

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

Aula 16

CSS - Responsividade

Professor: Thiago Theiry de Oliveira

O que vimos na aula passada:

- Modelo de caixa.
 - Box-sizing e border-box
 - Espaçamentos
 - Overflow.
 - Altura e larguras máximas e mínimas.

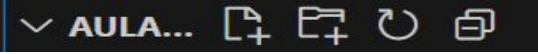


Objetivos da aula.. (Spoilers)

- CSS - Responsividade.
 - Flexbox.
 - Grid.
 - Media Queries



Configurando o ambiente

- Dentro da pasta aulaPratica02, vamos construir um arquivo de nome css1.html;
 - Selecione o primeiro ícone ao lado 
 - Desde já, TODOS os nossos arquivos DEVEM ter a extensão *.html
 - Use o atalho do vs code para abrir a estrutura padrão do html (pode ser o html simples ou html5)

O que é CSS Flexbox?

- Flexbox é a abreviação de módulo Flexible Box Layout.
- Flexbox é um método de layout para organizar itens em linhas ou colunas.
- O Flexbox facilita o design de uma estrutura de layout responsiva e flexível, sem usar float ou posicionamento.

Módulo de Layout de Caixa Flexível CSS

- Antes do módulo Layout de Caixa Flexível, havia quatro modos de layout:
 - Bloco, para seções em uma página da web
 - Em linha, para texto
 - Tabela, para dados de tabela bidimensionais
 - Posicionado, para posição explícita de um elemento

Componentes CSS Flexbox

- Um flexbox sempre consiste em:
 - um Flex Container - o elemento pai (container) <div>
 - Itens Flex - os itens dentro do contêiner <div>

Componentes CSS Flexbox

- Um contêiner flexível com três itens flexíveis
 - Para começar a usar o CSS Flexbox, você precisa primeiro definir um flex container.
 - O contêiner flexível se torna flexível ao definir a propriedade como display : flex

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

```
.flex-container {  
  display: flex;  
  background-color:  
  DodgerBlue;  
}
```

Componentes CSS Flexbox

- Um contêiner flexível com três itens flexíveis
 - Para começar a usar o CSS Flexbox, você precisa primeiro definir um flex container.
 - O contêiner flexível se torna flexível ao definir a propriedade como display : flex

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

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

A propriedade flex-direction do CSS

- A flex-direction propriedade especifica a direção de exibição dos itens flexíveis no contêiner flexível.
 - Esta propriedade pode ter um dos seguintes valores:
 - row
 - column
 - row-reverse
 - column-reverse

A propriedade flex-direction do CSS

- O row é o valor padrão e exibe os itens flexíveis horizontalmente (da esquerda para a direita):

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

A propriedade flex-direction do CSS

- O column valor exibe os itens flexíveis verticalmente (de cima para baixo):

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

A propriedade flex-direction do CSS

- O row-reverse valor exibe os itens flexíveis horizontalmente (mas da direita para a esquerda):

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

A propriedade flex-direction do CSS

- O column-reverse valor exibe os itens flexíveis verticalmente (mas de baixo para cima):

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

A propriedade CSS flex-wrap

- A flex-wrap propriedade especifica se os itens flexíveis devem ser quebrados ou não, caso não haja espaço suficiente para eles em uma linha flexível.
 - Esta propriedade pode ter um dos seguintes valores:
 - nowrap
 - wrap
 - wrap-reverse

A propriedade CSS flex-wrap

- O nowrap valor especifica que os itens flexíveis não serão quebrados (este é o padrão):

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

A propriedade CSS flex-wrap

- O nowrap valor especifica que os itens flexíveis não serão quebrados (este é o padrão):

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

A propriedade CSS flex-wrap

- O wrap valor especifica que os itens flexíveis serão encapsulados se necessário:

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

A propriedade CSS flex-wrap

- O wrap-reverse valor especifica que os itens flexíveis serão encapsulados, se necessário, na ordem inversa:

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

A propriedade justify-content do CSS

- A justify-content propriedade é usada para alinhar os itens flexíveis quando eles não usam todo o espaço disponível no eixo principal (horizontalmente).
 - Esta propriedade pode ter um dos seguintes valores:
 - center
 - flex-start
 - flex-end
 - space-around
 - space-between
 - space-evenly

A propriedade justify-content do CSS

- O valor center posiciona os itens flexíveis no centro do contêiner:

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

A propriedade justify-content do CSS

- O flex-start valor posiciona os itens flexíveis no início do contêiner (este é o padrão):

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

A propriedade justify-content do CSS

- O flex-endvalor posiciona os itens flexíveis no final do contêiner:

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

A propriedade justify-content do CSS

- O space-around valor exibe os itens flexíveis com espaço ao redor deles:

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

A propriedade justify-content do CSS

- O space-between valor exibe os itens flexíveis com espaço entre eles:

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

A propriedade justify-content do CSS

- O space-evenly valor exibe os itens flexíveis com espaço igual ao redor deles:

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

A propriedade justify-content do CSS

- O space-evenly valor exibe os itens flexíveis com espaço igual ao redor deles:

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

A propriedade align-items do CSS

- A align-items propriedade é usada para alinhar os itens flexíveis quando eles não usam todo o espaço disponível no eixo cruzado (verticalmente).
 - Esta propriedade pode ter um dos seguintes valores:
 - center
 - flex-start
 - flex-end
 - stretch
 - baseline
 - normal

A propriedade align-items do CSS

- O valor center posiciona os itens flexíveis no meio do contêiner:

```
.flex-container {  
    display: flex;  
    height: 200px;  
    align-items: center;  
}
```

A propriedade align-items do CSS

- O flex-start valor posiciona os itens flexíveis no topo do contêiner:

```
.flex-container {  
    display: flex;  
    height: 200px;  
    align-items: flex-start;  
}
```

A propriedade align-items do CSS

- O flex-end valor posiciona os itens flexíveis na parte inferior do contêiner:

```
.flex-container {  
    display: flex;  
    height: 200px;  
    align-items: flex-end;  
}
```

A propriedade align-items do CSS

- O stretch valor estica os itens flexíveis para preencher o contêiner (isso é igual a "normal", que é o padrão):

```
.flex-container {  
    display: flex;  
    height: 200px;  
    align-items: stretch;  
}
```

A propriedade align-items do CSS

- O baseline valor posiciona os itens flexíveis na linha de base do contêiner:

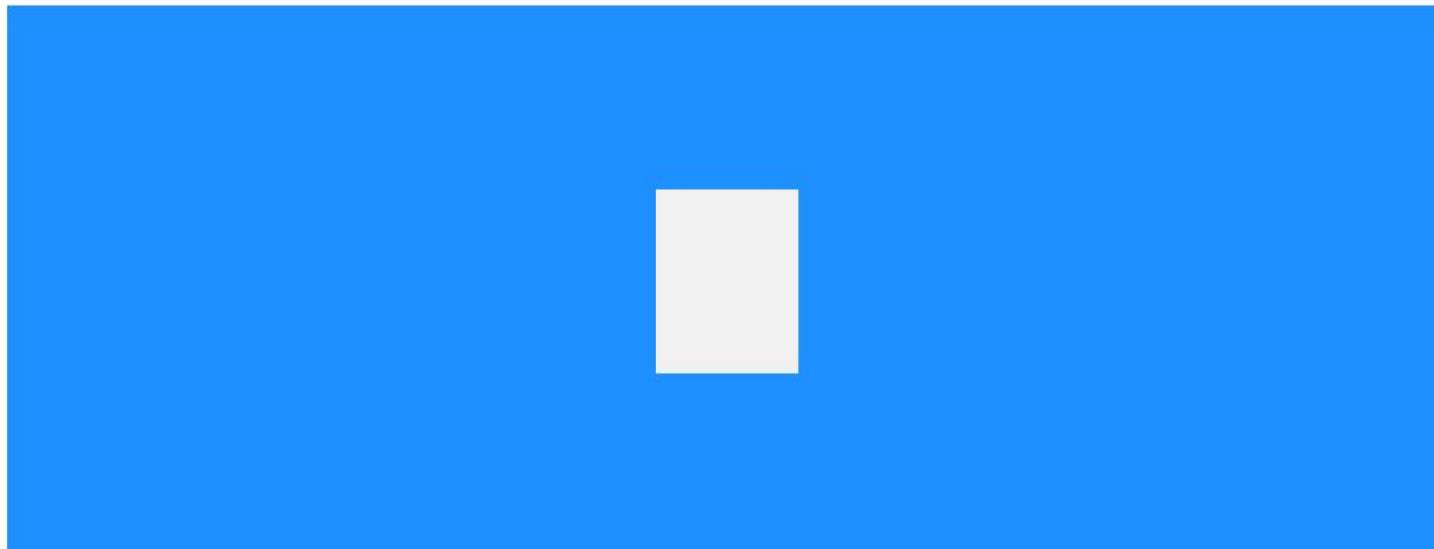
```
.flex-container {  
    display: flex;  
    height: 200px;  
    align-items: baseline;  
}
```

A propriedade align-content do CSS

- A align-content propriedade é semelhante a align-items, mas em vez de alinhar itens flexíveis, ela alinha as linhas flexíveis.
 - Esta propriedade pode ter um dos seguintes valores:
 - center
 - stretch
 - flex-start
 - flex-end
 - space-around
 - space-between
 - space-evenly

Exercício rápido - Centralização Perfeita

- Resolveremos um problema comum de estilo: centralização perfeita.



Recapitulando...

- CSS - Responsividade.
 - Flexbox. - Feito ✓
 - Grid. - Próxima aula
 - Media Queries - Próxima aula



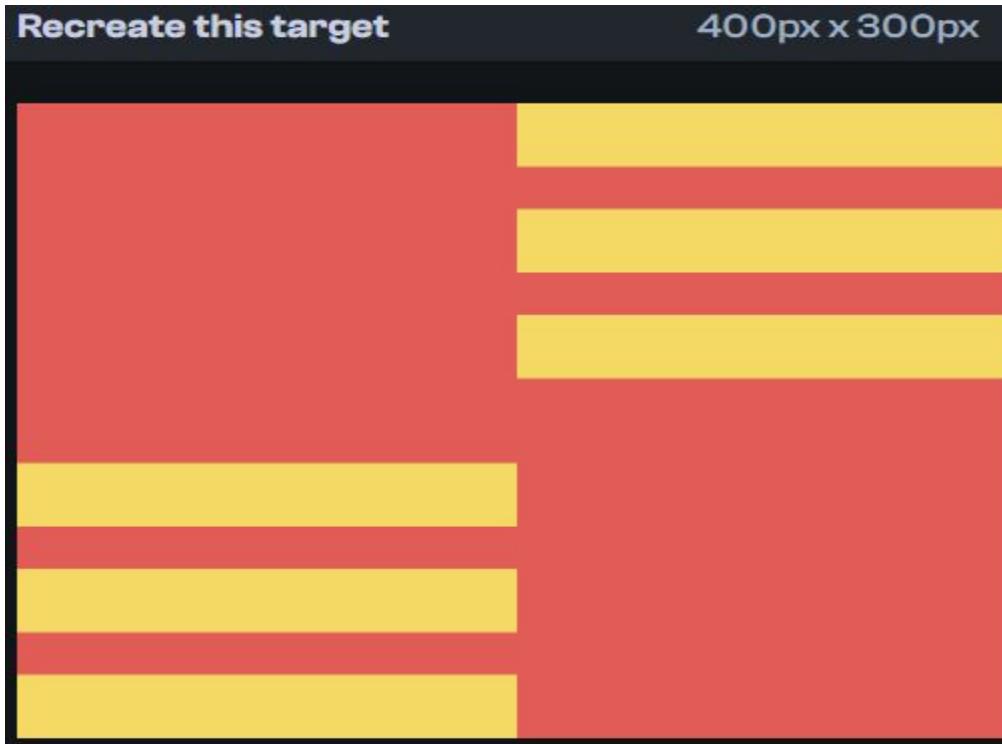
Dúvidas?

Entre em contato: thiago.thierry.17@gmail.com ou sala dos professores



Exercício 1

- Faça o css para gerar essa imagem:



Exercício 2

- Faça o css para gerar essa imagem:



Exercício 3

- Faça o css para gerar essa imagem (Dimensões e cores à sua escolha):

