

**INSTITUTO
FEDERAL**
Santa Catarina

Posicionando elementos com CSS

Programação Frontend I

Análise e Desenvolvimento de Sistemas

Prof. Adriano Lima

adriano.lima@ifsc.edu.br



INSTITUTO FEDERAL

Santa Catarina

Câmpus São José

PENSAMENTO
COMPUTACIONAL E
ALGORITMOS

Análise e Desenvolvimento
de Sistemas

cores

#111027

#277756

#16ABCD

#FFF4EC

#C74E23

fontes

Fira Sans Extra Condensed

Ubuntu

Roboto Mono



Seletores

podem ser

- elementos
- classes
- ids



Seletores

podem ser

- elementos
 - classes
 - ids
 - **pseudo-classes**
- selecionam elementos a partir de estados monitorados pelos navegador
 - não estão na marcação do documento

Seletores

podem ser

- elementos
- classes
- ids
- **pseudo-classes**

```
elemento:pseudo-classe {  
    propriedade: valor  
}
```

Seletores

podem ser

- elementos
- classes
- ids
- **pseudo-classes**

```
elemento:pseudo-classe {  
    propriedade: valor  
}
```

exemplos

:link (link não clicado)

:visited (link já clicado)

:focus (selecionado e pronto para entrada)

:hover (mouse sobre o elemento)

:active (elemento sendo clicado)



Seletores

podem ser

- elementos
 - classes
 - ids
 - pseudo-classes
 - **pseudo-elementos**
- selecionam elementos gerados pelos navegador
 - não estão na marcação do documento

Seletores

podem ser

- elementos
- classes
- ids
- pseudo-classes
- **pseudo-elementos**

```
elemento::pseudo-elemento {  
    propriedade: valor  
}
```


Seletores

podem ser

- elementos
- classes
- ids
- pseudo-classes
- **pseudo-elementos**

```
elemento::pseudo-elemento {  
    propriedade: valor  
}
```

exemplos

::first-line (primeira linha do elemento)
::first-letter (primeira letra do elemento)
::before (conteúdo gerado antes)
::after (conteúdo gerado antes)



Seletores

podem ser

- elementos
 - selecionam elementos com base em nomes ou valores de atributos
- classes
- ids
- pseudo-classes
- pseudo-elementos
- **atributos**

Seletores

podem ser

- elementos
- classes
- ids
- pseudo-classes
- pseudo-elementos
- **atributos**

```
elemento[atributo] {  
    propriedade: valor  
}
```

Seletores

podem ser

- elementos
- classes
- ids
- pseudo-classes
- pseudo-elementos
- **atributos**

(valor exato)

```
elemento[atributo="valor"] {  
    propriedade: valor  
}
```

(valor parcial)

```
elemento[atributo~="valor"] {  
    propriedade: valor  
}
```

(valor separado por hífen)

```
elemento[atributo|="valor"] {  
    propriedade: valor  
}
```



Seletores

podem ser

- elementos
- classes
- ids
- pseudo-classes
- pseudo-elementos
- **atributos**

(primeira parte do valor)

```
elemento[atributo^="valor"] {  
    propriedade: valor  
}
```

(última parte do valor)

```
elemento[atributo$="valor"] {  
    propriedade: valor  
}
```

(qualquer parte do valor)

```
elemento[atributo*="valor"] {  
    propriedade: valor  
}
```



Float



- forma de posicionamento de elementos
- move o elemento para o lado (esquerda/direita)
- permite que o texto se envolva ao redor do elemento flutuante

Float

float

valores left | right | none

padrão none

se aplica a todos os elementos

herança não



Float

sem alteração

```
<p>Lorem ipsum ... </p>
```



Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

Float

flutuando

```
<p>Lorem ipsum ... </p>
```

```
img {  
    float: right;  
}
```

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.



Float

flutuando com margem

```
<p>Lorem ipsum ... </p>
```

```
img {  
  float: right;  
  margin: 1em;  
}
```

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.



Float

elementos em linha

```
span {  
  width: 20%;  
  float: right;  
  padding: .5em;  
  margin: 1em;  
}
```

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

AVISO: Lorem ipsum dolor sit amet consectetur adipisicing elit.



Float

elementos em linha

- sempre forneça uma largura para elementos de texto flutuantes
- elementos de texto flutuantes se comportam como elementos de bloco
- as margens dos elementos flutuantes não colapsam

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

AVISO: Lorem ipsum dolor sit amet consectetur adipisicing elit.

Float

elementos em bloco

- sempre forneça uma largura para elementos em bloco flutuantes
- os elementos não flutuam mais alto do que sua referência
- elementos não flutuantes mantêm o fluxo normal

AVISO: Lorem ipsum dolor sit amet consectetur adipisicing elit. Sapiente voluptatibus, quia laudantium dolorem assumenda deleniti, autem officiis nesciunt fuga voluptatum harum fugit sequi nihil. Magnam nemo praesentium dolorem corrupti? Tenetur?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

Título

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Aliquid doloribus repudiandae aspernatur porro id. Architecto neque tenetur eum quos nam ducimus, libero error, repellendus praesentium aperiam, quis harum cumque officia?

Float

clear

valores left | right | both | none

padrão none

se aplica a apenas elementos de bloco

herança não



Float

limpando a flutuação

```
#outraSecao {  
  clear: both;  
}
```

AVISO: Lorem ipsum dolor sit amet consectetur adipisicing elit. Sapiente voluptatibus, quia laudantium dolorem assumenda deleniti, autem officiis nesciunt fuga voluptatum harum fugit sequi nihil. Magnam nemo praesentium dolorem corrupti? Tenetur?

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

Título

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Aliquid doloribus repudiandae aspernatur porro id. Architecto neque tenetur eum quos nam ducimus, libero error, repellendus praesentium aperiam, quis harum cumque officia?



Float

shape-outside

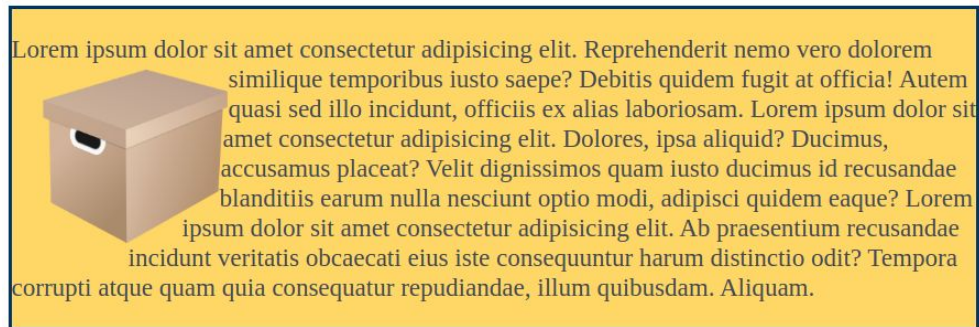
valores	<code>none circle() ellipse() polygon() url() [margin-box padding-box content-box]</code>
padrão	<code>none</code>
se aplica a	apenas elementos de bloco
herança	não



Float

contornando o elemento flutuante

```
img {  
  float: right;  
  margin: 1em;  
  shape-outside: url('img/caixa.png');  
}
```



Float

contornando o elemento flutuante

```
img {  
  float: right;  
  margin: 1em;  
  shape-outside: circle(25%);  
}
```



Lorem ipsum dolor sit amet consectetur adipisicing elit. Reprehenderit nemo vero dolorem similique temporibus iusto saepe? Debitis quidem fugit at officia! Autem quasi sed illo incidunt, officiis ex alias laboriosam. Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolores, ipsa aliquid? Ducimus, accusamus placeat? Velit dignissimos quam iusto ducimus id recusandae blanditiis earum nulla nesciunt optio modi, adipisci quidem eaque? Lorem ipsum dolor sit amet consectetur adipisicing elit. Ab praesentium recusandae incidunt veritatis obcaecati eius iste consequuntur harum distinctio odit? Tempora corrupti atque quam quia consequatur repudiandae, illum quibusdam. Aliquam.

Flexbox



Flexbox

- maior controle sobre a organização dos itens ao longo de um eixo
- flexibilidade para lidar com ampla variedade de tamanhos de tela
- capacidade de fazer com que todos os itens vizinhos tenham a mesma altura
- fácil centralização horizontal e vertical
- capacidade de alterar a ordem de exibição dos itens



Flexbox

display

valores

inline | block | run-in | flex | grid | flow |
flow-root | list-item | table | table-row-group |
table-header-group | table-footer-group | table-row |
table-cell | table-column-group | table-column |
table-caption | ruby | ruby-base | ruby-text |
ruby-base-container | ruby-text-container |
inline-block | inline-table | inline-flex |
inline-grid | contents | none

padrão

inline

se aplica a

todos os elementos

herança

sim



Flexbox

display

valores

`inline | block | run-in | flex | grid | flow |
flow-root | list-item | table | table-row-group |
table-header-group | table-footer-group | table-row |
table-cell | table-column-group | table-column |
table-caption | ruby | ruby-base | ruby-text |
ruby-base-container | ruby-text-container |
inline-block | inline-table | inline-flex |
inline-grid | contents | none`

padrão

`inline`

se aplica a

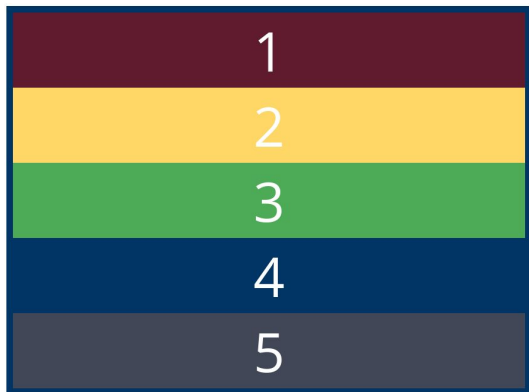
todos os elementos

herança

sim



Flexbox



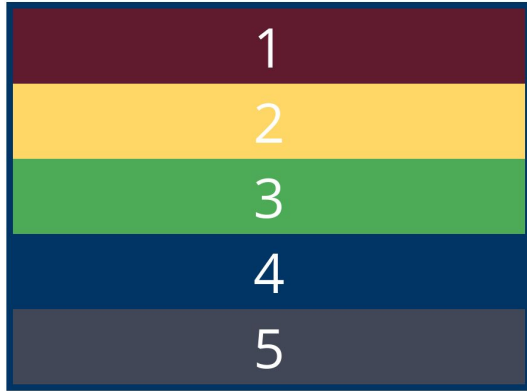
```
<section class="container-flex">  
  <div class="vermelho">1</div>  
  <div class="laranja">2</div>  
  <div class="verde">3</div>  
  <div class="azul">4</div>  
  <div class="roxo">5</div>  
</section>
```

Flexbox

```
.container-flex {  
  display: flex;  
}
```



Flexbox



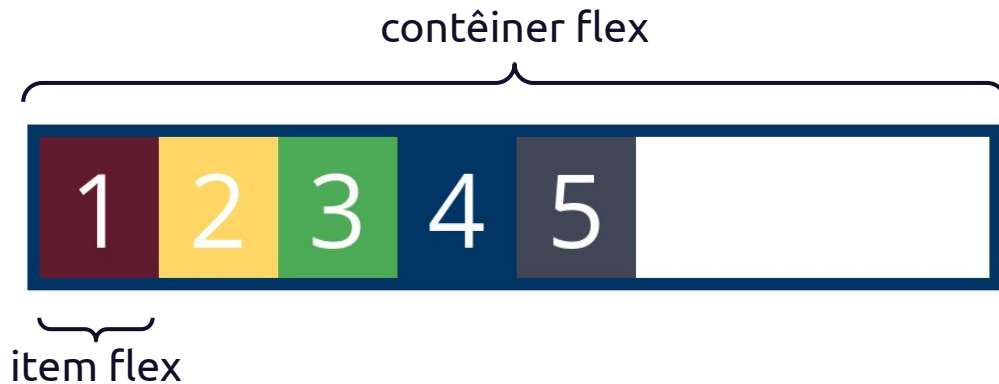
padrão



flex

Flexbox

```
.container-flex {  
  display: flex;  
}
```



Flexbox

flex-direction

valores	row column row-reverse column-reverse
padrão	row
se aplica a	contêineres flexíveis (<i>flex</i>)
herança	não



Flexbox

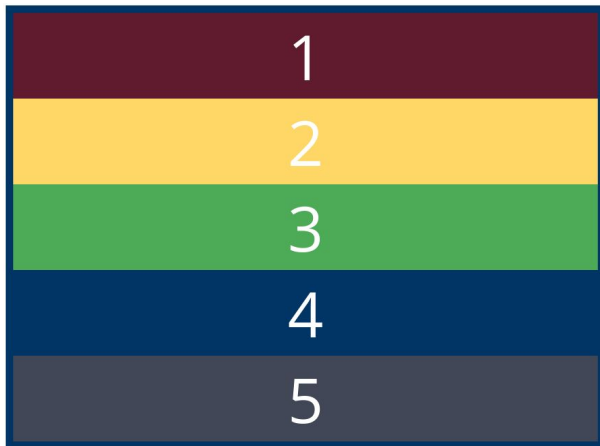
`flex-direction: row; (padrão)`



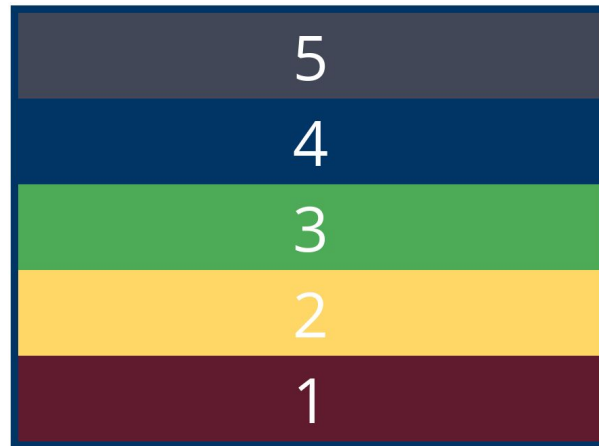
`flex-direction: row-reverse;`



`flex-direction: column;`

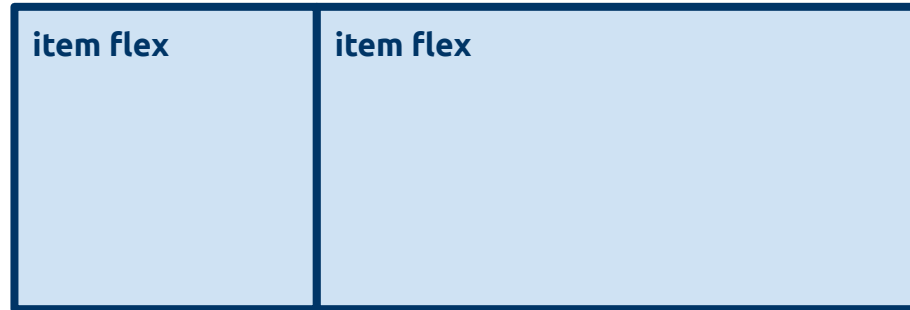


`flex-direction: column-reverse;`



Flexbox

`flex-direction: row;`



contêiner flex

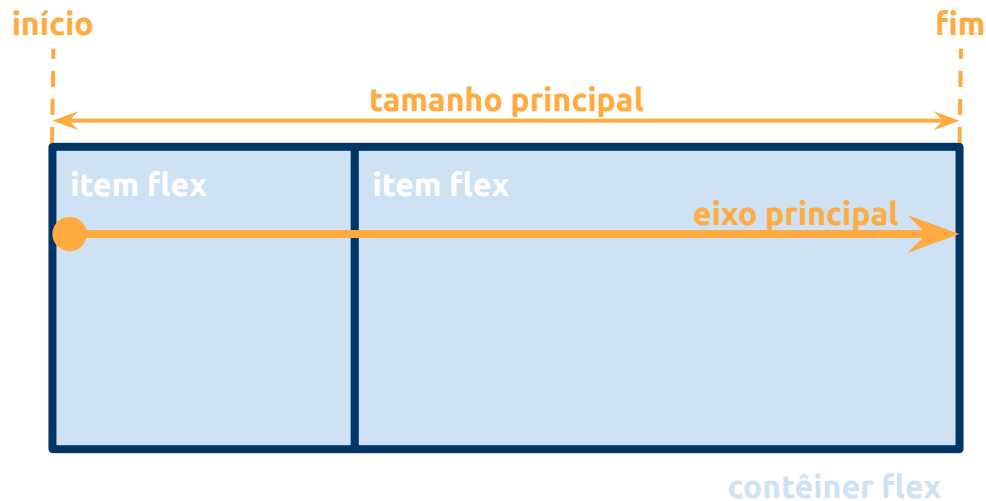
Flexbox

`flex-direction: row;`



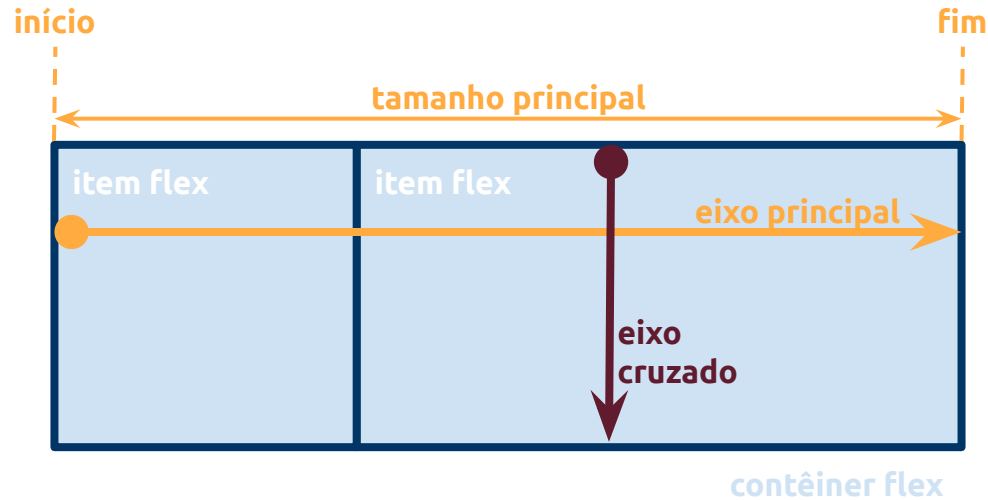
Flexbox

`flex-direction: row;`



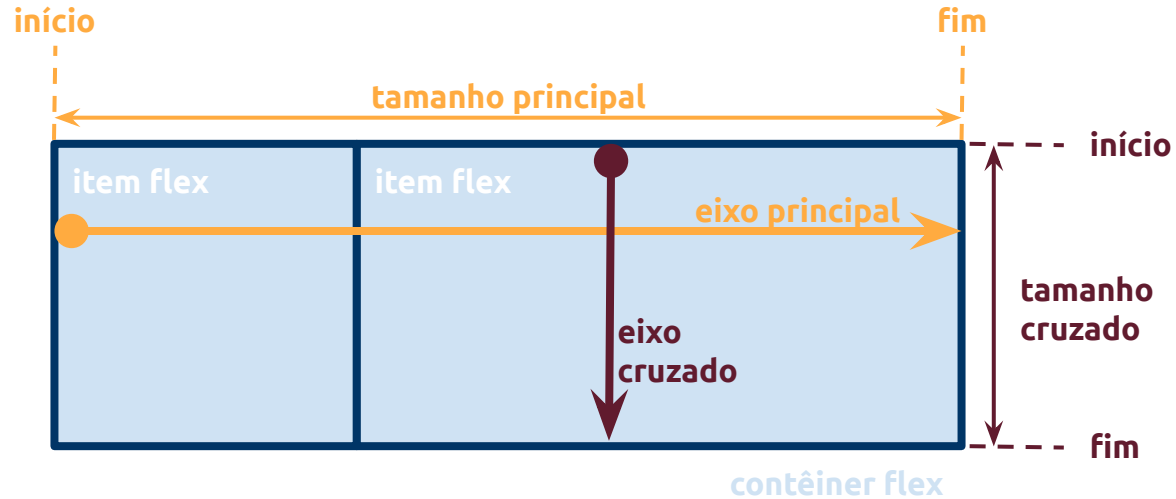
Flexbox

`flex-direction: row;`



Flexbox

`flex-direction: row;`



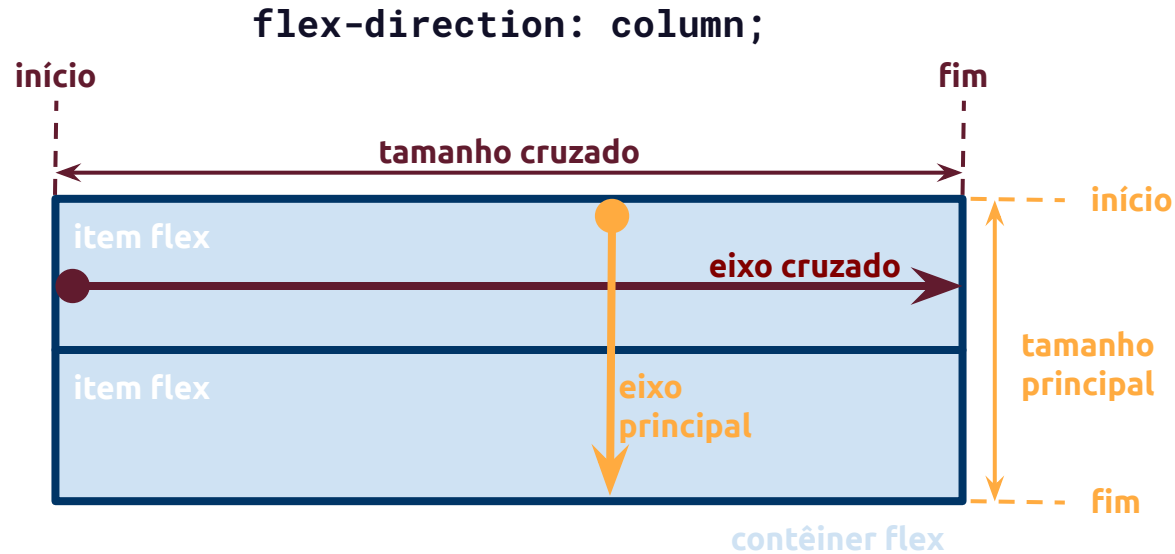
Flexbox

`flex-direction: column;`



contêiner flex

Flexbox



Flexbox

```
flex-direction: row;
```



Flexbox

flex-wrap

valores nowrap | wrap | wrap-reverse

padrão nowrap

se aplica a contêineres flexíveis (*flex*)

herança não



Flexbox

`flex-direction: row;`

`flex-wrap: wrap;`

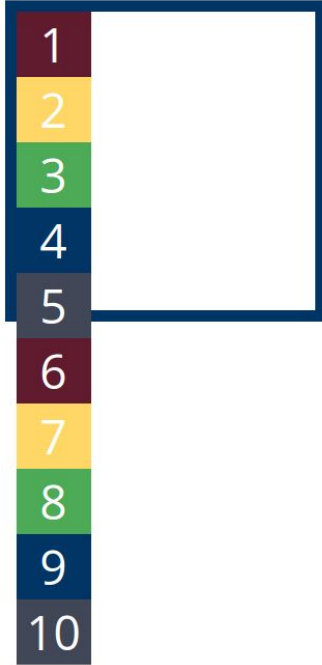
1	2	3	4
5	6	7	8
9	10		

`flex-wrap: wrap-reverse;`

9	10		
5	6	7	8
1	2	3	4

Flexbox

`flex-wrap: nowrap; (padrão)`



`flex-direction: column;`

`flex-wrap: wrap;`



`flex-wrap: wrap-reverse;`



Flexbox

flex-flow

valores *flex-direction flex-wrap*

padrão row nowrap

se aplica a contêineres flexíveis (*flex*)

herança não



Flexbox

justify-content

valores	<code>flex-start</code> <code>flex-end</code> <code>center</code> <code>space-between</code> <code>space-around</code>
padrão	<code>flex-start</code>
se aplica a	contêineres flexíveis (<i>flex</i>)
herança	não

■ alinhamento no eixo principal



Flexbox

`justify-content: flex-start; (padrão)`



`flex-direction: row;`

`justify-content: flex-end;`



`justify-content: center;`



`justify-content: space-between;`



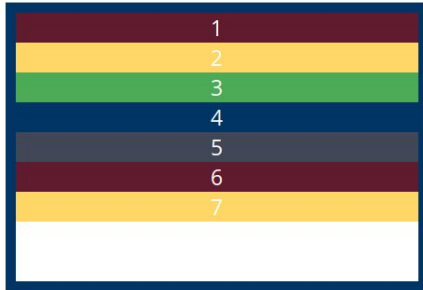
`justify-content: space-around;`



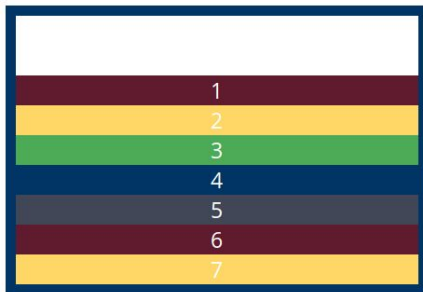
Flexbox

(padrão)

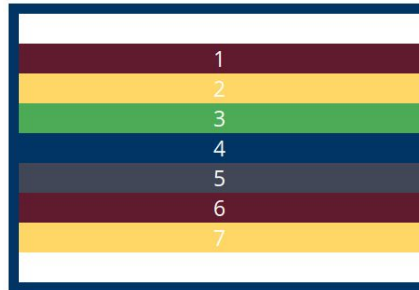
`justify-content: flex-start;`



`justify-content: flex-end;`

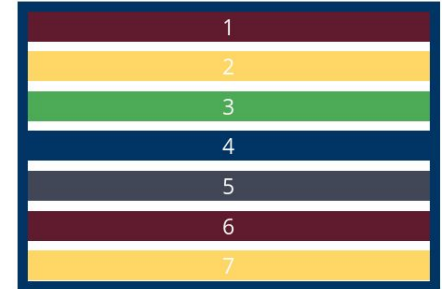


`justify-content: center;`

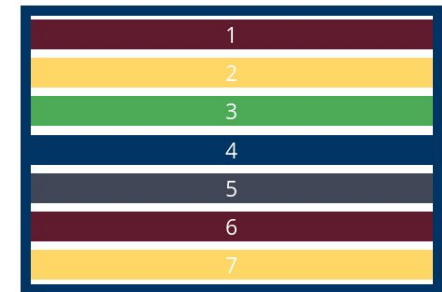


`flex-direction: column;`

`justify-content: space-between;`



`justify-content: space-around;`



Flexbox

align-items

valores	<code>flex-start</code> <code>flex-end</code> <code>center</code> <code>baseline</code> <code>stretch</code>
padrão	<code>stretch</code>
se aplica a	contêineres flexíveis (<i>flex</i>)
herança	não

■ alinhamento no eixo cruzado



Flexbox

`flex-direction: row;`

`align-items: flex-start;`



`align-items: flex-end;`



`align-items: center;`



`align-items: stretch;` (*padrão*)



os itens são alinhados para
que as linhas de base do texto
fiquem alinhadas

`align-items: baseline;`



Flexbox

align-self

valores	flex-start flex-end center baseline stretch
padrão	stretch
se aplica a	itens flexíveis (<i>flex</i>)
herança	não

- alinhamento individual no eixo cruzado

Flexbox

`align-self: flex-end;`



Flexbox

align-content

valores	flex-start flex-end center space-around space-between stretch
padrão	stretch
se aplica a	contêineres flexíveis (<i>flex</i>) multilinhas
herança	não

■ alinhamento de linhas no eixo cruzado



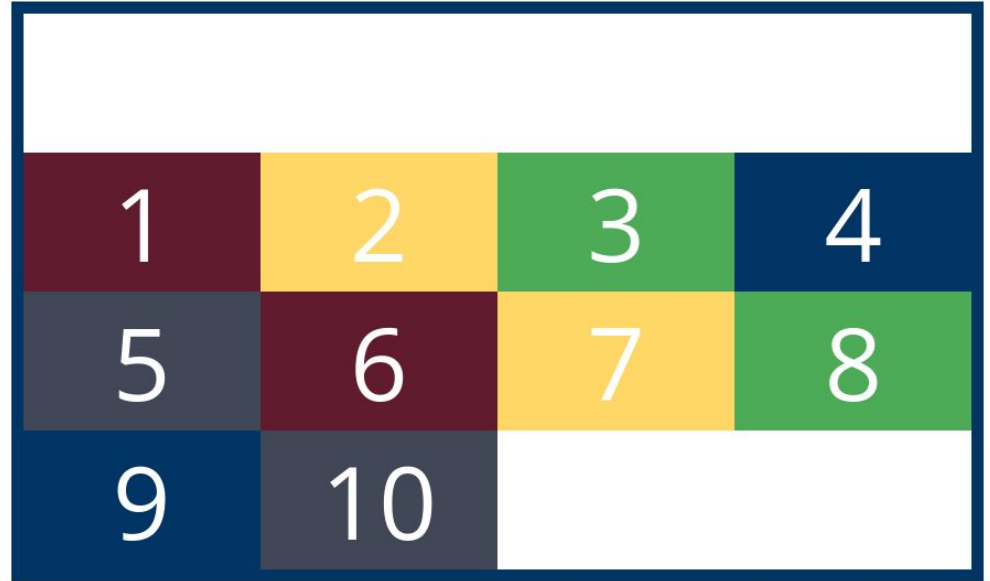
Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: flex-start;  
}  
  
.item {  
  width: 25%;  
}
```



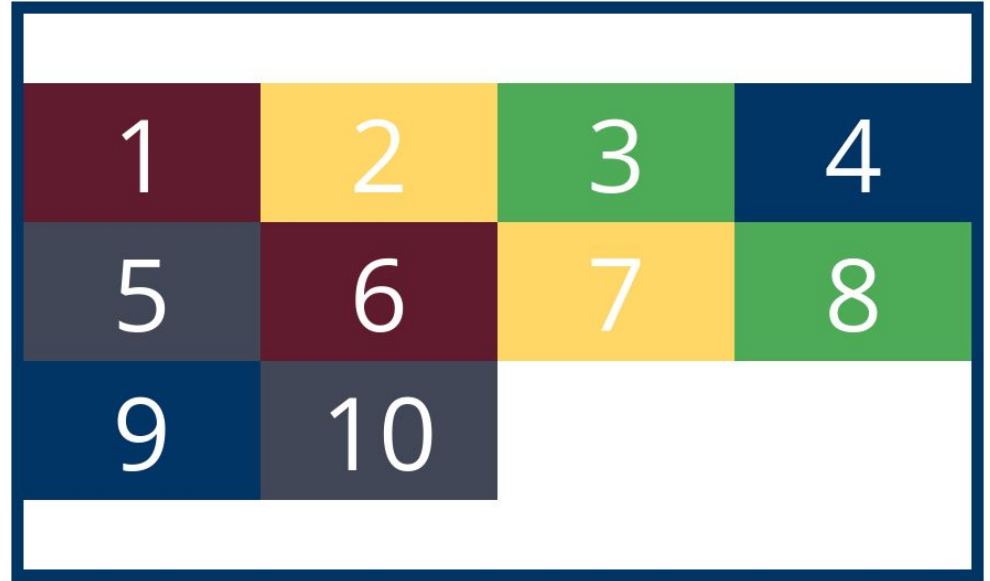
Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: flex-end;  
}  
  
.item {  
  width: 25%;  
}
```



Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: center;  
}  
  
.item {  
  width: 25%;  
}
```



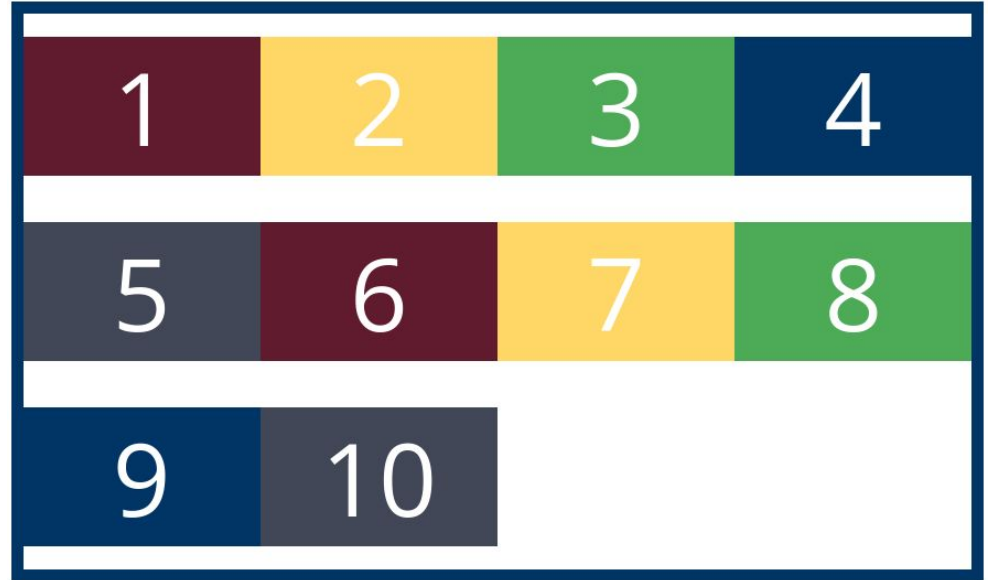
Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: space-between;  
}  
  
.item {  
  width: 25%;  
}
```



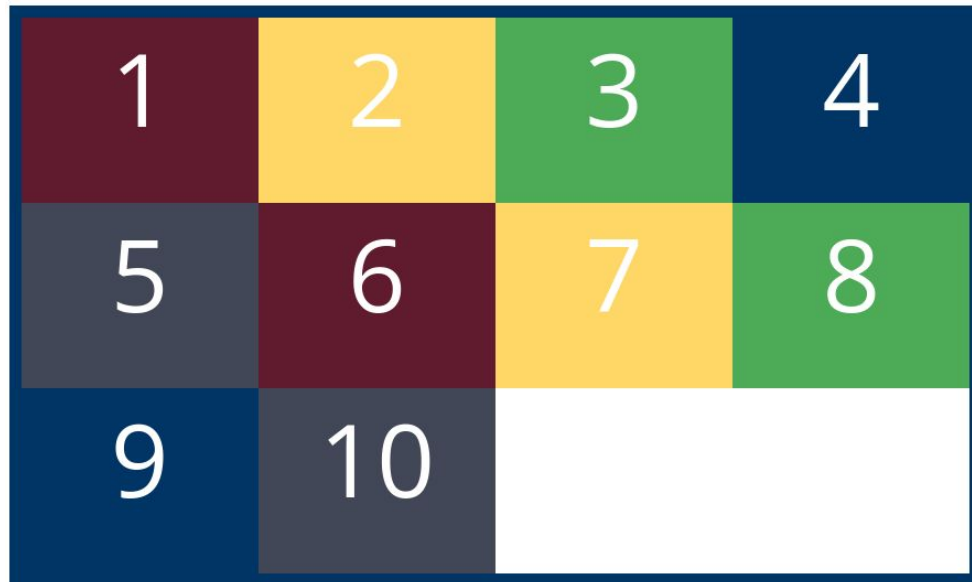
Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: space-around;  
}  
  
.item {  
  width: 25%;  
}
```



Flexbox

```
#container {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  height: 350px;  
  align-content: stretch;  
}  
  
.item {  
  width: 25%;  
}
```



Flexbox

flex

valores	none <code>'flex-grow flex-shrink flex-basis'</code>
padrão	0 1 auto
se aplica a	itens flexíveis (<i>flex</i>)
herança	não

- determina o quanto cada item pode ser “flexionado”

Flexbox

flex-grow (expansão)

valores	<i>número</i>
padrão	0
se aplica a	itens flexíveis (<i>flex</i>)
herança	não

Flexbox



`flex: 0 1 auto;`



`flex: 1 1 auto;`



`flex: 3 1 auto;`



Flexbox

flex-shrink (redução)

valores	<i>número</i>
padrão	1
se aplica a	itens flexíveis (<i>flex</i>)
herança	não

- 0 e 1 são suficiente
- os elementos para de encolher quando atingem o tamanho mínimo

Flexbox

flex-basis

valores *comprimento* | *percentual* | *content* | *auto*

padrão *auto*

se aplica a itens flexíveis (*flex*)

herança não

- define o tamanho inicial do item

Flexbox

`flex: 0 1 100px;`



Flexbox

order

valores *inteiro*

padrão 0

se aplica a itens flexíveis (*flex*)

herança não



Flexbox



Flexbox

```
.verde {  
    order: 1;  
}
```



Flexbox

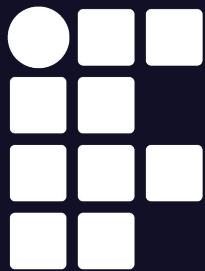
```
.verde .amarelo {  
  order: 1;  
}  
  
.cinza {  
  order: -1;  
}
```



Exercícios

1. Clone o repositório enviado na aulas anterior e crie uma nova seção.
2. A seção deve ter três colunas.
3. Na primeira coluna, faça uma lista com os 10 melhores filmes que você já assistiu.
4. Na segunda coluna, crie uma galeria com imagens relacionadas às suas atividades de lazer, com pelo menos 9 imagens. A galeria deve ser responsiva (ajustar-se à tela do dispositivo)
5. Na terceira coluna, faça uma lista de páginas web que você recomenda.
6. Use pseudo-classes e pseudo-elementos em todas as colunas.
7. O documento deve ser validado!





**INSTITUTO
FEDERAL**
Santa Catarina

Posicionando elementos com CSS

Programação Frontend I

Análise e Desenvolvimento de Sistemas

Prof. Adriano Lima

adriano.lima@ifsc.edu.br



INSTITUTO FEDERAL

Santa Catarina

Câmpus São José

PENSAMENTO
COMPUTACIONAL E
ALGORITMOS

Análise e Desenvolvimento
de Sistemas