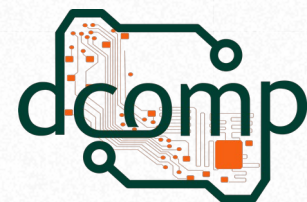




Universidade Federal do Espírito Santo  
Centro de Ciências Exatas, Naturais e da Saúde  
Departamento de Computação



# Fundamentos de Programação Web Flexbox

## **Fundamentos de Programação WEB**

Site: <http://www.jeiks.net/fundpweb>

E-mail: [jacson.silva@ufes.br](mailto:jacson.silva@ufes.br)

# Modo de Layout com Flexbox

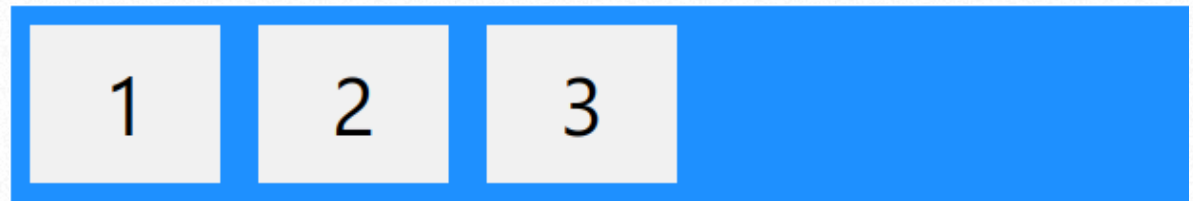
- Antes do módulo Flexbox Layout, havia quatro modos de layout:
  - block: para seções em uma página da web.
  - inline: para texto.
  - table: para dados de tabela bidimensionais.
  - positioned: para posição explícita de um elemento.
- Porém, o módulo flexível (flex):
  - Facilita o projeto de layout responsivo;
  - Não é necessário usar flutuação ou posicionamento.



# Itens do Flexbox

- Para começar a usar o flexbox, deve-se definir um *container*:

```
<div class="container">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
</div>
```



```
<style>  
  .container {  
    display: flex;  
    background-color: DodgerBlue;  
  }  
  .container > div {  
    background-color: #f1f1f1;  
    margin: 10px;  
    padding: 20px;  
    font-size: 30px;  
  }  
</style>
```

# Seletor CSS

Qual a diferença entre usar “**espaço**” e “**>**” ?

```
<div class="seletor">
  <section>
    <div>DIV dentro de seletor.
      Porém, filho de "section"
    </div>
  </section>
  <div>DIV dentro do seletor e
    também filho do seletor
  </div>
</div>
```

```
.seletor {
  width: 600px;
  margin: 0 auto;
  text-align: center;
}

.seletor div {
  background-color: #AFA;
}

.seletor > div {
  color: #00F;
  font-size: 20px;
}
```

# Propriedades do *container* Flex

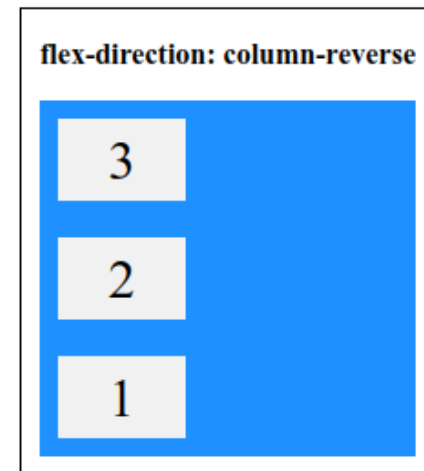
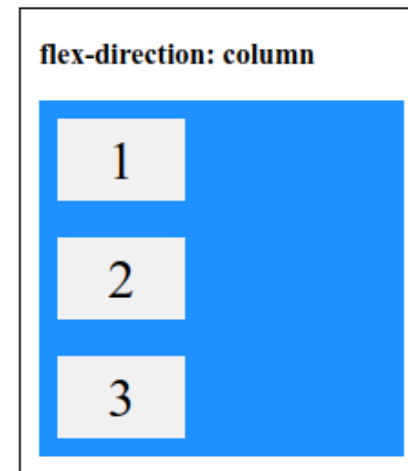
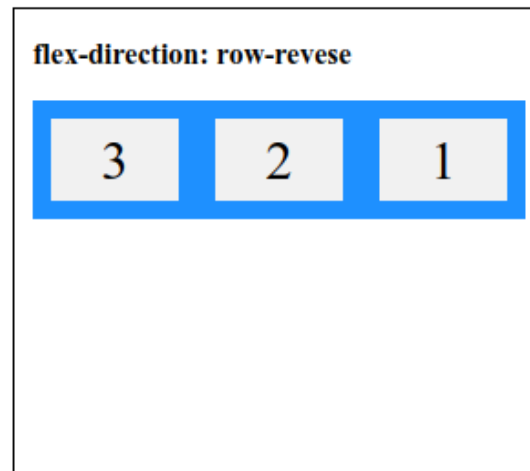
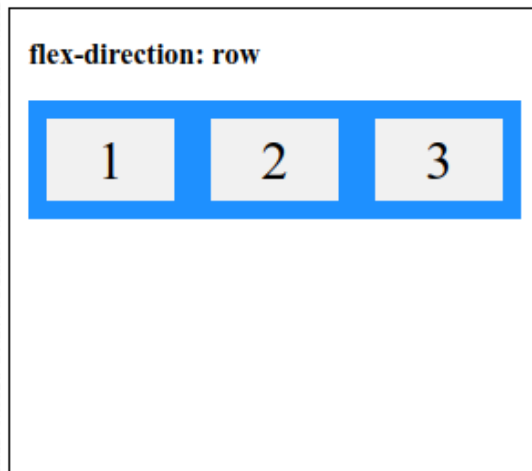
- Para tornar um *container* flexível, define-se sua propriedade *display* como *flex*:  

```
.container { display: flex; }
```
- Após isso, ele aceita as propriedades de um *container* flexível, que são:
  - flex-direction
  - flex-wrap
  - flex-flow
  - justify-content
  - align-items
  - align-content



# flex-direction

- Define a direção de seus itens:
  - row*: linha (padrão).
  - row-reverse*: linha ao contrário.
  - column*: coluna.
  - column-reverse*: coluna invertida.



# flex-wrap

- Define se os itens podem quebrar linha:
  - wrap*: quebra a linha.
  - nowrap*: não quebra a linha (padrão).
  - wrap-reverse*: quebra a linha, mas inverte a ordem.

flex-wrap: wrap

0	1	2	3
4	5	6	7
8	9		

flex-wrap: nowrap

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

flex-wrap: wrap-reverse

8	9		
4	5	6	7
0	1	2	3

# flex-flow

- Atalho para aplicar, juntas, as propriedades *flex-direction* e *flex-wrap*. Sintaxe:

*flex-flow*: [flex-direction] [flex-wrap];

- Exemplos:

*flex-flow*: row wrap;

*flex-flow*: row-reverse wrap-reverse;

*flex-flow*: row-reverse wrap;

**flex-flow: row wrap**

0	1	2	3
4	5	6	7
8	9		

**flex-flow: row-reverse wrap-reverse**

		9	8
7	6	5	4
3	2	1	0

**flex-flow: row-reverse wrap**

3	2	1	0
7	6	5	4
		9	8



# justify-content

- Utilizada para alinhar os itens na horizontal:

*center*: centro.

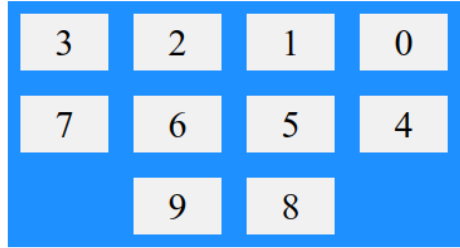
*flex-end* e *flex-start* (padrão): fim ou início flex.

*right* e *left*: direita ou esquerda.

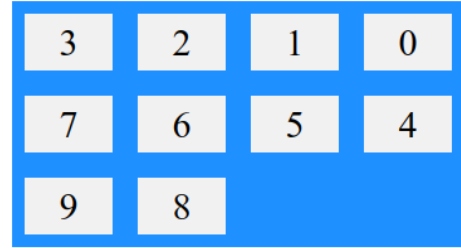
*space-around*: espaço ao redor dos itens.

*space-between*: espaço entre os itens.

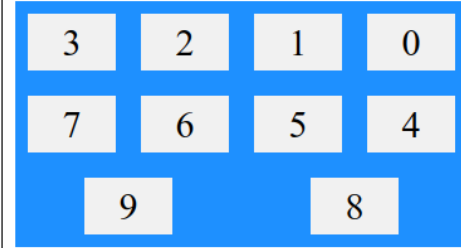
justify-content: center



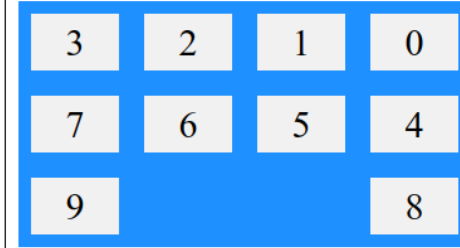
justify-content: flex-end (right)



justify-content: space-around

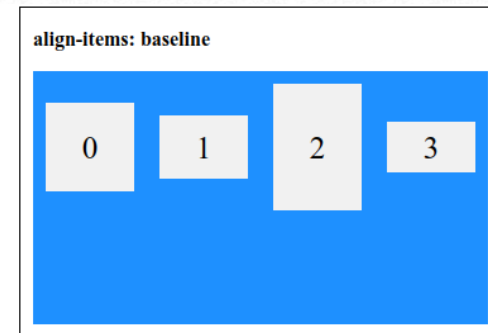
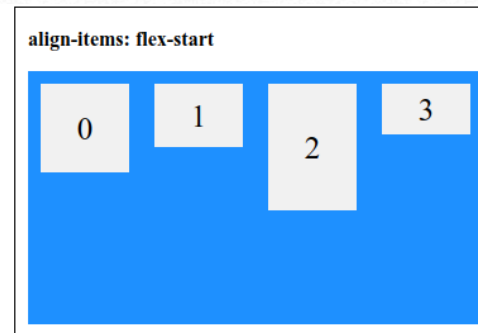
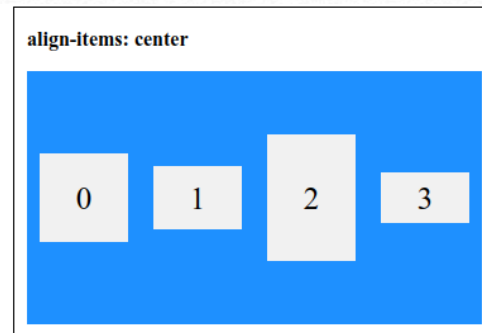
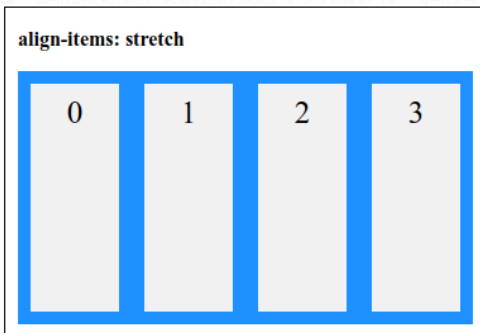


justify-content: space-between



# align-items

- Utilizada para alinhar os itens na vertical:
  - stretch*: estica para caber no *container* (padrão).
  - center*: no meio.
  - flex-start* e *flex-end*: no topo e no fim (embaixo).
  - baseline*: linha de base dos itens (meio).



# align-content

- Utilizada para alinhar as linhas:
  - stretch*: esticar (padrão).
  - center*: centralizar.
  - space-around*: espaço ao redor das linhas.
  - space-between*: espaço entre as linhas.
  - flex-start* e *flex-end*: início ou fim.

