

FLEX-BOX

2º ano de informática

Nome:

Arthur Barth



**INSTITUTO
FEDERAL**
Farroupilha

Definição de Flex-Box:

O Flexbox é um conceito do CSS que permite organizar os elementos de uma página HTML dentro de seus containers de forma dinâmica. Ou seja, independente das dimensões da tela, os elementos se ajustam automaticamente, garantindo uma melhor experiência de usuário em diferentes dispositivos.

Flex-direction:

A propriedade flex-direction define a direção do fluxo dos elementos filhos dentro do container flex. Ela determina se os elementos serão organizados horizontalmente (em linha) ou verticalmente (em coluna).

VALORES POSSÍVEIS

row: Organiza os elementos horizontalmente, da esquerda para a direita (padrão).

row-reverse: Organiza os elementos horizontalmente, da direita para a esquerda.

column: Organiza os elementos verticalmente, de cima para baixo.

column-reverse: Organiza os elementos verticalmente, de baixo para cima.

Exemplo de aplicação: Uma flex-box com X elementos filhos organizados verticalmente.

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

Flex-wrap:

A propriedade flex-wrap define se os elementos filhos de um container flex podem quebrar em várias linhas ou colunas.

VALORES POSSÍVEIS

nowrap: Os elementos filhos não quebram em várias linhas ou colunas (padrão).

wrap: Os elementos filhos quebram em várias linhas ou colunas conforme necessário.

wrap-reverse: Os elementos filhos quebram em várias linhas ou colunas conforme necessário, mas na direção inversa.

Exemplo de utilização: Uma flex-box com X elementos filhos que quebram em várias linhas quando necessário:

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

Flex-flow:

A propriedade flex-flow é um atalho para definir as propriedades flex-direction e flex-wrap simultaneamente. Ela determina como os elementos filhos de um container flex serão organizados e se poderão quebrar em várias linhas ou colunas.

Inserir primeiro o valor do flex-direction e depois do flex-wrap.

Exemplo de utilização: Uma flex-box com X elementos filhos organizados verticalmente que quebram em várias linhas quando necessário:

```
.container {  
  display: flex;  
  flex-flow: column wrap;  
}
```

Justify-content:

A propriedade justify-content define como os elementos filhos de um container flex serão distribuídos ao longo do eixo principal (horizontal para flex-direction: row e vertical para flex-direction: column).

Valores Possíveis:

flex-start: Alinha os elementos filhos na extremidade inicial do container flex.

flex-end: Alinha os elementos filhos na extremidade final do container flex.

center: Centraliza os elementos filhos dentro do container flex.

space-between: Distribui os elementos filhos uniformemente ao longo do container flex, com espaços iguais entre eles.

space-around: Distribui os elementos filhos uniformemente ao longo do container flex, com espaços iguais entre eles e nas extremidades.

space-evenly: Distribui os elementos filhos uniformemente ao longo do container flex, com espaços iguais entre eles e nas extremidades, exceto pelo primeiro e último elementos, que têm metade do espaço.

Exemplo de utilização: Uma flex-box com X elementos filhos que são distribuídos uniformemente ao longo do eixo horizontal.

```
.container {  
  display: flex;  
  justify-content: space-between;  
}
```

Align-items:

A propriedade align-items define como os elementos filhos de um container flex serão distribuídos ao longo do eixo secundário (vertical para flex-direction: row e horizontal para flex-direction: column).

Valores possíveis:

flex-start: Alinha os elementos filhos na extremidade inicial do container flex.

flex-end: Alinha os elementos filhos na extremidade final do container flex.

center: Centraliza os elementos filhos dentro do container flex.

baseline: Alinha os elementos filhos pela linha de base do texto.

stretch: Estica os elementos filhos para preencher todo o espaço disponível no container flex.

Exemplo de utilização: Uma flex-box com X elementos filhos centralizados verticalmente.

```
.container {  
  display: flex;  
  align-items: center;  
}
```

Align-content:

A propriedade align-content define como as linhas de elementos filhos de um container flex serão distribuídas ao longo do eixo transversal (vertical para flex-direction: row e horizontal para flex-direction: column).

Valores Possíveis:

flex-start: Alinha as linhas na extremidade inicial do container flex.

flex-end: Alinha as linhas na extremidade final do container flex.

center: Centraliza as linhas dentro do container flex.

space-between: Distribui as linhas uniformemente ao longo do container flex, com espaços iguais entre elas.

space-around: Distribui as linhas uniformemente ao longo do container flex, com espaços iguais entre elas e nas extremidades.

space-evenly: Distribui as linhas uniformemente ao longo do container flex, com espaços iguais entre elas e nas extremidades, exceto pela primeira e última linhas, que têm metade do espaço.

stretch: Estica as linhas para preencher todo o espaço disponível no container flex.

Exemplo de Utilização: Uma flex-box com X linhas de elementos filhos, distribuídas uniformemente no eixo vertical.

```
.container {  
  display: flex;  
  flex-direction: column;  
  align-content: space-between;  
}
```


Order:

A propriedade `order` define a ordem dos elementos filhos de um container flex. Por padrão, os elementos filhos são exibidos na ordem em que aparecem no código HTML. No entanto, a propriedade `order` permite alterar essa ordem.

Valores possíveis:

Qualquer número inteiro.

Valores negativos são permitidos.

Exemplo de utilização: Uma flex box com 3 elementos filhos, que terão suas ordens alteradas.

```
.container {  
  display: flex;  
}  
.item-1 {  
  order: 2;  
}  
.item-3 {  
  order: -1;  
}
```

Flex-grow:

A propriedade flex-grow define quanto espaço extra um elemento filho de um container flex receberá quando o container tiver espaço extra disponível.

Valores possíveis:

Qualquer número positivo ou zero.

O valor padrão é 0.

Exemplo de utilização: Flex-box com dois elementos filhos, com o segundo crescendo para ocupar o espaço disponível.

```
.container {  
  display: flex;  
}  
.elemento-2 {  
  flex-grow: 1;  
}
```

Flex-shrink:

A propriedade flex-shrink define quanto espaço um elemento filho de um container flex irá encolher quando o container tiver menos espaço disponível do que o necessário para exibir todos os seus filhos.

Valores possíveis:

Qualquer número positivo ou zero.

O valor padrão é 1.

Exemplo de utilização: Uma flex-box com dois elementos filhos, encolhendo o segundo para caber no espaço disponível.

```
.container {  
  display: flex;  
}  
.elemento-2 {  
  flex-shrink: 1;  
}
```

Flex-basis:

A propriedade flex-basis define o tamanho inicial de um elemento filho de um container flex. É o tamanho do elemento antes de qualquer espaço extra ser distribuído usando flex-grow e flex-shrink.

Valores possíveis:

Um valor de comprimento (por exemplo, "100px", "50%")

auto (o valor padrão)

Exemplo de utilização: Uma flex-box com dois elementos filhos, tendo o tamanho inicial do segundo definido.

```
.container {  
  display: flex;  
}  
.elemento-2 {  
  flex-basis: 200px;  
}
```

Flex:

A propriedade flex é uma abreviação para definir as seguintes propriedades de layout flexíveis em uma única linha:

- Flex-grow;
- Flex-shrink e
- Flex-basis.

Valores possíveis:

1° - flex-grow

2° - flex-shrink

3° - flex-basis

Exemplo de utilização: Uma flex-box com dois elementos filhos, definindo como eles devem crescer, encolher, e seu tamanho inicial.

```
.container {  
  display: flex;  
}  
.elemento-1 {  
  flex: 1 1 100px;  
}  
.elemento-2 {  
  flex: 2 1 200px;  
}
```

Align-self:

A propriedade `align-self` define como um elemento flexível individual se alinha dentro do seu container flex, substituindo o valor de `align-items` definido para o container flex.

Valores possíveis:

`auto`: O valor padrão, que usa o valor de `align-items` do container flex.

`flex-start`: Alinha o elemento na borda inicial do container flex.

`flex-end`: Alinha o elemento na borda final do container flex.

`center`: Alinha o elemento no centro do container flex.

`baseline`: Alinha o elemento com a linha de base do texto no container flex.

`stretch`: Estica o elemento para preencher todo o container flex em sua direção principal.

Exemplo de utilização: Flex-box com X elementos filhos e definir como o segundo deve se alinhar.

```
.container {  
  display: flex;  
  align-items: center;  
}  
  
.elemento-2 {  
  align-self: flex-end;  
}
```