

Aula 12

Informações de rodapé

Unidade

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

Parte 1

O que vamos aprender?



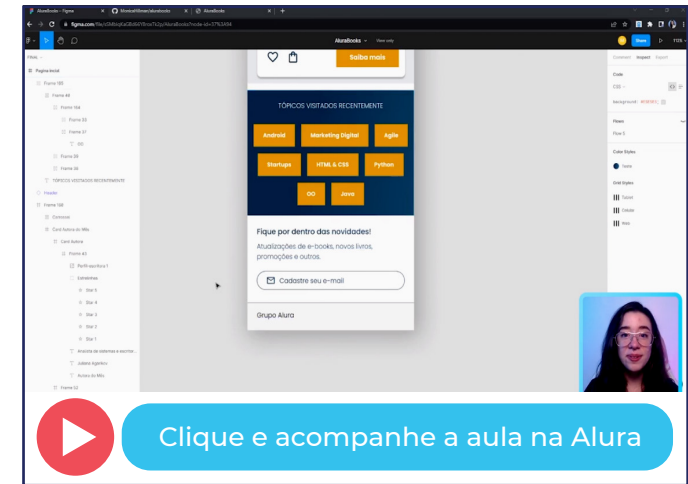
Construir a seção Contato.



Utilizar o input do tipo e-mail.

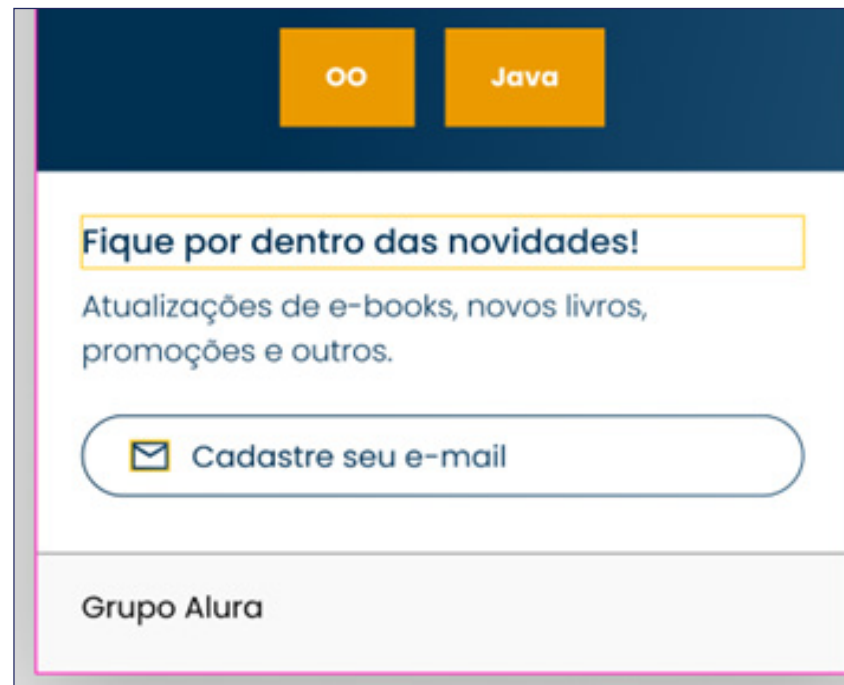


Construir o rodapé.



Contato e footer

Seguindo o projeto da página da Alura Book, chegamos à última etapa do desenvolvimento do layout para mobile. Vamos construir a seção de Contato e o footer. Este último nada mais é que o rodapé da página. Sempre vamos nos basear no protótipo que vemos no Figma. Observe pela imagem a seguir que o rodapé contém a frase *Grupo Alura*.

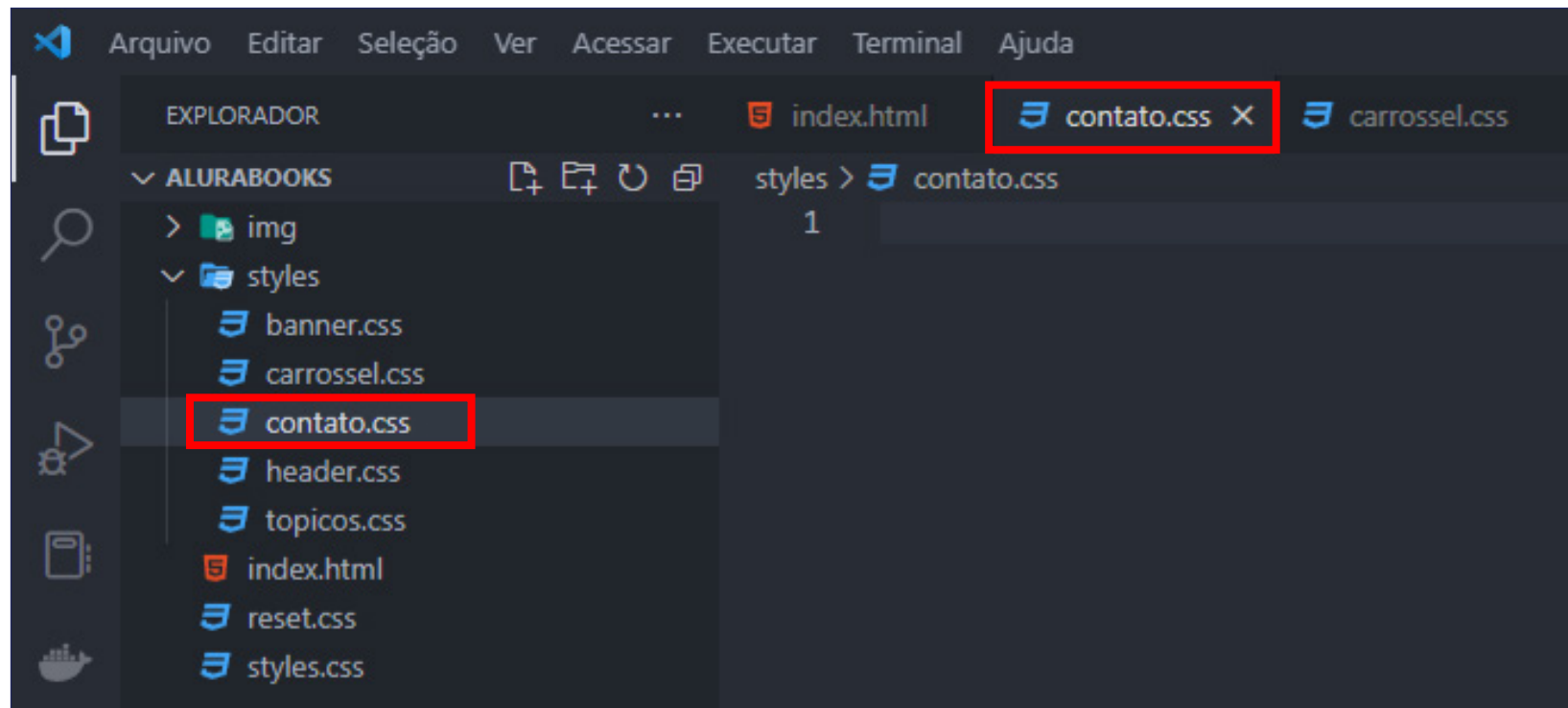


The image shows a mobile layout prototype for the Alura Book website. At the top, there is a dark blue header with two orange buttons labeled "OO" and "Java". Below the header, there is a white section with a yellow border. Inside this section, there is a yellow box with the text "Fique por dentro das novidades!". Below this, there is a line of text: "Atualizações de e-books, novos livros, promoções e outros." Below the text, there is a rounded rectangle containing an envelope icon and the text "Cadastre seu e-mail". At the bottom of the prototype, there is a light gray footer with the text "Grupo Alura".

Após a última tag **<section>** vamos criar uma nova tag de classe chamada **"contato"**. Utilizaremos as tags **<h2>** para o título, **<p>** para o texto e **<input>** para a caixa de e-mail.

```
<section class="contato">
  <h2 class="contato__titulo">Fique por dentro das novidades!</h2>
  <p class="contato__texto">Atualizações de e-books, novos livros,
promoções e outros.</p>
  <input type="email" placeholder="Cadastre se e-mail"
class="contato__email">
</section>
```

Em seguida, criaremos um arquivo chamado *contato.css* para a estilização da nova seção que vamos construir. O processo de criar arquivos já foi explorado em outras aulas, por essa razão, fica como tarefa criar esse arquivo sozinhos ou com os colegas. Repare na imagem a seguir o destaque na cor laranja do arquivo criado.



Em seguida, devemos realizar a importação do arquivo *contato.css* em *styles.css*. Vamos repetir o processo já feito escrevendo **@import url ("styles/contato.css");**.

```
@import url("styles/header.css");  
@import url("styles/banner.css");  
@import url("styles/carrossel.css");  
@import url("styles/topicos.css");  
@import url("styles/contato.css");
```

Em seguida, no arquivo *contato.css*, iniciaremos a estilização definindo as propriedades de cor de fundo (**background**) e espaçamento interno da seção (**padding**).

```
.contato {  
  background-color: var(--branco);  
  padding: 1em;  
}
```

O título e o texto da seção possuem a cor em degradê. Podemos utilizar o **background-clip** para aplicar o fundo nas letras, junto com suas respectivas propriedades em versão webkit para garantir uma maior compatibilidade com outros navegadores.

```
.contato__titulo,  
.contato__texto {  
    background: var(--azul-degrade);  
    -webkit-background-clip: text;  
    -webkit-text-fill-color: transparent;  
    background-clip: text;  
}
```

Como o tamanho e o peso da fonte são diferentes, vamos defini-los separadamente na classe. O título possui fonte tamanho 18 pixels e peso 500.

```
.contato__titulo {  
    font-size: 18px;  
    font-weight: 500;  
}
```

O texto possui tamanho de fonte padrão, portanto, precisamos apenas definir o seu peso em 300. Por fim, podemos definir uma margem para o alinhamento correto dos elementos.

```
.contato_texto {  
    font-weight: 300;  
    margin: 1em 0;  
}
```

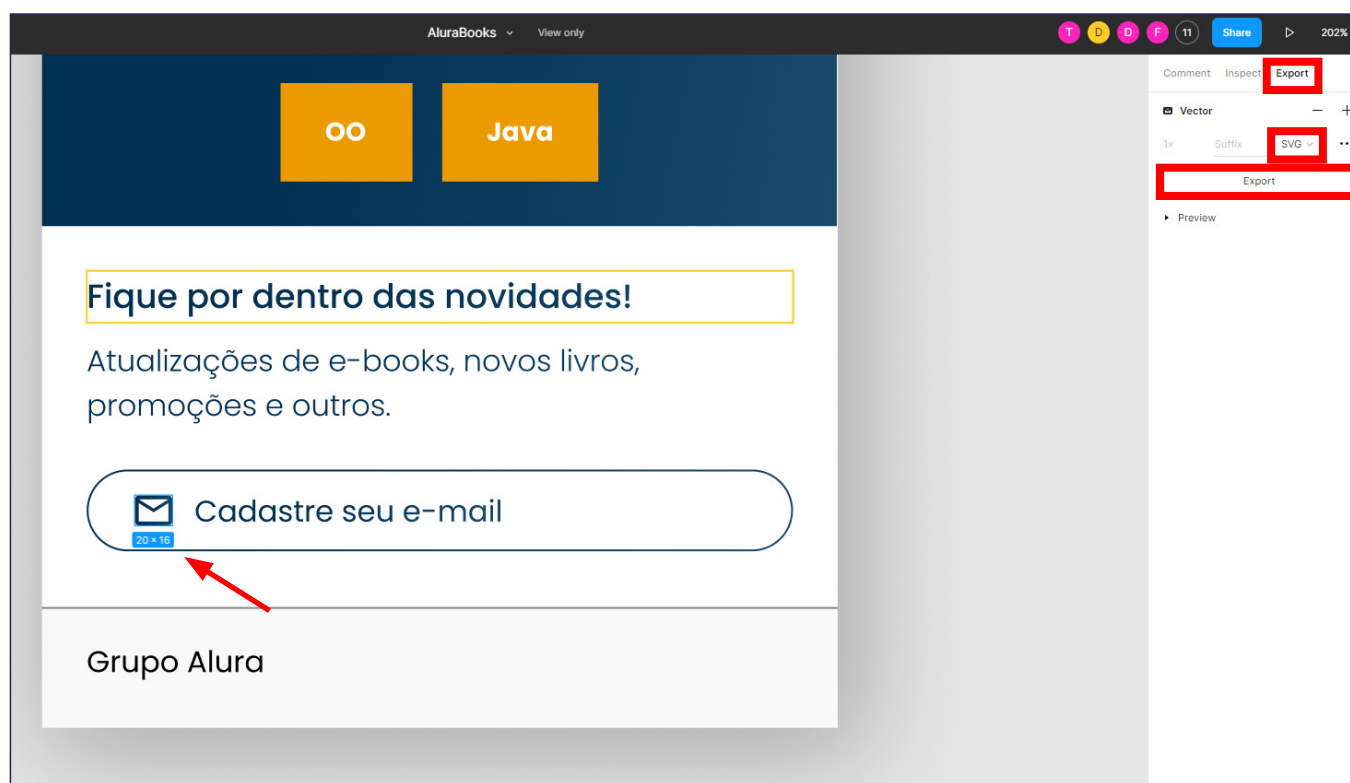
O site deverá ter agora a seguinte aparência:



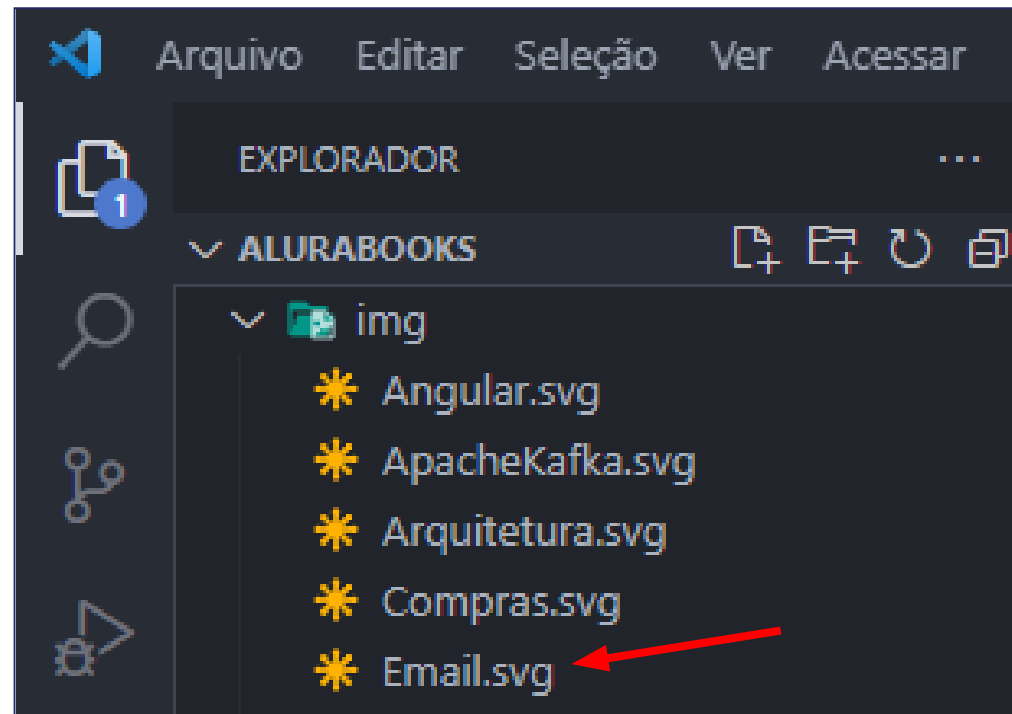
Partindo para a estilização do input, vamos definir uma margem interna, borda sólida e arredondamento e largura de 90% na cor azul, seguindo o padrão que utilizamos anteriormente no banner.

```
.contato_email {  
  padding: 1em;  
  border: 1px solid var(--azul);  
  border-radius: 24px;  
  width: 90%;  
  color: var(--azul);  
}
```

Devemos exportar o ícone de envelope do Figma. Clique neste ícone e, em seguida, em *Export*. Faça o download, nomeando e adicionando o formato “svg”, por exemplo, como *ícone_envelope.svg*.



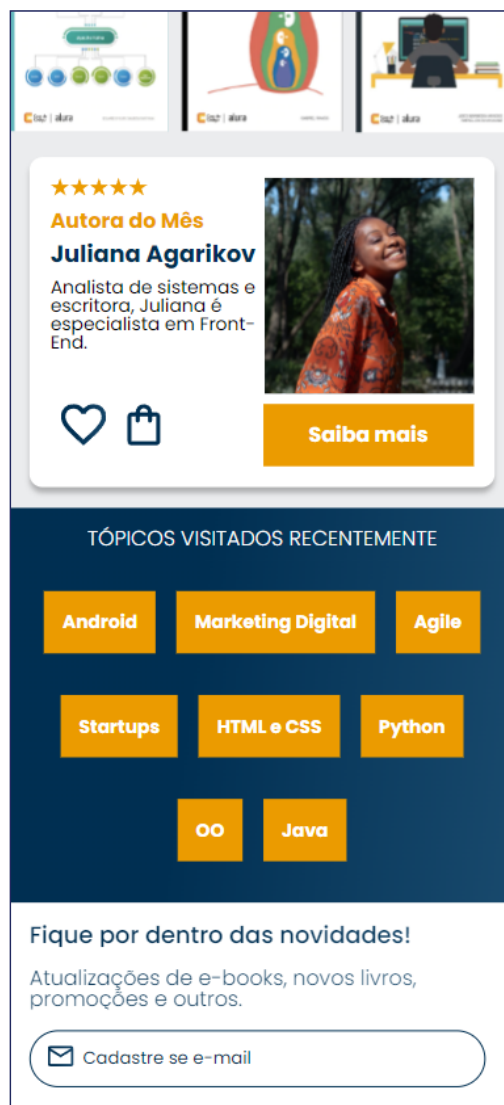
Acrescente o arquivo recém-criado na pasta “img” do projeto e renomeie para “Email”.



Por fim, vamos estilizar o **placeholder** utilizando o pseudoelemento **::** assim como fizemos anteriormente no banner, definindo a fonte, a cor e o fundo, sendo este a nossa imagem de e-mail. Para que a imagem de e-mail não fique em cima do texto, vamos adicionar uma margem interna à direita para espaçar corretamente os elementos.

```
.contato_email::placeholder {  
  font-family: var(--fonte-principal);  
  color: var(--azul);  
  background: url("../img/email.svg") no-repeat;  
  padding-left: 2em;  
}
```

O resultado, após as modificações, deve ser assim:

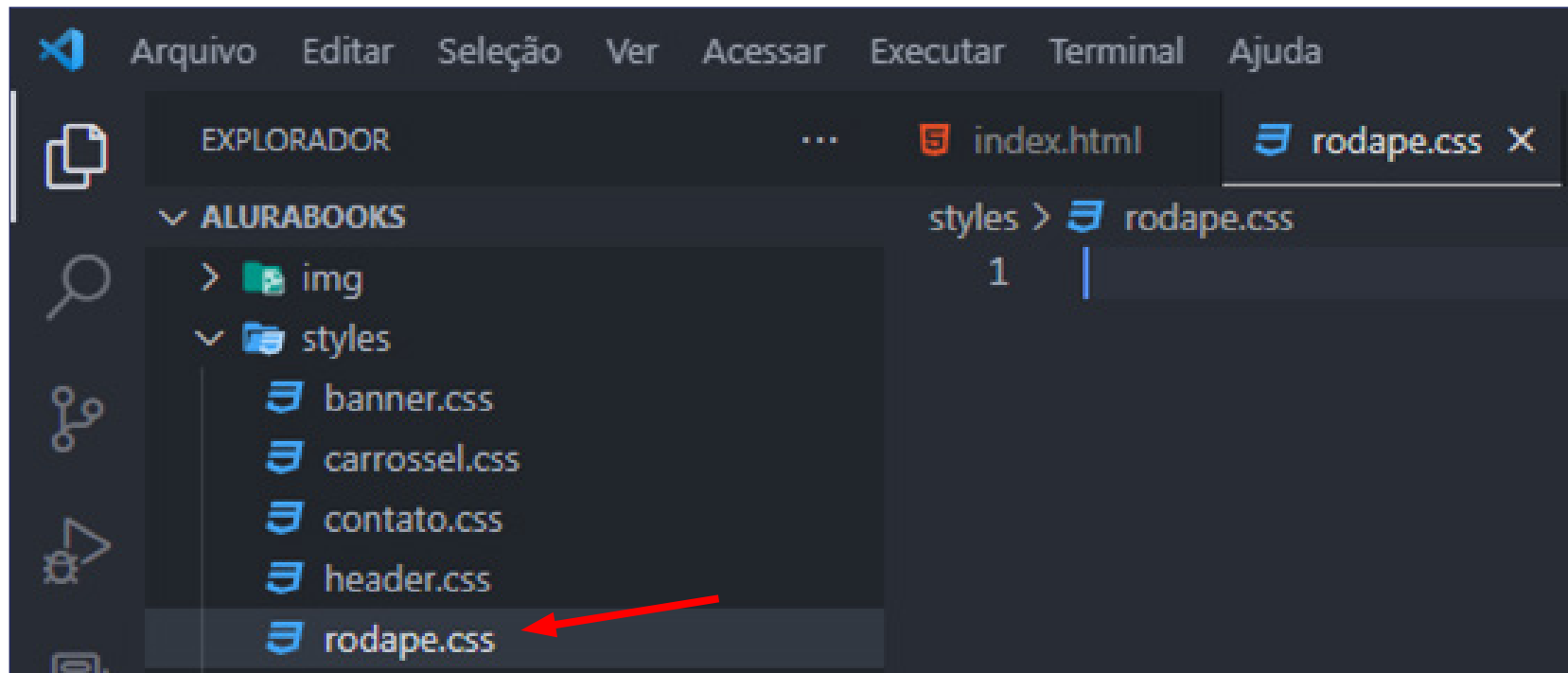


Para o rodapé, vamos utilizar a tag **<hr/>**, que cria uma linha divisória para separar os dois campos, a tag **<footer>** com classe “rodape” e a tag **<h2>** para o texto “Grupo Alura”.

```
<hr />

<footer class="rodape">
  <h2 class="rodape__titulo">Grupo Alura</h2>
</footer>
```

Criaremos um arquivo chamado *rodape.css* para a estilização do rodapé.



Vamos realizar a importação do novo arquivo *rodape.css* em *styles.css*.

```
@import url("styles/header.css");  
@import url("styles/banner.css");  
@import url("styles/carrossel.css");  
@import url("styles/topicos.css");  
@import url("styles/contato.css");  
@import url("styles/rodape.css");
```



De volta ao arquivo *rodape.css*, vamos retirar a margem da tag **<hr/>** para que ela fique colada à seção anterior.

```
hr {  
    margin: 0;  
}
```

Por fim, vamos definir a cor de fundo para o rodapé e aplicar uma margem interna.

```
.rodape {  
    background-color: var(--branco);  
    padding: 1em;  
}
```


O resultado das modificações podemos conferir a seguir.



Desafio

Nesta aula, terminamos todo o layout mobile-first do AluraBooks. Até agora vimos vários elementos que compõem o nosso código HTML, como **<header>**, **<section>** e **<footer>**. Esses elementos são chamados de elementos block-level, mas também existem outros que não foram vistos ainda. O desafio é pesquisar o que é block-level e trazer exemplos de elementos que são block-level.

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