

Aula 6

Checando a responsividade

Unidade

**Página web: utilizando a responsividade
em aplicações com HTML e CSS**

Parte 2

O que vamos aprender?



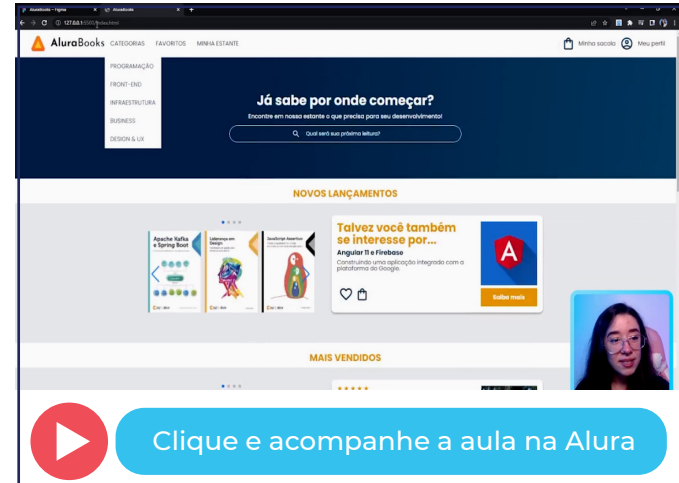
Revisar os elementos para o layout desktop.



Refatorar o CSS.

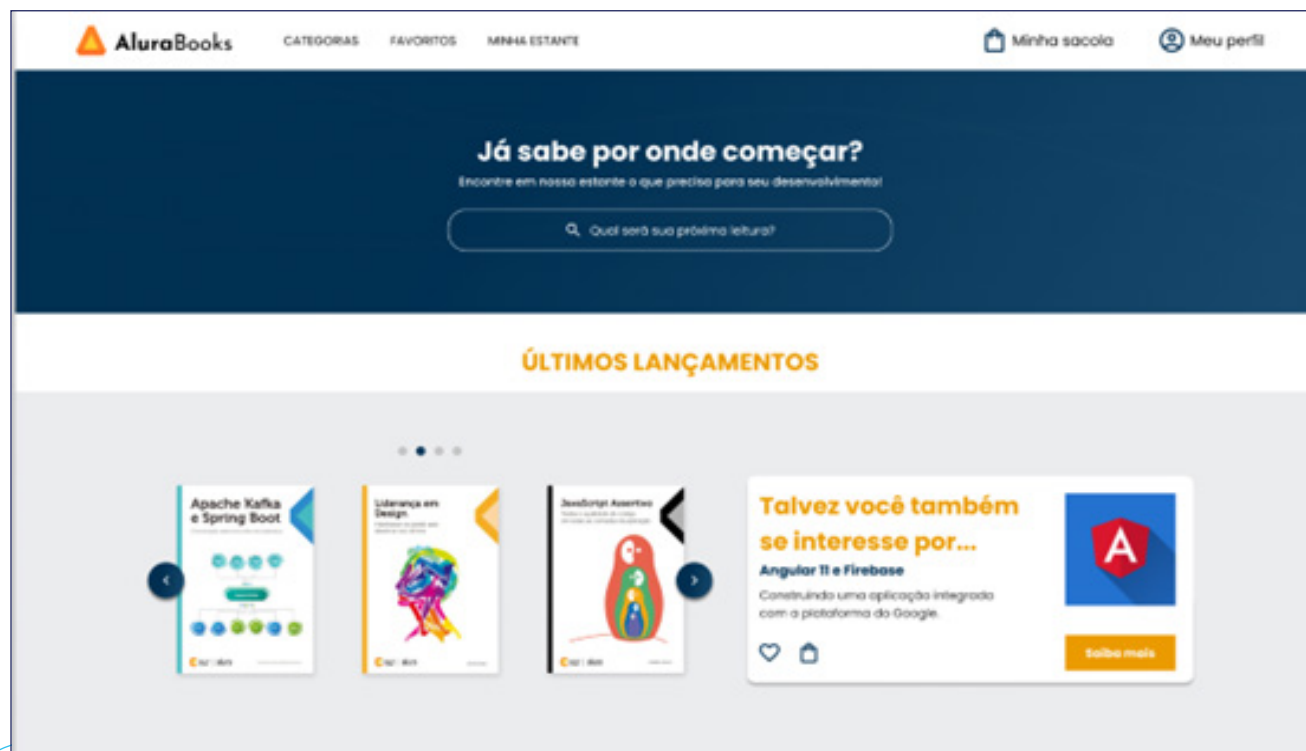


Utilizar as Media Queries.



Revisando o flexbox

Dando continuidade à etapa de aumentar o layout para desktop, vamos observar no Figma quais elementos ainda temos que ajustar, começando pelo banner.



Podemos observar que a barra de pesquisa necessita de ajuste de largura. Para ficar mais próximo do que vemos no Figma, vamos iniciar uma nova Media Query, no arquivo *banner.css*, e ajustar a largura.

```
@media screen and (min-width: 1728px) {  
  .banner_pesquisa {  
    width: 35%;  
  }  
}
```

Utilizando a propriedade **background-position**, ajustaremos a posição da imagem da lupa.

```
.banner_pesquisa::placeholder {  
  background-position: 12em;  
}
```

Por fim, vamos aumentar o espaçamento interno do banner com a propriedade **padding**.

```
.banner {  
  padding: 6em 0;  
}
```

Partindo para o menu carrossel, no arquivo *index.html*, vamos criar uma tag **<div>** para agrupar o carrossel e o card, para que possamos alinhá-los lado a lado no CSS.

```
<div class="carrossel__container">
  <div class="swiper">
    <!-- Additional required wrapper -->
    <!-- If we need pagination -->
    <div class="swiper-pagination"></div>

    <div class="swiper-wrapper">
      <!-- Slides -->
```

No arquivo *carrossel.css*, vamos criar uma nova Media Query, selecionar a nova classe **carrossel__container** e transformá-la em um objeto **flex**.

```
@media screen and (min-width: 1728px) {  
  .carrossel__container {  
    display: flex;  
  }  
}
```

Com a propriedade **margin**, vamos limitar automaticamente o tamanho dos componentes, além de realizar o alinhamento ao centro dos mesmos com a propriedade **align-items: center**.

```
.carrossel__container {  
  display: flex;  
  margin: 0 20vw 3em 20vw;  
  align-items: center;  
}
```

Vamos também, em seguida, ajustar a margem da paginação do carrossel, assim como seu tamanho total, utilizando as propriedades **margin 1em 0** e **width: 50%**.

```
.swiper-pagination {  
  margin: 1em 0;  
}  
  
.swiper {  
  width: 50%;  
}
```

Com a propriedade **font-size**, vamos ajustar os tamanhos de fonte do título e texto do card. Atribuiremos o valor **16px**.

```
.descricao_titulo {  
  font-size: 32px;  
}  
  
.descricao_texto {  
  font-size: 16px;  
}
```

Por fim, ajustaremos as margens e tamanho do card.

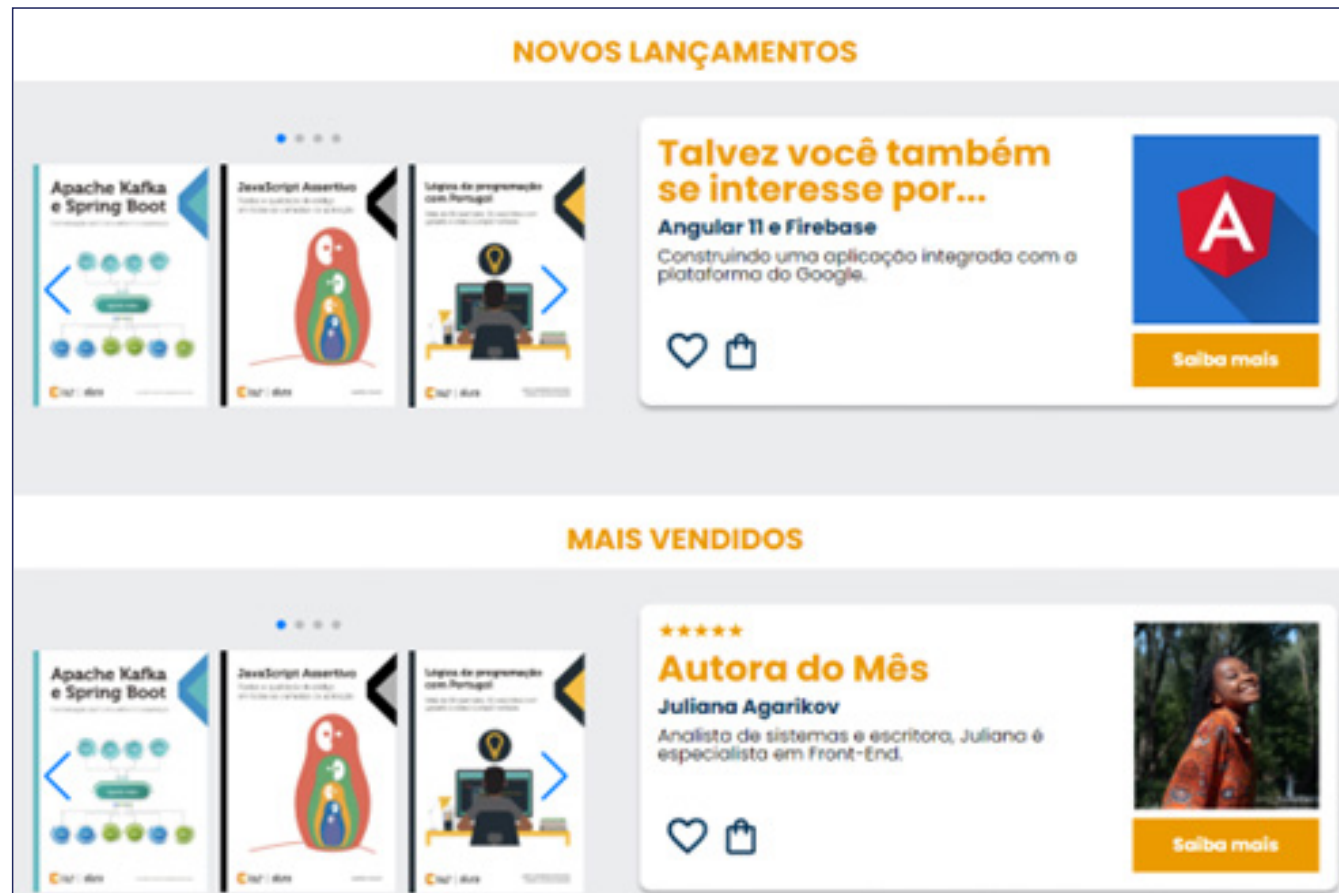
```
.descricao {  
  margin-right: 2em;  
}  
  
.card {  
  width: 60%;  
  margin-left: 3em;  
}
```


Para a seção de *Mais vendidos*, precisaremos apenas criar a tag envolvendo o carrossel e o card, com a mesma classe, no arquivo *index.html*. Fazendo isso, poderemos reutilizar as mesmas estilizações CSS que usamos na seção anterior.

```
<div class="carrossel__container">
  <div class="swiper">
    <!-- Additional required wrapper -->
    <!-- If we need pagination -->
    <div class="swiper-pagination"></div>

    <div class="swiper-wrapper">
      <!-- Slides -->
```

Com essas modificações, o resultado deve estar semelhante ao da imagem a seguir.



Vamos agora criar uma nova Media Query para ajustar o espaçamento interno da seção de *Tópicos*, no arquivo *topicos.css*.

```
@media screen and (min-width: 1728px) {  
  .topicos {  
    padding: 3em 30vw;  
  }  
}
```

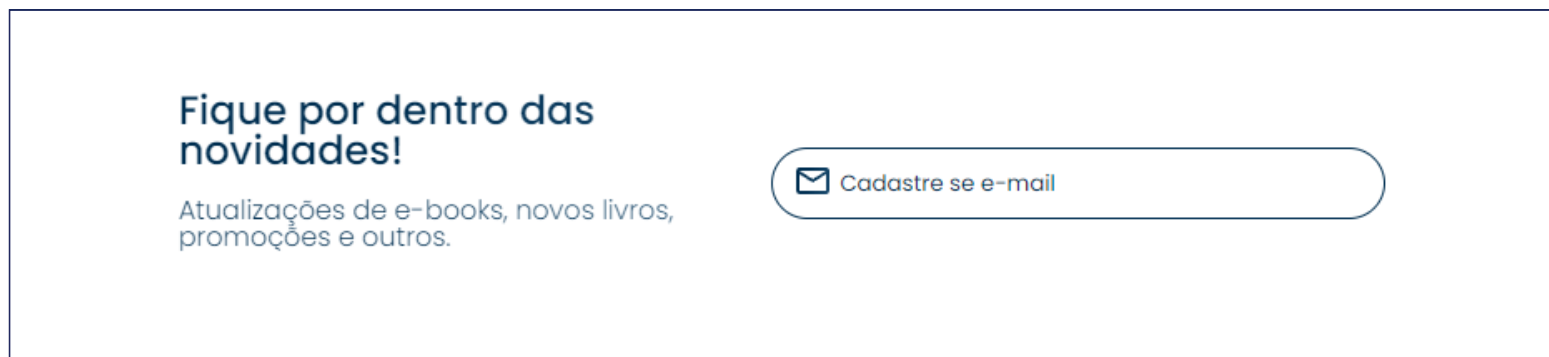
O resultado deve ser semelhante ao da imagem a seguir.



Na seção de *Contato*, vamos aplicar as mesmas estilizações da seção anterior, para espaçar corretamente o conteúdo das bordas, no arquivo *contato.css*.

```
@media screen and (min-width: 1728px) {  
  .contato {  
    padding: 3em 20vw;  
  }  
}
```

Obtemos o resultado a seguir:



Em seguida, no arquivo *rodape.css*, vamos acrescentar uma borda à esquerda das listas e ajustar o espaçamento interno, utilizando uma nova Media Query.

```
@media screen and (min-width: 1728px) {  
  .lista-rodape {  
    border-left: 1px solid var(--cinza-claro);  
    padding-left: 1em;  
  }  
  
  .rodape {  
    padding: 3em 0;  
  }  
}
```

Com essa última modificação, o rodapé deverá ficar com o aspecto da imagem a seguir.



Desafio

Nesta aula aplicamos a responsividade com Media Queries no restante do projeto.

Utilizamos a propriedade **border-left** para criar especificamente uma borda apenas à esquerda. O desafio será pesquisar sobre quais tipos de propriedades referentes à borda nós temos no CSS.

Avalie este material acessando o link: <https://forms.gle/EcEZdj59zGS9QTWY9>