit.asp?filename=tryresponsive_breakpoints

Responsividade

Veja também:

- https://www.w3schools.com/css/css3_flexbox.asp
- <https://codepen.io/estelle/pen/brDpB>
- <https://blog.hellojs.org/re-create-css-flexbox-media-queries-with-javascript-c192e4e16a63>
- https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Flexible_Box_Layout/Conceitos_Basicos_do_Flexbox