

02

## Limitações do placeholder

### Transcrição

Vimos como funcionam os placeholders , e agora podemos aplicar esse conhecimento para transformar o *mixin* de border-radius em placeholder :

```
%borda-redonda($raio: 10px) {  
  -webkit-border-radius: $raio;  
  border-radius: $raio;  
}
```

Como no *mixin*, manteremos a variável \$raio para definir um valor padrão para o raio (de 10 px), e ainda poder alterar esse valor para o botão da área de destaque, que deve possuir o valor de 50px nessa variável.

Voltaremos a comparar os dois arquivos na tela do editor de texto, usando o atalho "CTRL + ALT + 2". Quando salvamos o .scss , arquivo compilado apresenta o seguinte erro:

```
Error: Invalid CSS after "%borda-redonda": expected selector, was "($raio: 10px)"  
on line 21 of helpers/mixins.scss  
from line 2 of estilos.scss
```

Parece haver algo de errado com a variável \$raio . Quando deletamos o "10px", o erro persiste. Precisamos dessa variável para quando o `borda-redonda` for chamado, para podermos escolher entre manter o valor padrão ou definir outro valor.

Mas o placeholder não funciona da mesma maneira que um *mixin*. Atualmente, ele não suporta variáveis, apenas copia e cola literalmente o que é definido do placeholder para o arquivo .css . Então, para o caso da `borda-redonda` , é mais proveitoso manter o *mixin* e deletar o placeholder que acabamos de criar.

Tanto o *mixin* como o placeholder são possibilidades para evitar repetições de código no Sass, mas o placeholder apenas é uma alternativa para o *mixin* quando lidamos com parâmetros fixos. Quando é preciso passar algum valor na chamada da função, o *mixin* é mais recomendado.