

Aula 13

# Finalizando nossa página

Curso

**HTML e CSS: praticando**  
**HTML e CSS**

# O que vamos aprender?



Estilizar a seção **footer**.



Conhecer as pseudoclasses.

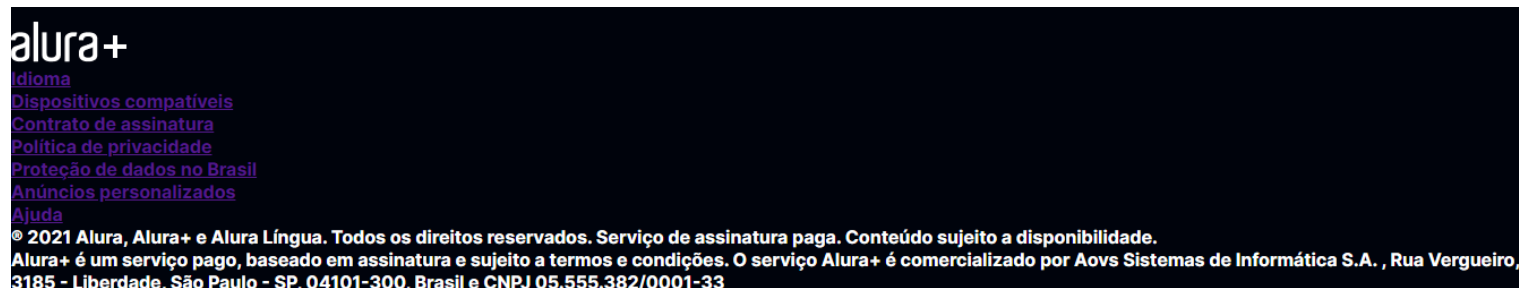


Aplicar as pseudoclasses : **hover** e : **active** nos elementos da página.



# Pseudoclasses

Na última aula, foi lançado o desafio para que fosse criada de forma autônoma a última parte que pertence à seção **footer**. O **footer** é composto pelos elementos que se encontram no rodapé do site. Utilizando as propriedades codificadas no CSS, o **footer** ou rodapé tal como mostrado na imagem a seguir deve ser estilizado para ficar mais agradável.



O resultado deveria ficar parecido com o rodapé mostrado na imagem a seguir:



Nesta aula vamos continuar construindo o **footer**.

Vamos começar alterando o display da nossa lista. No arquivo style.css vamos criar a classe **rodape\_\_lista**. Feito isso, vamos definir a propriedade **display** com o valor **flex**.

```
.rodape__lista {  
  display: flex;
```

Então, vamos centralizar todo o conteúdo desta lista, utilizando a propriedade **justify-content**.

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


Os textos desta lista não devem possuir os *bullets* (os pontinhos no início do texto). Para removê-los, utilizamos a propriedade **list-style-type** com o valor **none**.

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

E por fim, definimos uma margem para separar as imagens dos dispositivos e a nossa lista. Vamos aplicar à propriedade **margin-top** o valor **1em**. Lembrando que cada **1em** corresponde a **16 pixels**.

```
List-style-type: none;  
margin-top: 1em;
```

Ao carregarmos a página, podemos observar que nossas estilizações foram aplicadas, mas ainda existem algumas alterações para serem realizadas. Podemos perceber que os itens da lista estão na cor azul, agrupados uns nos outros e sublinhados. Vamos aprender a como modificar isso.

A screenshot of the Alura+ logo, which consists of the word "alura" in a white, lowercase, sans-serif font, followed by a plus sign "+". The logo is set against a dark blue background.A screenshot of the Alura+ website footer, showing a horizontal list of links in a small, blue, sans-serif font. The links are: "Idioma", "Dispositivos compatíveis", "Contrato de assinatura", "Política de privacidade", "Proteção de dados no Brasil", "Anúncios personalizados", and "Ajuda".

Ainda no elemento lista, precisamos fazer algumas alterações. Vamos combinar a classe **.lista\_\_link** com a tag **<a>**. Isso significa que todo elemento **<a>** que possuir a classe **.lista\_\_link** receberá tais alterações. E quais seriam elas? A primeira delas é retirar o sublinhado dos itens. Para isso, vamos aplicar o valor **none** à propriedade **text-decoration**.

```
.lista__link a{  
text-decoration: none;
```

Agora vamos definir a cor dos elementos do rodapé, que deverá seguir um dos padrões de cores já criados.

```
text-decoration: none;  
color: var(--branco-principal);
```

O próximo passo é criar um espaçamento externo entre os elementos para que eles não fiquem grudados. Para isso, vamos utilizar a propriedade **margin** com os valores 0, 16 pixels, 0 e 16 pixels. Para espaçamento maior ou menor entre os textos, pode-se estilizar da maneira que desejar.

```
Color: var(--branco-principal);  
margin: 0 16px 0 16px;
```

E por fim, vamos definir o tamanho da fonte utilizando a propriedade **font-size** com o valor de 14 pixels.

```
margin: 0 16px 0 16px;  
font-size: 14px;
```

Ao carregarmos a página no navegador, podemos observar que os estilos do CSS foram aplicados. A lista de elementos está na cor branca e eles estão separados.

Idioma Dispositivos compatíveis Contrato de assinatura Política de privacidade Proteção de dados no Brasil Anúncios personalizados Ajuda

Dando continuidade à estilização, vamos alterar as propriedades dos parágrafos. Para isso, vamos criar a classe **.rodape\_texto**. Definimos margem para cima e para baixo com o valor de **1em**, enquanto as demais margens terão valor **0**.

```
.rodape_texto {  
margin: 1em 0;
```

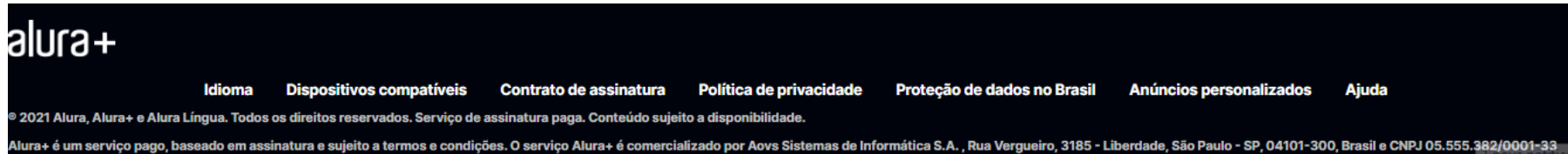
Aplicamos a cor **--cinza-secundario** aos textos.

```
margin: 1em 0;  
color: var(--cinza-secundario);
```

E por fim, definimos o tamanho da fonte com valor **12px**.

```
color: var(--cinza-secundario);  
font-size: 12px;
```

Ao carregarmos a página, podemos observar que o CSS foi aplicado.



Agora, precisamos centralizar todos os elementos da seção **footer**, para isso, utilizamos a **classe .rodape**.

```
.rodape {  
  text-align: center;
```

Precisamos definir também os espaçamentos entre os elementos. Neste caso, podemos estilizar, usando **5em** (para cima e para baixo) e **3em** (para os lados).

```
  text-align: center;  
  margin: 5em 3em;
```



Ao carregarmos a página no navegador, podemos observar que nossa seção está finalizada.



Vamos agora aprender mais estilizações utilizando as pseudoclasses.

Até aqui, as estilizações que aplicamos aos elementos da página Alura Plus são fixas, ou seja, ao recarregarmos a página no navegador, todas as cores, tamanhos e decoração já estão aplicados. Também podemos aplicar estilizações quando alguma ação ocorrer, por exemplo, quando o usuário clicar ou passar o cursor do mouse por cima de um elemento. Isso é possível por meio das pseudoclasses. Uma pseudoclasse adiciona um comportamento ao elemento e existem diversos tipos delas. A propriedade do elemento é alterada apenas quando um determinado comportamento ou ação acontece. Vamos à pseudoclasse **:hover**.

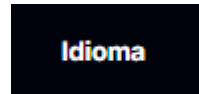
A pseudoclasse **:hover** define que a estilização é aplicada quando o usuário passar o cursor do mouse por cima de um determinado elemento da página definido por nós. Vamos aplicar essa pseudoclasse em cada elemento da tag **<a>** de nossa lista. Para aplicarmos, antes de escrevermos o nome da pseudoclasse, adicionamos dois-pontos.

```
.lista_link a:hover {
```

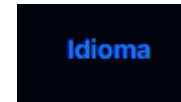
Feito isso, vamos definir que, sempre que o usuário passar o cursor do mouse por cima do elemento, sua cor deverá tornar-se azul.

```
    color: var(--botao-azul);  
}
```

Sem o cursor do mouse:



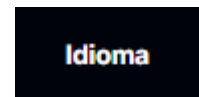
Com o cursor do mouse:



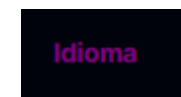
Podemos aplicar uma cor diferente quando o usuário clicar no elemento. Para isto, usamos a pseudoclasse **:active**. Veja que para aplicar essa pseudoclasse sempre precisamos adicionar dois-pontos junto ao nome dela.

```
.lista_link a:active {  
  color: purple;  
}
```

Antes do clique:



Durante o clique:



A cor só é aplicada no momento do clique ou enquanto o usuário segurar o botão esquerdo do mouse.

Agora que já aprendemos sobre as pseudoclasses, podemos utilizá-las em outros elementos da nossa página. Vamos estilizar os botões.



Definiremos que, quando o usuário passar o mouse por cima do botão, a cor da fonte será alterada da cor branca para a cor preta e a cor de fundo do botão ficará em um tom mais claro de azul.

O primeiro passo é definir a nova cor de fundo do botão. Para isso, no arquivo style.css, vamos em **:root** para criar uma variável para armazenar o novo tom de azul.

```
--botao-azul-efeito:
```

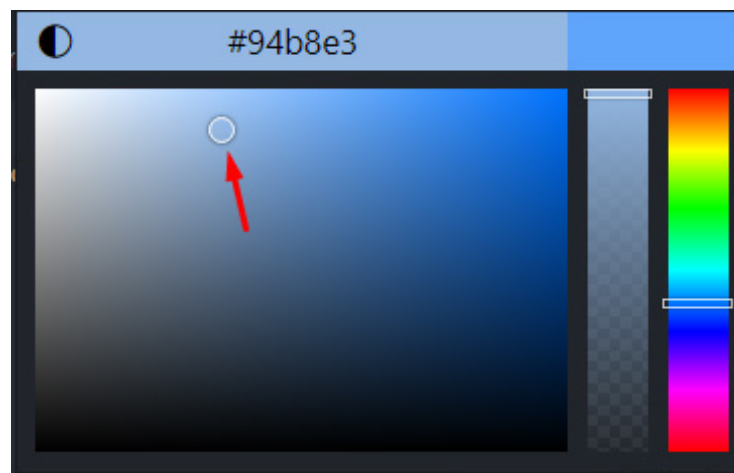
Então, vamos copiar o mesmo valor da variável **—botao-azul** na nova variável.

```
--botao-azul: #167BF7;  
--botao-azul-efeito #167BF7;
```

Feito isto, vamos passar o cursor do mouse em cima do quadrado da cor da variável **—botao-azul-efeito**, tal como indicado pela seta na imagem a seguir.



E então, arrastar a bolinha para a esquerda da tela para obter tons mais claros. Você pode escolher o de sua preferência.



Feito isso, o tom escolhido será aplicado à variável.

```
--botao-azul-efeito:#96b7e0;
```

Vamos aplicar essa nova cor ao botão; para isso, vamos utilizar a classe **.container\_botao** e a pseudo-classe **:hover**.

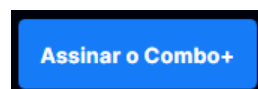
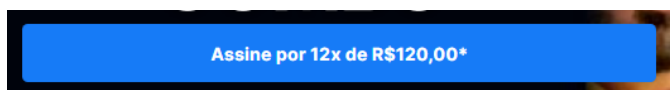
```
.container_botao:hover{  
background-color: var(--botao-azul-efeito);  
}
```

Além da cor de fundo, vamos alterar a cor da fonte, que será preta. Já temos uma variável que armazena a cor preta.

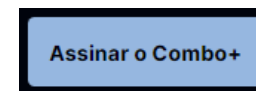
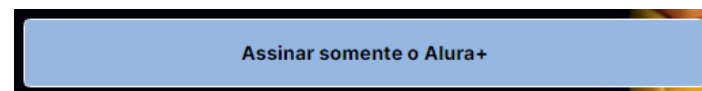
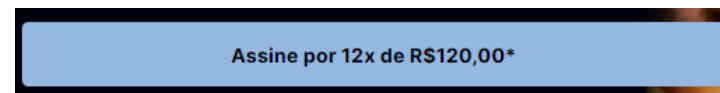
```
background-color: var(--botao-azul-efeito);  
color: var(--cor-de-fundo);
```

Ao carregarmos a página, podemos observar que o CSS foi aplicado.

Antes do cursor do mouse:



Depois do cursor do mouse:



É possível aplicar pseudoclasses em qualquer tipo de elemento da página da web, criando, assim, diferentes estilizações em nossa página. Com isto, concluímos o desenvolvimento da página Alura Plus. Na próxima aula aprenderemos a compartilhar o nosso projeto.

# Desafio

Nesta aula aprendemos uma nova forma de estilização do CSS ao usar as pseudoclasses. Existem inúmeras pseudoclasses que podem ser aplicadas nos elementos de uma página da web. O desafio será a realização de uma breve pesquisa sobre outras pseudoclasses. Depois de pesquisar na internet, selecione duas pseudoclasses para aplicar e testar no arquivo style.css da página Alura Plus.