

Aula 11

Finalizando nossa página

Curso

HTML e CSS: praticando
HTML e CSS

O que vamos aprender?



Aprender como fazer o alinhamento de imagem e de texto no site.



Aplicar o display flex.



Criar uma nova seção no site.

Disponível nos seus dispositivos favoritos



Clique e acompanhe a aula na Alura

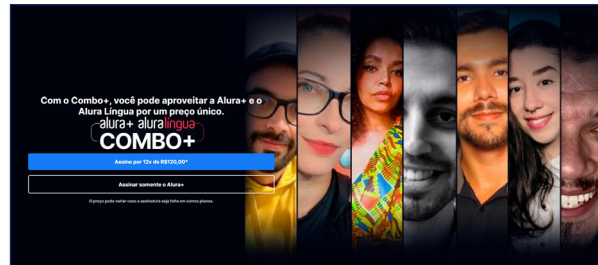
TV

Computadores

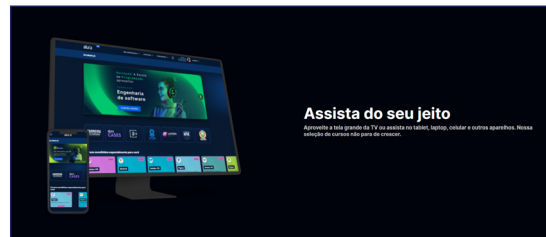
Celular

Flexbox

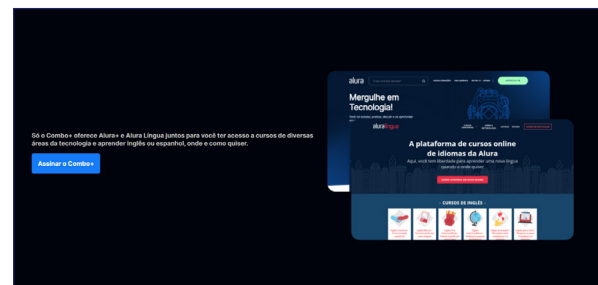
O nosso projeto Alura Plus já possui três seções. A primeira delas exibe dois botões para que o usuário se torne assinante; uma imagem de fundo com alguns dos instrutores da Alura; um título e uma imagem menor.



A segunda seção possui uma imagem de dois dispositivos diferentes, um monitor e um celular, além de um título e um texto com uma breve descrição.

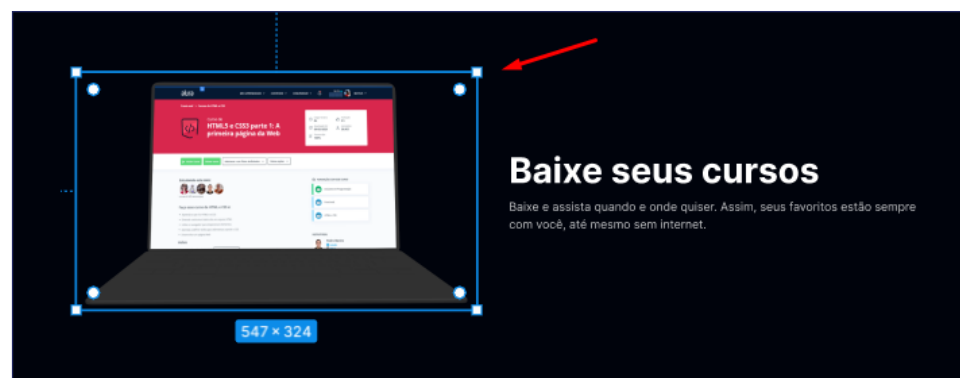


Já a terceira seção possui duas imagens do site da Alura, um pequeno texto e um botão para que o usuário se torne assinante.

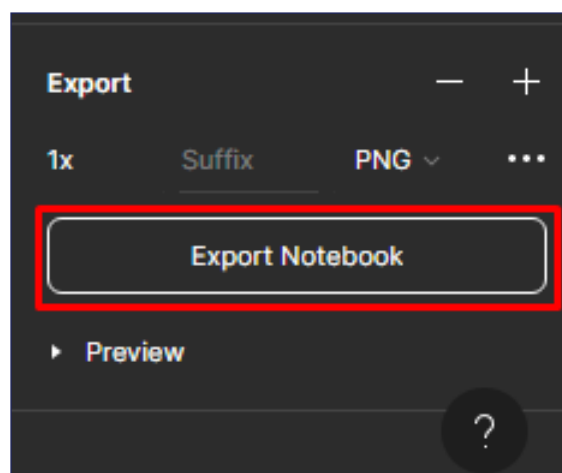
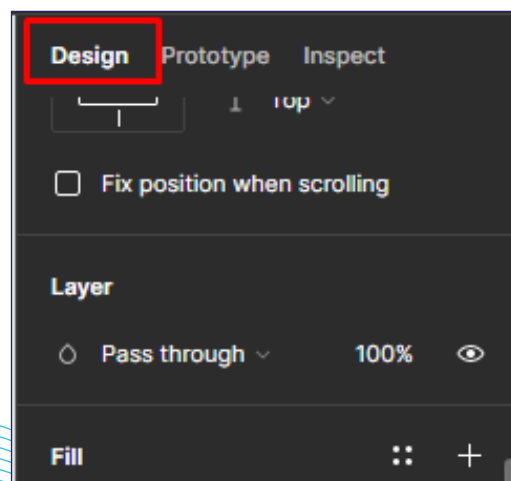


O desafio agora é criar uma quarta seção. Para colocar esse desafio em prática, vamos relembrar como criamos uma seção.

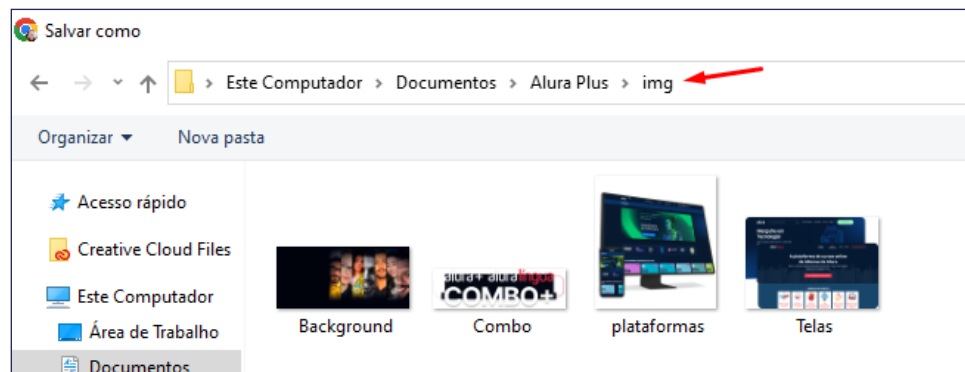
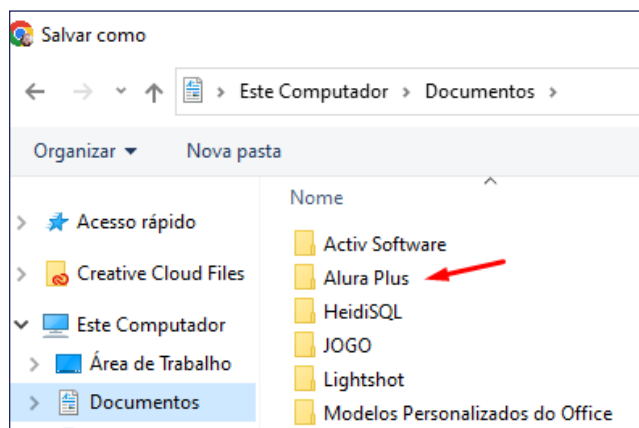
O primeiro passo é realizar o download da imagem. Para isso, devemos ir até o site figma.com e, então, clicar na imagem de um notebook.



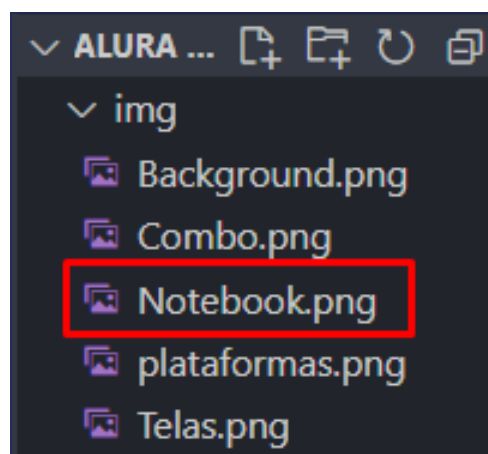
Ao clicarmos na imagem, a coluna *Design* deverá aparecer. Em seguida, devemos rolar a página até encontrarmos a opção *Export Notebook* e clicar nela.



O próximo passo é buscar pela pasta Alura Plus criada no computador e, então, salvar a imagem do notebook na subpasta nomeada como img.



Ao salvar a imagem e abrir o projeto no VSCode, é possível visualizar o arquivo da imagem no editor de código.



O passo seguinte é abrir no editor de código VSCode o arquivo index.html. Devemos criar no código uma nova seção por meio da tag **<section>**. Essa tag deverá ser aberta logo após o fechamento da tag **</section>** correspondente à última seção criada, que é a terceira seção.

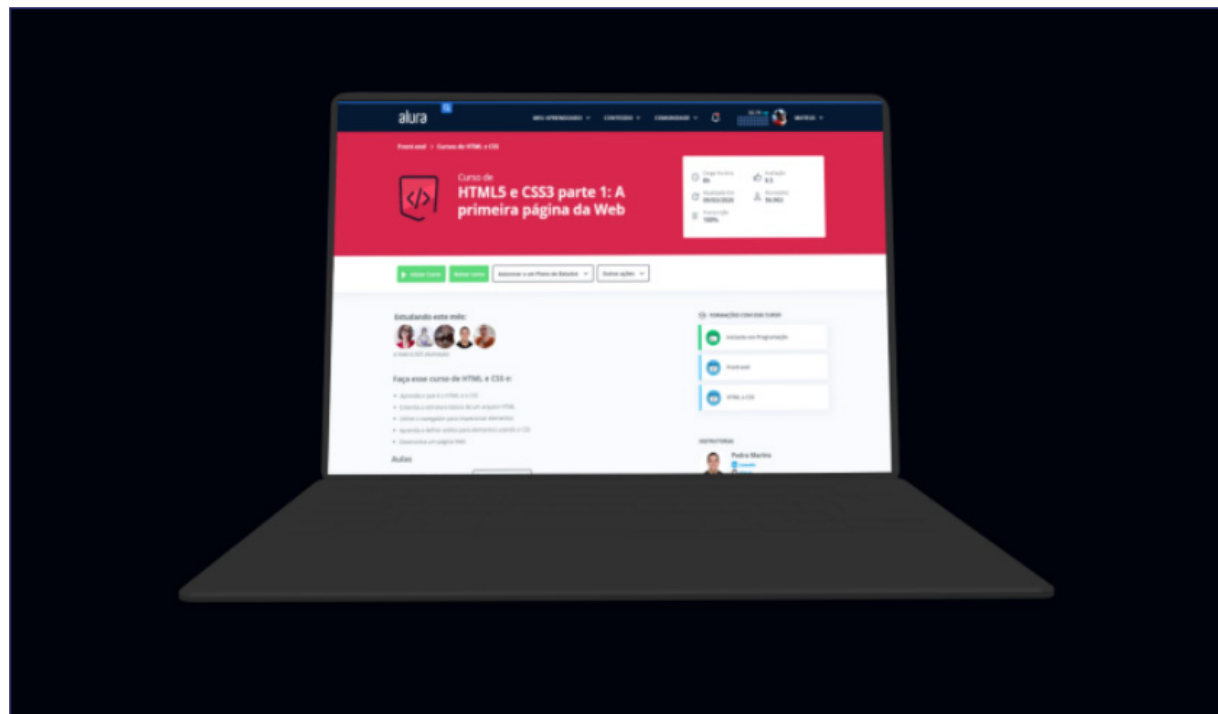
```
</section>

<section class="container secundario">
</section>
```

A imagem da nova seção que estamos criando agora deve ser posicionada à esquerda da tela do site. Isso significa que a imagem irá ocupar a primeira coluna da nossa página. Dessa forma, utilizamos a tag **** com o atributo **src="img/Notebook.png"**, que indica onde está a imagem. É importante lembrar a boa prática de programação que consiste em acrescentar uma descrição de cada imagem utilizando a propriedade **alt**.

```
<section class="container secundario">
  
```

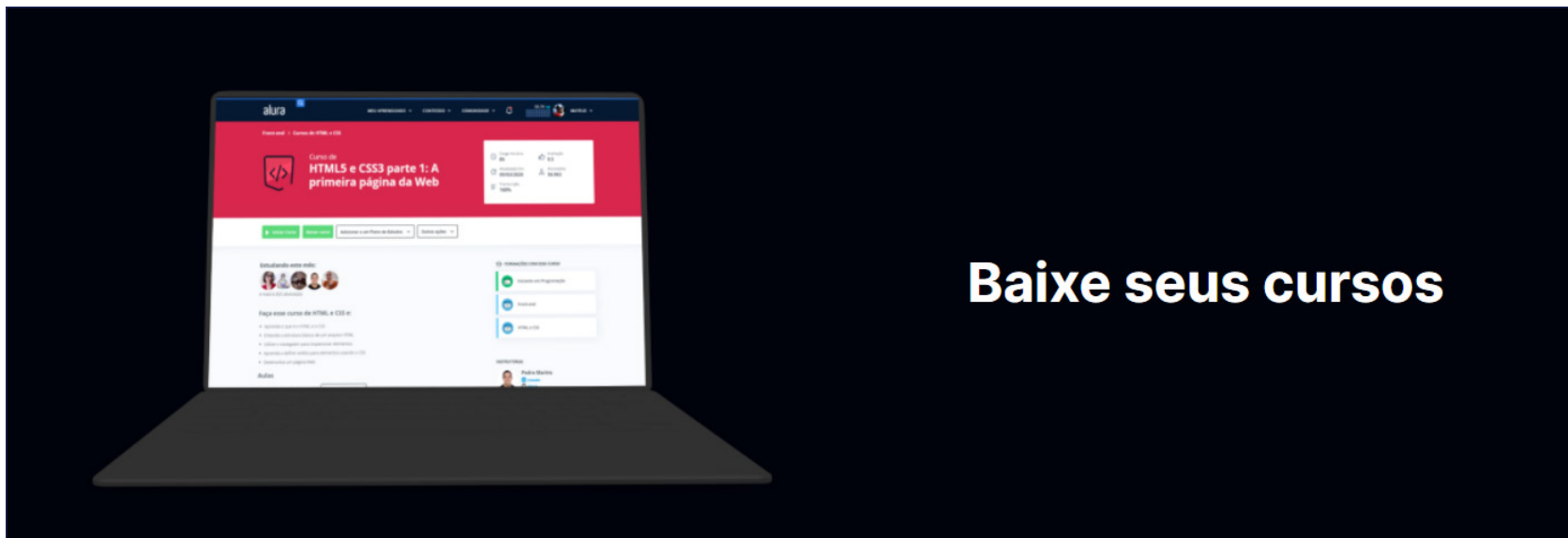
Ao salvarmos o arquivo e recarregarmos a página no navegador, podemos observar que a imagem do notebook já está posicionada no site.



Precisamos adicionar um título ao lado direito da imagem. Para isso utilizamos a tag de parágrafo **<h2>** logo abaixo da tag ****. E adicionamos a classe **.descricao__titulo**, que corresponde à estilização do título no CSS.

```
<h2 class="descricao__titulo">Baixe seus cursos</h2>
```

Ao carregarmos a página, podemos observar que o título está posicionado corretamente.



Vamos adicionar a descrição logo abaixo do título; para isso, utilizamos a tag `<p>` e a classe `.descricao__texto`.

```
<p class="descricao__texto">Baixe e assista quando e onde quiser.  
    Assim, seus favoritos estão sempre com você, até mesmo sem internet.</p>  
</section>
```

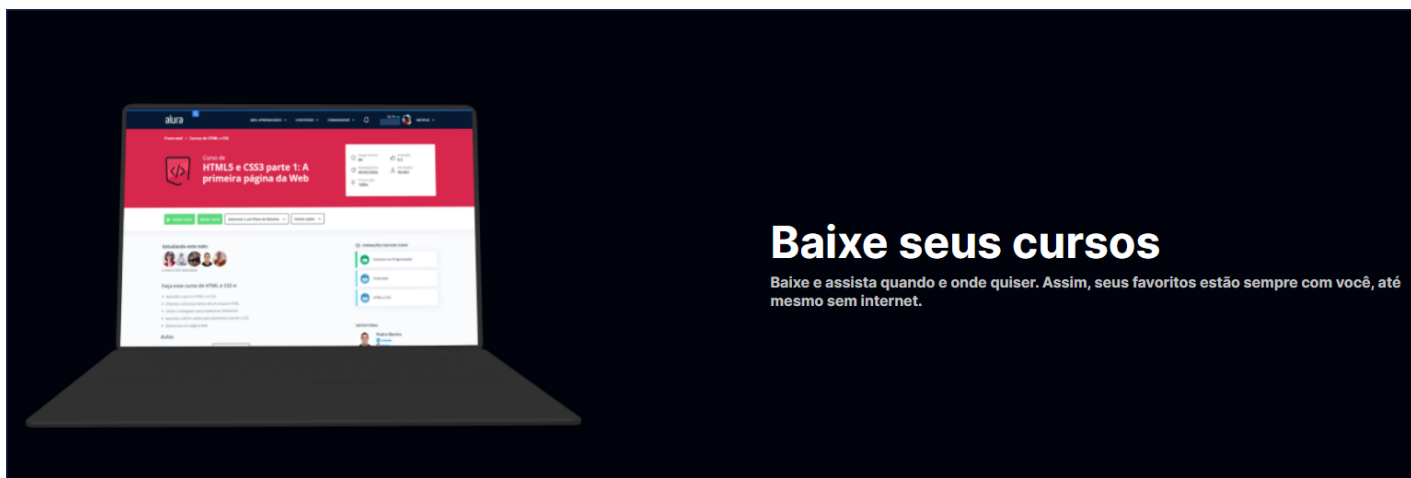
Ao carregarmos a página, temos o seguinte resultado para o texto. Note como o texto está distante do título “Baixe seus cursos”.



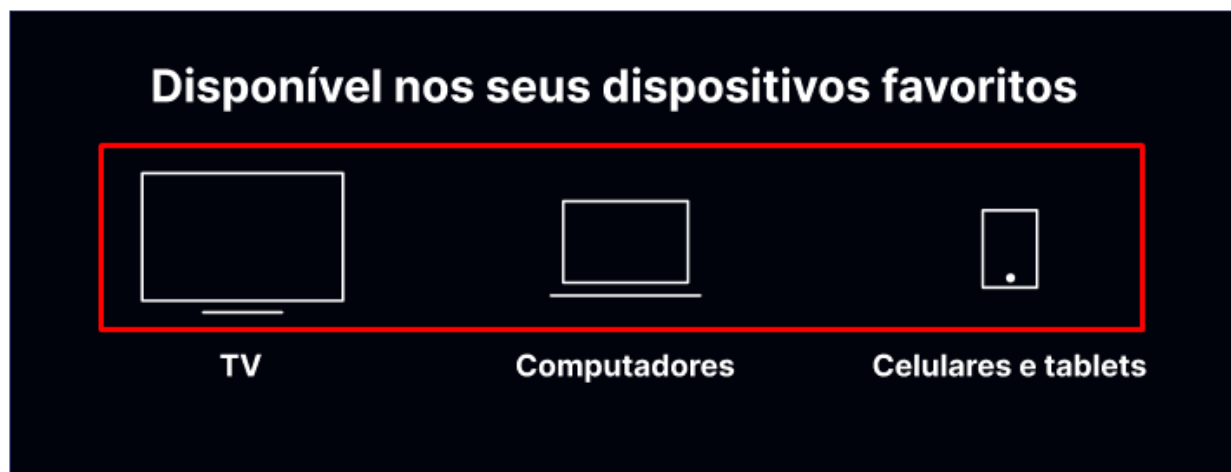
Os elementos título e texto de descrição ficaram distantes um do outro porque não estamos os envolvendo na mesma divisão. O comportamento automático da página é alocar cada elemento em uma coluna diferente da página, ou seja, em uma divisão diferente. Para resolvermos isso, vamos criar uma tag **<div>** e aplicarmos a classe **.container_descricao**.

```
<div class="container_descricao">  
  <h2 class="descricao_titulo">Baixe seus cursos</h2>  
  <p class="descricao_texto">Baixe e assista quando e onde quiser.  
    Assim, seus favoritos estão sempre com você, até mesmo sem internet.</p>  
</div>
```

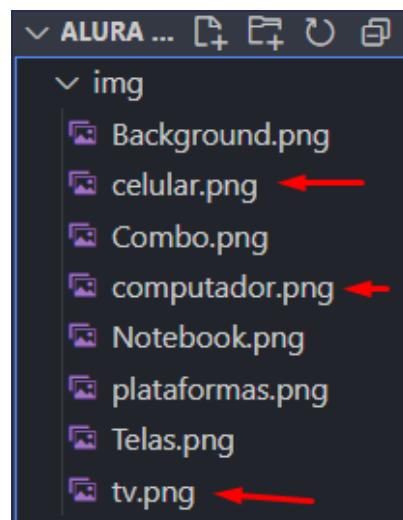
Ao carregarmos a página, percebemos que conseguimos criar a nova seção da página Alura Plus.



Vamos agora criar mais uma nova e última seção. Para criarmos essa última seção da página Alura Plus, vamos até o figma.com para fazer o download das imagens abaixo:



O processo para baixar e salvar as imagens é o mesmo realizado anteriormente. Uma vez feito, será possível visualizar as imagens no projeto do VSCode.



Vamos observar a nova seção. Perceba que ela possui um *display* bem diferente das outras seções criadas. Um *display* define a forma que as informações são exibidas em uma página. Vamos criar uma nova seção com uma classe chamada **.dispositivos**. Ela deve ser inserida logo abaixo do fechamento da última **</section>** no arquivo index.html.



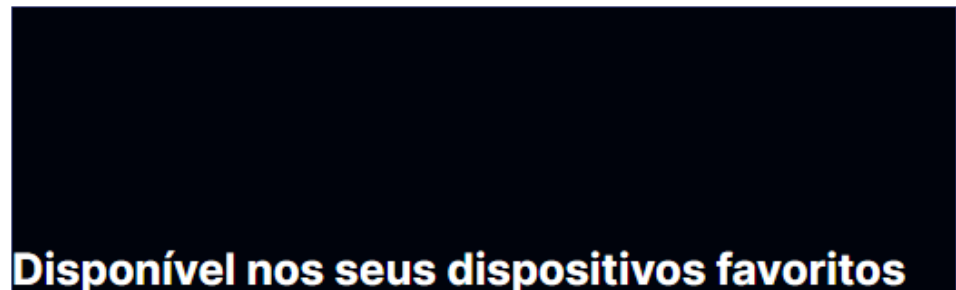
```
<section class="dispositivos">

</section>
```

Agora, vamos criar o título da seção. Para isto, usaremos a tag **<h2>** com uma nova classe chamada **.dispositivos_titulo**.

```
<section class="dispositivos">
  <h2 class="dispositivos_titulo">Disponível nos
  seus dispositivos favoritos</h2>
```

Ao carregarmos a página, podemos observar que o título foi adicionado no canto inferior esquerdo da tela. Ajustaremos o posicionamento por meio da linguagem CSS.



Disponível nos seus dispositivos favoritos

Para criarmos as imagens lado a lado, com um título embaixo de cada uma, será necessário utilizar a tag ****. Essa tag cria uma lista não ordenada de elementos. Mas o que significa uma lista não ordenada em HTML? Significa que essa lista não possui uma numeração e, portanto, a ordem dos itens é irrelevante. Portanto, vamos aplicar à nossa tag **** uma classe chamada **.dispositivos__lista**.

```
<h2 class="dispositivos__titulo">Disponível nos seus dispositivos favoritos</h2>
<ul class="dispositivos__lista">

</ul>
```

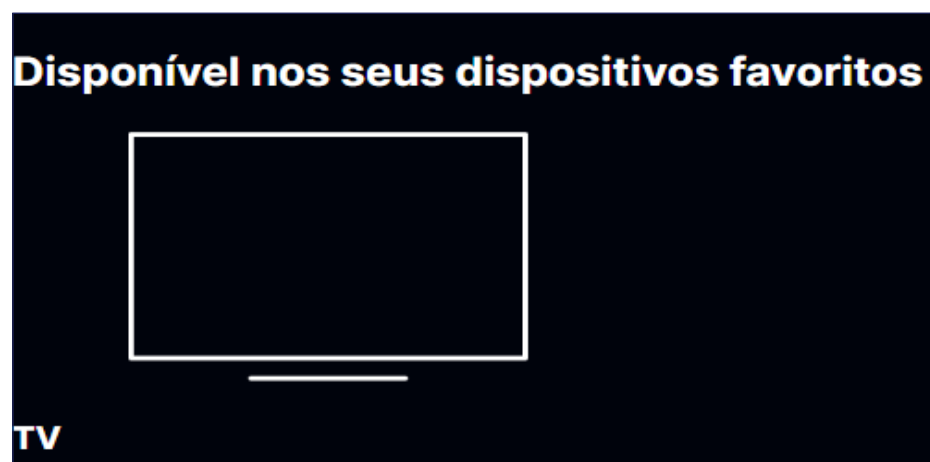
Cada elemento adicionado em uma lista **** é representado pela tag ****. Portanto, nós teremos três tags ****, sendo cada tag uma imagem e o título correspondente.

O título da imagem deve ter tamanho menor. Portanto, usaremos a tag **<h3>**. Quanto maior o valor na tag, por exemplo **<h4>**, **<h5>** e **<h6>**, menor será o tamanho da fonte. Esse padrão de títulos começa com o número 1 e vai até o número 6 somente.

Vamos começar com o primeiro elemento. Adicionamos a tag **** para exibir a imagem de televisão e adicionamos o título “TV”.

```
<ul class="dispositivos__lista">
  <li>
    
    <h3 class="lista__item">TV</h3>
  </li>
```

Ao carregarmos a página, podemos observar que os elementos foram adicionados.



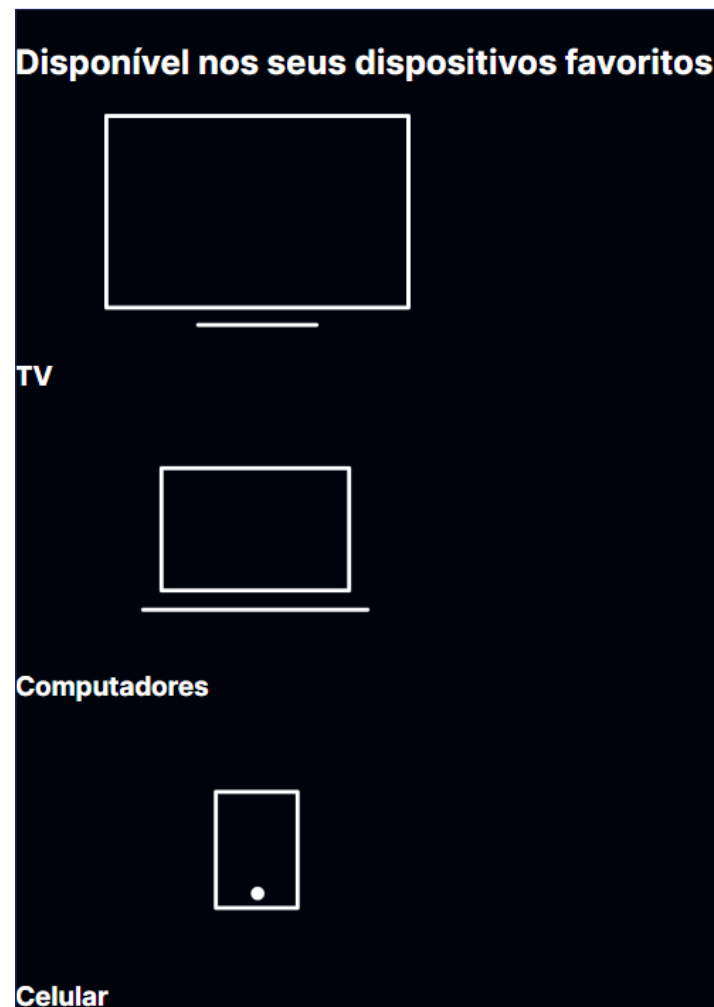
O próximo elemento **** será a imagem do computador e o título. Portanto, faremos o mesmo processo: acrescentamos uma nova tag **** e adicionamos a imagem e o título.

```
<li>  
    
  <h3 class="lista__item">Computadores</h3>  
</li>
```

Faremos o mesmo passo com o elemento celular. Perceba que as tags **** estão sendo adicionadas em sequência.

```
<li>  
    
  <h3 class="lista__item">Celular</h3>  
</li>
```

Ao carregarmos a página, podemos observar que todos os elementos estão adicionados. No entanto, eles devem ser ajustados para que se pareçam da mesma forma que o protótipo no figma.com.

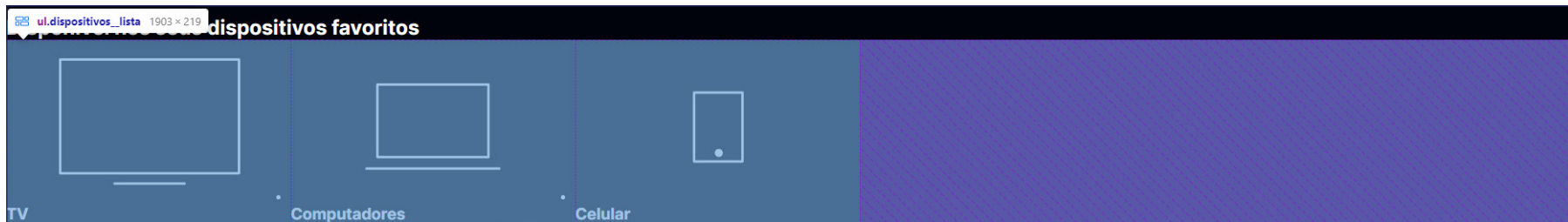


Para organizarmos os elementos um ao lado do outro, vamos até o arquivo style.css para trabalhar na classe **.dispositivos__lista**.

Até agora aprendemos diferentes formas de organizar os elementos de acordo com o nosso objetivo. Desta vez, utilizaremos a propriedade **display** com o valor **flex**. O valor **flex** cria uma faixa horizontal para organizar os elementos lado a lado. Então, vamos adicionar essa propriedade à classe.

```
.dispositivos__lista {  
  display: flex;  
}
```

Perceba que, ao carregarmos a página, os elementos estão divididos em colunas distintas, porém, todos estão dispostos horizontalmente. Utilizando as ferramentas do desenvolvedor, por meio da tecla F12 podemos visualizar a estrutura tal como a da imagem a seguir.



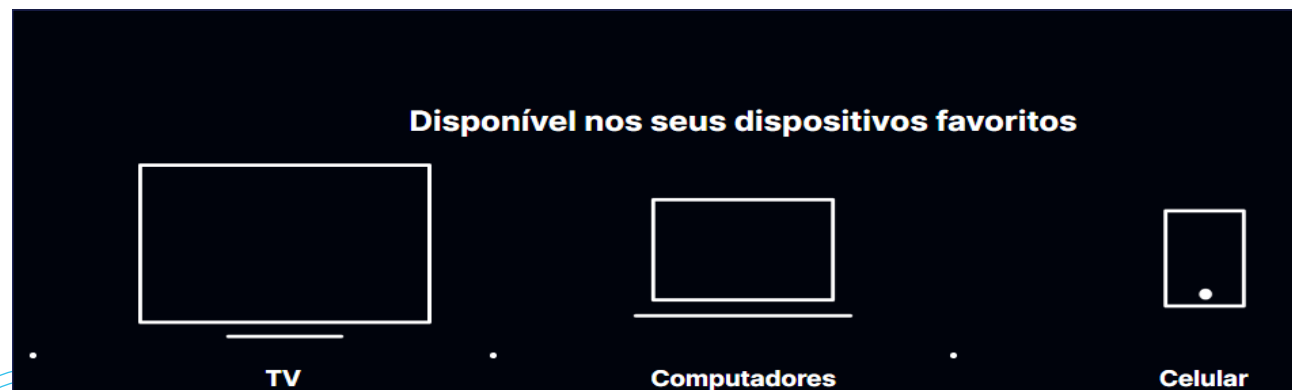
As imagens, assim como o título, devem estar centralizadas. Para isso, vamos acrescentar a propriedade **justify-content**, que, em inglês, significa “justificar o conteúdo da página”, e definimos seu valor como **center**.

```
display: flex;  
justify-content: center;
```

Para centralizar o título, vamos utilizar a classe **.dispositivos** e, então, adicionar o valor **center** à propriedade **text-align**. Essa propriedade centraliza todos os elementos de uma mesma linha.

```
.dispositivos {  
  text-align: center;  
}
```

Ao carregarmos a página, podemos observar que os elementos estão posicionados de forma similar ao protótipo do figma.com; no entanto, cada título exibe junto de si um elemento que não queremos em nossa página. Aprenderemos como retirá-lo.



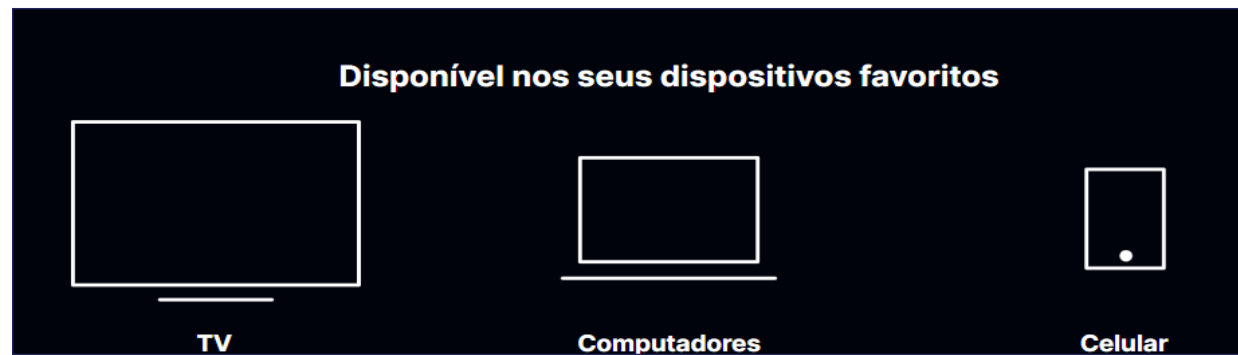
Em algum momento de sua vida, você criou uma lista de compras, de tarefas ou de tópicos do conteúdo que caem na prova? Caso sim, ao escrevermos listas é comum utilizarmos um ponto ou um traço antes de cada item, certo? Quando usamos a tag `` para criarmos uma lista no HTML, o comportamento padrão do elemento é a inserção de um *bullet* (ponto) antes de seu texto, como podemos observar na imagem abaixo:



Porém, em nosso projeto não queremos que os *bullets* sejam exibidos. Para que isso não ocorra, vamos até a classe **.dispositivos_lista** e aplicamos a propriedade **list-style-type**, que define um estilo para os elementos da lista. Para essa propriedade, vamos adicionar o valor **none**, que significa “nenhum”.

```
justify-content: center;
list-style-type: none;
}
```

Ao carregarmos a página, podemos observar que os *bullets* não estão presentes na tela.



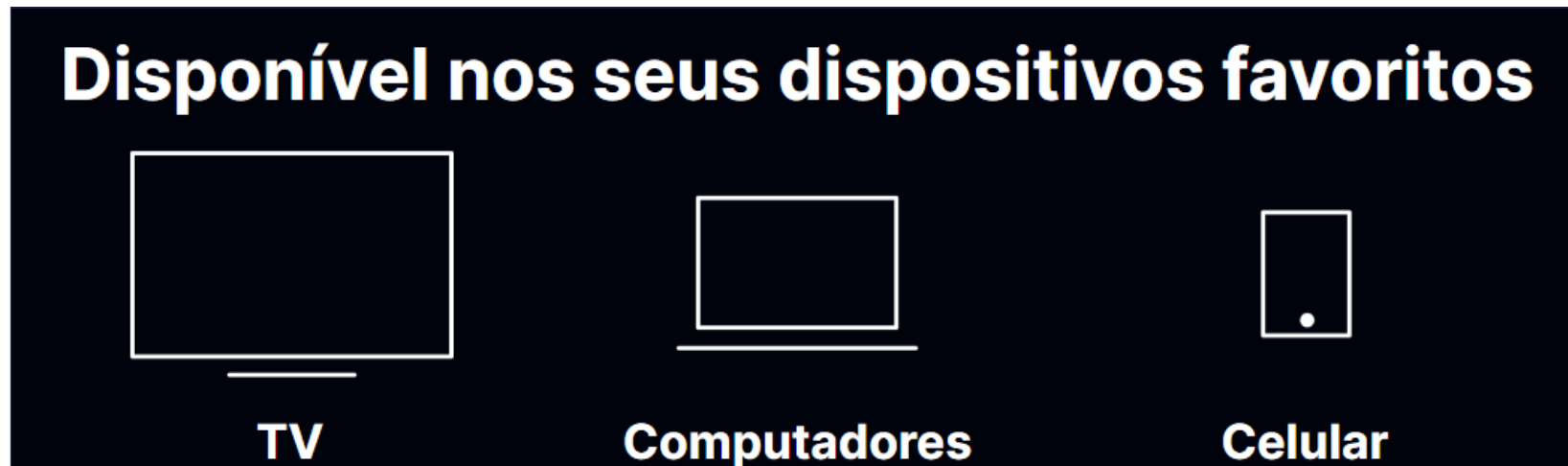
Vamos continuar a estilização dos elementos. O próximo passo é definir o tamanho de fonte e a cor dos títulos. Na classe **.dispositivos__titulo**, vamos acrescentar o tamanho **48px** e a cor branca. Lembrando que já criamos uma variável que armazena o padrão de cor estabelecido na página inteira, e, apesar de ser branca, é uma boa prática de programação organizarmos as cores em variáveis.

```
.dispositivos__titulo {  
  font-size: 48px;  
  color: var(--branco-principal);  
}
```

Faremos o mesmo processo para os títulos correspondentes às imagens, mas, desta vez, utilizando a classe **.lista-item**.

```
.lista_item {  
  font-size: 32px;  
  color: var(--branco-principal);  
}
```

Ao carregarmos a página, podemos observar que os elementos estão corretamente posicionados e com a cor e o tamanho de fonte alterados.



No entanto, os elementos não possuem um espaçamento externo, isto é, a margem não está bem definida; por essa razão, os elementos estão “grudados” no fim da página. Para resolvermos isso, vamos até a classe **.dispositivos__lista** para aplicarmos a propriedade **margin** com os valores **5em** e **0**. Isto significa que as margens de cima e de baixo terão **5em** como medida e as margens direita e esquerda, **0**.

```
list-style-type:none;  
margin: 5em 0;
```

Ao carregarmos a página, podemos observar que os elementos agora estão organizados adequadamente.

Disponível nos seus dispositivos favoritos



TV



Computadores



Celular

Desafio

Aprendemos a criar uma lista de elementos e posicioná-los de acordo com o protótipo da página Alura Plus. Uma das propriedades que utilizamos é chamada de **list-style-type**, ou, em português, tipo de estilo de uma lista. Aplicamos o valor **none**, que excluía os pontos (*bullets*) do início de cada título. Além do valor **none**, a propriedade **list-style-type** possui diferentes estilizações que podem ser aplicadas nos elementos de uma lista. Para o desafio de hoje, você deverá fazer uma breve pesquisa sobre os possíveis valores da propriedade e então aplicar nos elementos da página.