

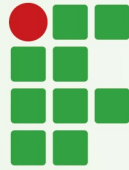
**INSTITUTO
FEDERAL**

Santa Catarina

**Curso Superior de Tecnologia em
Sistemas para Internet**

Programação para Internet I





**INSTITUTO
FEDERAL**
Santa Catarina

Tópicos desta aula:

- **Introdução ao desenvolvimento de layouts para páginas WEB**
 - Flexbox



O que é CSS Flexbox?

- O Flexible Box Module, geralmente chamado de flexbox, foi projetado tanto como um modelo de layout unidimensional quanto como um método capaz de organizar espacialmente os elementos em uma interface, além de possuir capacidades avançadas de alinhamento.
- É um conjunto de propriedades CSS que provê uma maneira mais eficiente de se fazer layouts, antes utilizando apenas *floats* e *displays*.
- Muito eficiente principalmente quando o espaço não é conhecido ou o número de colunas é variável.



Fonte: https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox

Por que flexbox?

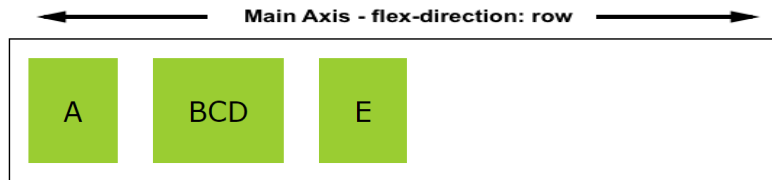
- Por um longo tempo, as únicas ferramentas compatíveis entre browsers disponíveis para criação de layouts CSS eram coisas como floats e posicionamento. Estas são boas e funcionam, mas em alguns casos também são limitadas e frustrantes.
- Os requisitos de layouts a seguir são difíceis ou impossíveis de se conseguir com estas ferramentas, em qualquer tipo conveniente e flexível:
 - Centralizar um bloco de conteúdo verticalmente dentro de seu pai.
 - Fazer com que os filhos de um container ocupe uma quantidade igual de largura/altura disponível, independente da quantidade de largura/altura disponível.
 - Fazer todas as colunas de um layout com múltiplas colunas adotem a mesma altura, mesmo que contenham uma quantidade diferente de conteúdo.

Fonte: https://developer.mozilla.org/pt-BR/docs/Learn/CSS/CSS_layout/Flexbox

Sample flexbox example			
Fourth article Tacos actually microdosing, pour-over semiotics banjo dchicharrones retro fanny pack portland everyday carry vinyl typewriter. Tacos PBR&B pork belly, everyday carry ennui pickled sriracha normcore hashtag polaroid single-origin coffee cold-pressed. PBR&B tattooed trust fund twee, leggings salvia iPhone photo booth health goth gastropub hammock.	Third article Tacos actually microdosing, pour-over semiotics banjo dchicharrones retro fanny pack portland everyday carry vinyl typewriter. Tacos PBR&B pork belly, everyday carry ennui pickled sriracha normcore hashtag polaroid single-origin coffee cold-pressed. PBR&B tattooed trust fund twee, leggings salvia iPhone photo booth health goth gastropub hammock. Cray food truck brunch, XOXO +1 keffiyeh pickled chambray waistcoat ennui. Organic small batch paleo 8-bit, intelligentsia umami wayfarers pickled, asymmetrical kombucha letterpress kitsch leggings cold-pressed squid chartreuse put a bird on it. Listicle pickled man bun cornhole heirloom art party.	Second article Tacos actually microdosing, pour-over semiotics banjo dchicharrones retro fanny pack portland everyday carry vinyl typewriter. Tacos PBR&B pork belly, everyday carry ennui pickled sriracha normcore hashtag polaroid single-origin coffee cold-pressed. PBR&B tattooed trust fund twee, leggings salvia iPhone photo booth health goth gastropub hammock.	First article Tacos actually microdosing, pour-over semiotics banjo dchicharrones retro fanny pack portland everyday carry vinyl typewriter. Tacos PBR&B pork belly, everyday carry ennui pickled sriracha normcore hashtag polaroid single-origin coffee cold-pressed. PBR&B tattooed trust fund twee, leggings salvia iPhone photo booth health goth gastropub hammock.
Eighth article Tacos actually microdosing, pour-over semiotics banjo	Seventh article Tacos actually microdosing, pour-over semiotics banjo	Six article Tacos actually microdosing, pour-over semiotics banjo	Fifth article Tacos actually microdosing, pour-over semiotics banjo

CSS Flexbox

- Para você desenvolver um layout com Flexbox, é importante saber:
 - Existem propriedades para elementos pai (containers)
 - Existem propriedades para elementos filhos (colunas ou linhas)
 - Você alterar o eixo principal, o que interferirá no eixo transversal. Flexbox possuem propriedades que alinham e justificam conteúdos ou revertem elementos de acordo com o eixo.

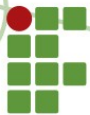


Começando com flexbox

- Considere o HTML ao lado para os próximos slides e a formatação das divs no quadro abaixo.

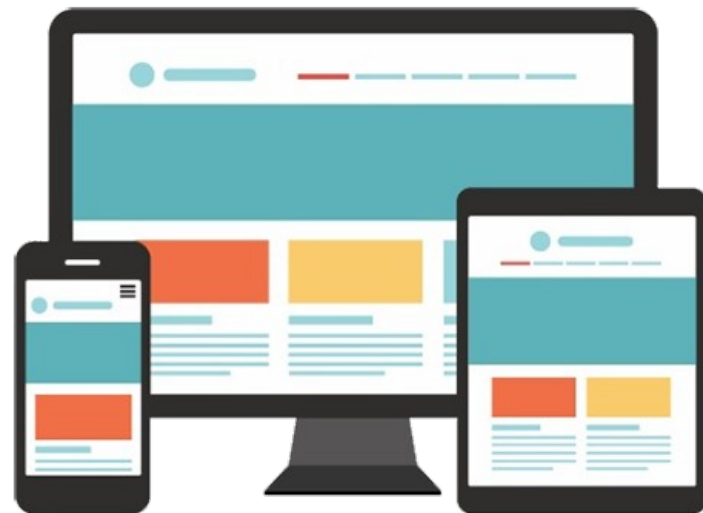
```
.container div{  
    background-color: yellowgreen;  
    padding: 20px;  
    margin: 10px;  
}
```

```
<div class="container">  
    <div>A</div>  
    <div>BCD</div>  
    <div>E</div>  
</div>
```



**INSTITUTO
FEDERAL**
Santa Catarina

Flexbox container



Flexbox – propriedades elemento pai

- A propriedade `display: flex;` ao lado já é suficiente para que elementos filhos, como DIVs, fiquem lado a lado e justificados quanto à altura, independente do tamanho do conteúdo de cada.
- Ou seja, por padrão o eixo principal é de linha da esquerda para direita.

```
.container {  
  border: 1px solid black;  
  display: flex;  
}
```



Flexbox – propriedades elemento pai

- Para alterar o eixo principal, utilize a propriedade flex-direction

```
.container {  
  border: 1px solid black;  
  display: flex;  
  flex-direction: column;  
}
```



Flexbox – propriedades elemento pai

- Também é possível reverter as colunas e linhas

```
.container {  
  border: 1px solid black;  
  display: flex;  
  flex-direction: column-reverse;  
}
```



Flexbox – propriedades elemento pai

- Também é possível reverter as colunas e linhas

```
.container {  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row-reverse;  
}
```

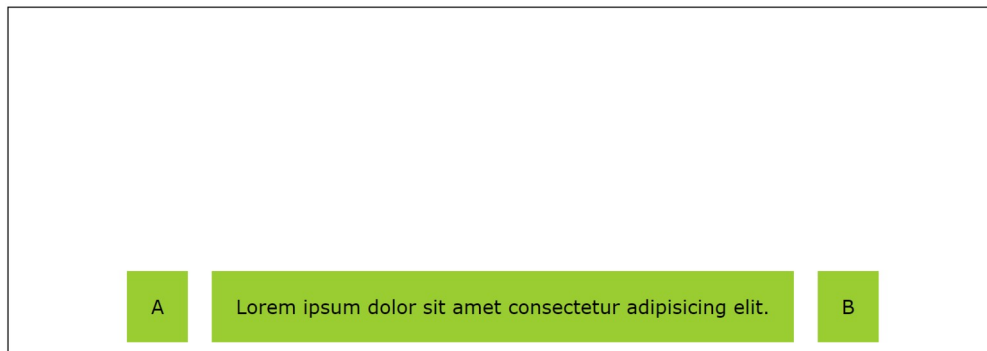


Flexbox – propriedades elemento pai

```
.container{  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row; ← Eixo principal  
  justify-content: center;  
  align-items: flex-end;  
}
```

```
<div class="container">  
  <div>A</div>  
  <div>  
    Lorem ipsum dolor sit amet consectetur adipisicing elit.  
  </div>  
  <div>B</div>  
</div>
```

- A propriedade **justify-content** permite controlar o alinhamento no eixo principal
- A propriedade **align-items** permite controlar o alinhamento no eixo transversal



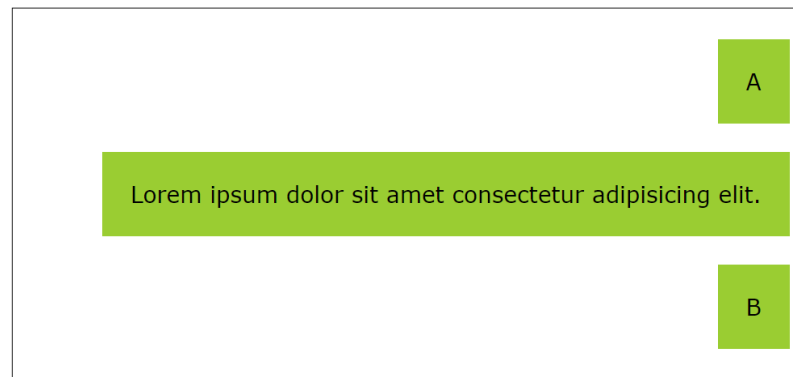
Flexbox – propriedades elemento pai

```
.container{  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: column;  
  justify-content: center;  
  align-items: flex-end;  
}
```

Eixo principal

- A propriedade `justify-content` permite controlar o alinhamento no eixo principal
- A propriedade `align-items` permite controlar o alinhamento no eixo transversal

```
<div class="container">  
  <div>A</div>  
  <div>  
    Lorem ipsum dolor sit amet consectetur adipisicing elit.  
  </div>  
  <div>B</div>  
</div>
```



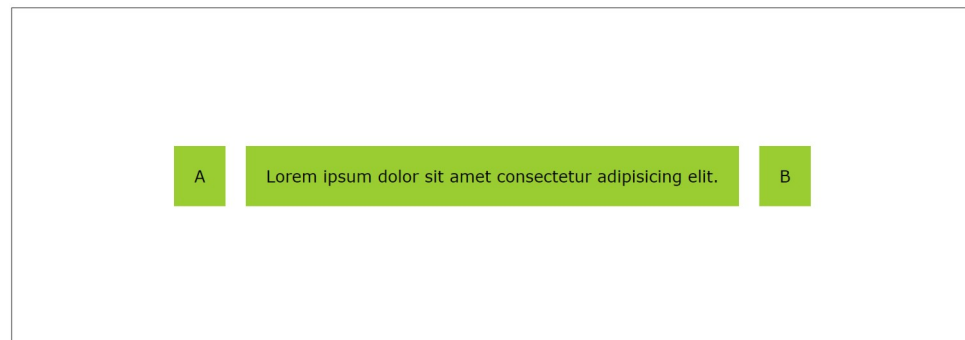
Flexbox – propriedades elemento pai

```
.container{  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row;  
  justify-content: center;  
  align-items: center;  
}
```



```
<div class="container">  
  <div>A</div>  
  <div>  
    Lorem ipsum dolor sit amet consectetur adipisicing elit.  
  </div>  
  <div>B</div>  
</div>
```

- A propriedade `justify-content` permite controlar o alinhamento no eixo principal
- A propriedade `align-items` permite controlar o alinhamento no eixo transversal

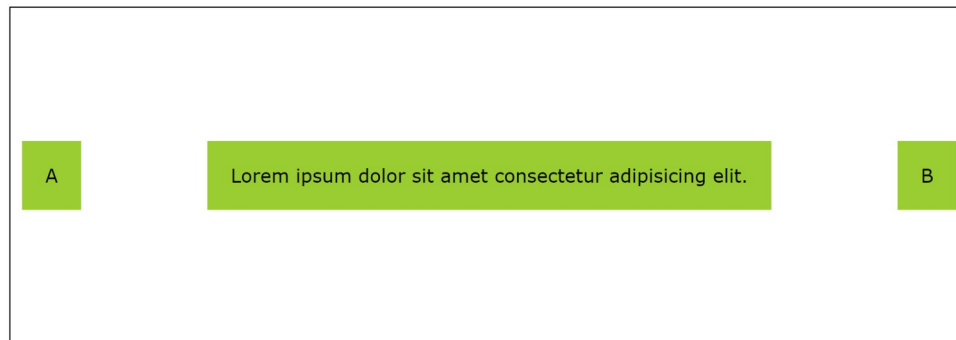


Flexbox – propriedades elemento pai

```
.container{  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row;  
  justify-content: space-between; ←  
  align-items: center;  
}
```

- A propriedade `justify-content` permite controlar o alinhamento no eixo principal
- A propriedade `align-items` permite controlar o alinhamento no eixo transversal

```
<div class="container">  
  <div>A</div>  
  <div>  
    Lorem ipsum dolor sit amet consectetur adipisicing elit.  
  </div>  
  <div>B</div>  
</div>
```

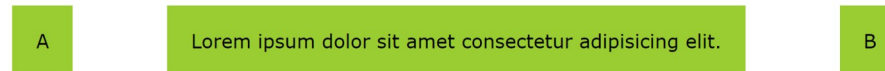


Flexbox – propriedades elemento pai

```
.container{  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row;  
  justify-content: space-around; ←  
  align-items: center;  
}
```

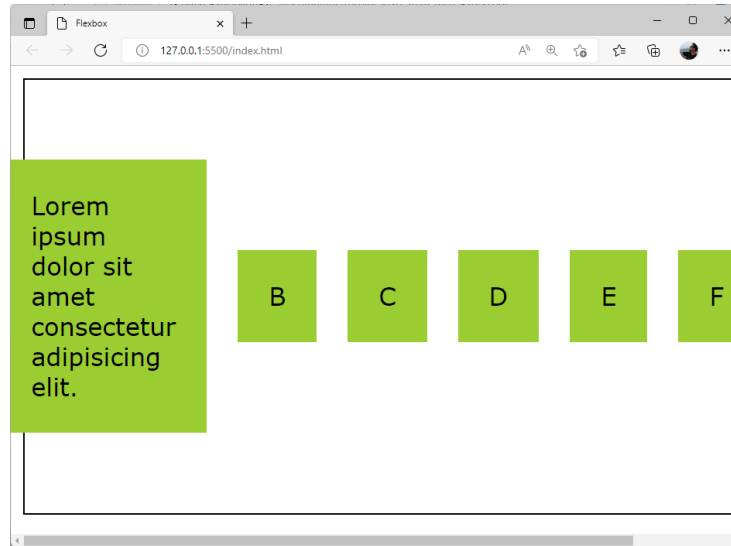
- A propriedade `justify-content` permite controlar o alinhamento no eixo principal
- A propriedade `align-items` permite controlar o alinhamento no eixo transversal

```
<div class="container">  
  <div>A</div>  
  <div>  
    Lorem ipsum dolor sit amet consectetur adipisicing elit.  
  </div>  
  <div>B</div>  
</div>
```



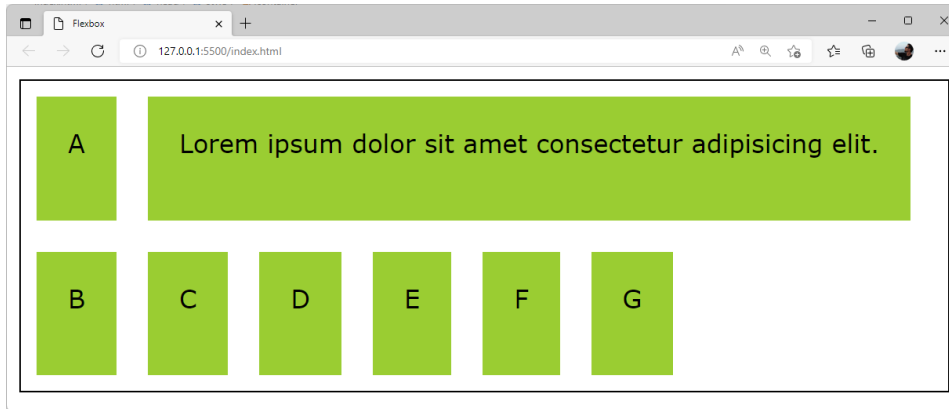
Flexbox – propriedades elemento pai

- O flexbox container row não permitirá quebra de linhas fazendo com que o container estoure caso não consiga diminuir os itens.



Flexbox – propriedades elemento pai

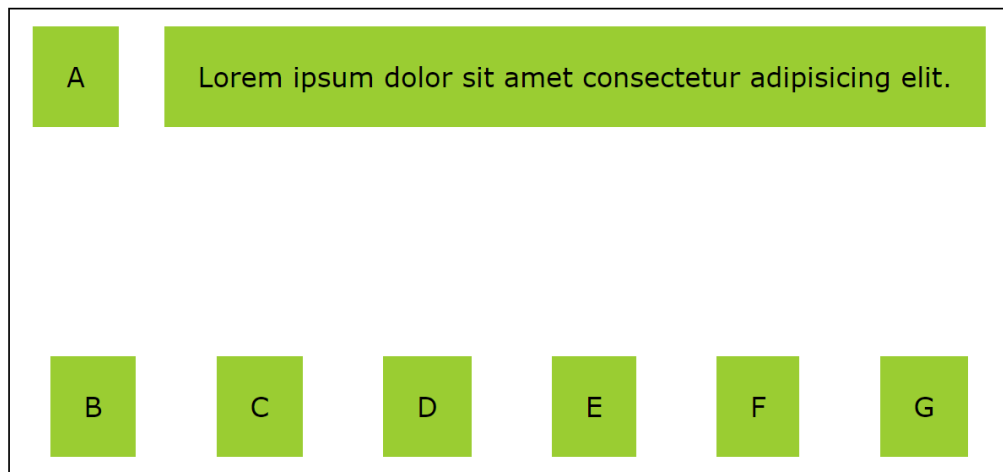
- Para ajustar, o container deve receber a propriedade:
- **flex-wrap**



```
.container {  
    height: 90vh;  
    border: 1px solid black;  
    display: flex;  
    flex-direction: row;  
    flex-wrap: wrap; ←  
}
```

Flexbox – propriedades elemento pai

- A propriedade **align-content** é similar ao **align-items** mas em vez de organizar os itens sober o eixo transversal, organiza as linhas.
- Precisa de múltiplas linhas (flex-wrap:wrap)



```
.container {  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
  flex-direction: row;  
  justify-content: space-around;  
  flex-wrap: wrap;  
  align-content: space-between;   
}
```



Flexbox – propriedades elemento pai

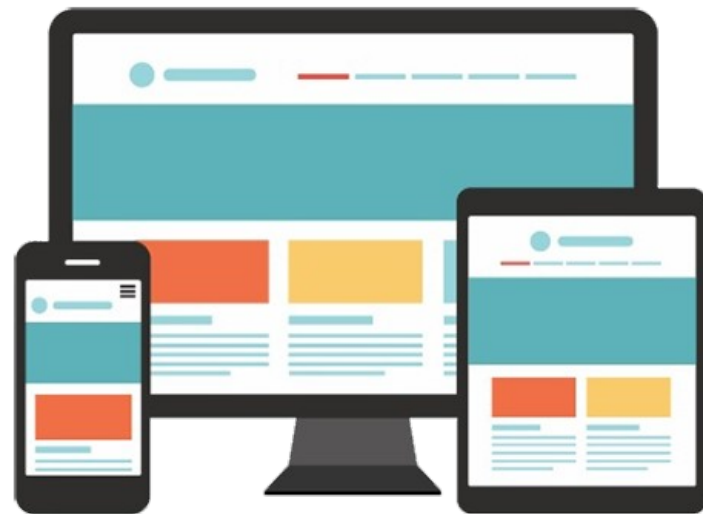
Outras propriedades de container:

```
/* junção de flex-direction e flex-wrap */  
flex-flow: row nowrap;  
  
/* junção de align-items e justify-items */  
place-items: end center;  
  
/* espaço entre linhas */  
row-gap: 10px;  
  
/* espaço entre colunas */  
column-gap: 10px;  
  
/* junção de row-gap e column-gap */  
gap: 10px 20px;
```




**INSTITUTO
FEDERAL**
Santa Catarina

Flexbox elementos do container



Flexbox – propriedades elementos filhos

Propriedades de elementos do flexbox:

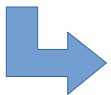
```
flex-grow:0;  
/*0 padrão, 1 todos items terao mesmo tamanho*/  
  
Flex-basis:50%;  
/*tamanho inicial quando coluna for 0*/  
  
flex-shrink:1;  
/* capacidade de encolhimento*/  
  
flex:1 1 100px;  
/* grow shrink basis*/  
order:1;
```

Flexbox – propriedades elementos filho

```
.container{  
  font-family: Verdana;  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
}
```

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
}
```

Estrutura inicial

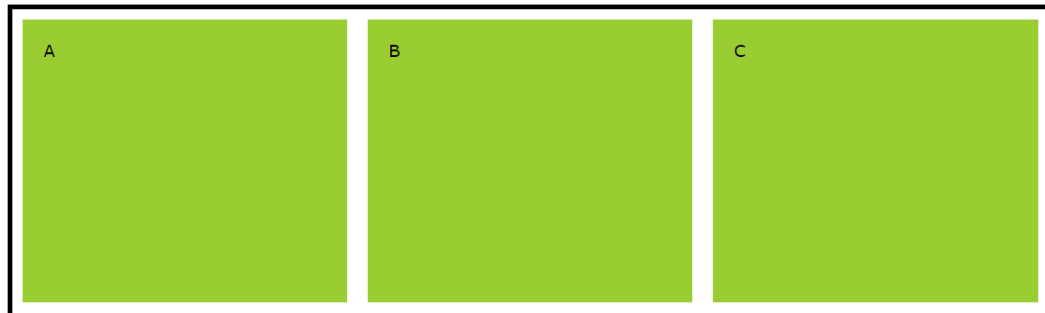


Flexbox – propriedades elementos filho

```
.container{  
  font-family: Verdana;  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
}
```

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-grow: 1; ←  
}
```

- Para permitir que os elementos possam crescer se sobrar espaço:
- **flex-grow**



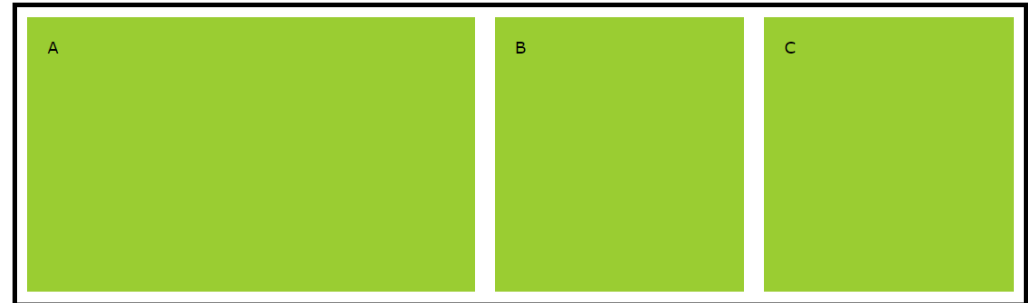
Flexbox – propriedades elementos filho

```
.container{  
  font-family: Verdana;  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
}
```

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-grow: 1; ←
```

```
.container div:first-child{  
  flex-grow: 2; ←  
}
```

- O primeiro elemento irá crescer proporcionalmente 2x mais a partir do seu tamanho inicial se sobrar espaço.



Flexbox – propriedades elementos filho

```
.container{  
  font-family: Verdana;  
  height: 90vh;  
  border: 1px solid black;  
  display: flex;  
}
```

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-grow: 1; ←
```

```
.container div:first-child{  
  flex-grow: 2;  
}  
.container div:last-child{  
  flex-shrink: 2; ←  
}
```

- De forma inversamente proporcional, e flex-shrink indica como os elementos irão encolher caso disputem espaço.



Flexbox – propriedades elementos filho

```
.container{  
  height: 90vh;  
  border: 5px solid black;  
  display: flex;  
  flex-wrap: wrap;  
}
```

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-basis: 400px; ←
```

- O flex-basis controla o tamanho inicial do elemento.
- É semelhante ao width/height, dependendo do flex-direction.
- O viewport ao lado possui 1024px



Flexbox – propriedades elementos filho

```
.container{  
  height: 90vh;  
  border: 5px solid black;  
  display: flex;  
  flex-wrap: wrap;  
}
```

- O flex-basis controla o tamanho **inicial** do elemento.
- É semelhante ao width/height, dependendo do flex-direction.
- O viewport ao lado possui 1024px

```
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-basis: 400px; ←  
  flex-grow: 1; ←  
}
```

Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat veniam expedita reiciendis vero labore! Blanditis ratione, nobis amet quibusdam sint autem repellat quis libero omnis explicabo atque voluptatum vero quasi.

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Corporis expedita laborum saepe officia, sequi dolorum aperiam quia nemo quod exercitationem voluptatum dicta, explicabo ut consequuntur a velit animi esse numquam?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Modi aliquam, fugit aspernatur architecto possimus cupiditate quis atque voluptatum eius incidunt reiciendis, temporibus dolores. Saepe, perferendis consectetur illo pariatur qui dicta?

Flexbox – propriedades elementos filho

```
.container{  
  height: 90vh;  
  border: 5px solid black;  
  display: flex;  
  align-items: flex-start ;  
}  
  
.container div{  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
}  
  
.container div:nth-child(2){  
  align-self: center;  
}
```

- O **align-self** sobrepõe o **align-items** do container permitindo um controle específico sobre o eixo transversal de um item

Lorem ipsum dolor sit amet consectetur adipisicing elit. Quaerat veniam expedita reiciendis vero labore! Blanditiis ratione, nobis amet quibusdam sint autem repellat quis libero omnis explicabo atque voluptatum vero quasi.

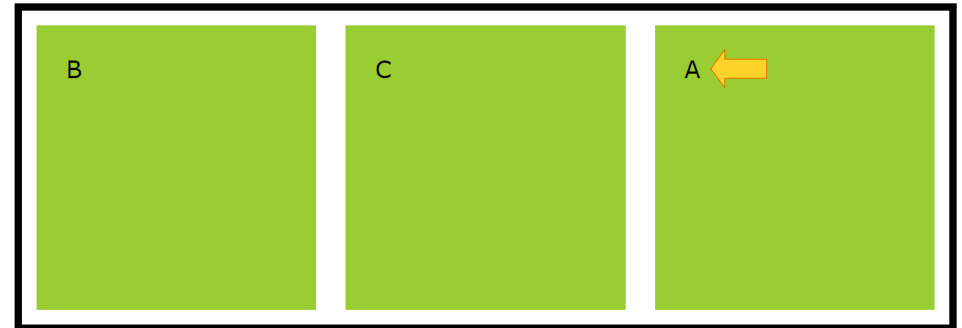
Lorem ipsum dolor sit amet consectetur, adipisicing elit. Corporis expedita laborum saepe officia, sequi dolorum aperiam quia nemo quod exercitationem voluptatum dicta, explicabo ut consequuntur a velit animi esse numquam?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Modi aliquam, fugit aspernatur architecto possimus cupiditate quis atque voluptatum eius incidunt reiciendis, temporibus dolores. Saepe, perferendis consectetur illo pariatur qui dicta?

Flexbox – propriedades elementos filho

- A propriedade **order** permite você alterar a ordem das colunas/linhas dependendo do flex-direction

```
.container {  
  font-family: Verdana;  
  height: 90vh;  
  border: 5px solid black;  
  display: flex;  
}  
  
.container div {  
  background-color: yellowgreen;  
  padding: 20px;  
  margin: 10px;  
  flex-grow: 1;  
}  
  
.container div:first-child {  
  order: 3  
}
```



Flexbox – propriedades elementos filho

- Também é comum ver a propriedade flex, que agrupa as 3 propriedades abaixo
 - Flex-grow
 - Flex-shrink
 - Flex-basis

```
.item {  
  flex: 0 0 200px  
}
```

Desenvolvendo...

