

## Criando a extensão da API String

Agora que implementamos o código que limita a categoria de acordo com a quantidade de caracteres, precisamos realizar o processo para tirar essa responsabilidade do adapter, ou seja, o mesmo procedimento que fizemos quando formatamos o valor e a data.

### Extraindo a função que limita caracteres

Para isso, primeiro crie a extension function `String.limitaEmAte()` recebendo como parâmetro o valor `caracteres: Int` para indicar a quantidade de caracteres que deseja limitar, então, retorne uma `String` que vai ser a `String` limitada.

Em seguida, pegue todo o código que faz a formatação da categoria e adicione para dentro da função `String.limitaEmAte()`.

### Adaptando a extension function

Dentro da extension function substitua todo o código referente à variável `categoriaFormatada` para o objeto `this`, pois agora o próprio objeto da `String` será utilizado como referência para realizar a formatação.

Note que deste jeito não é mais necessário o uso do objeto `transacao` para pegar o valor da `categoria`, pois ela mesmo chamará a função.

Depois de modificar as referências, faça um *early return*, ou seja, dentro do `if` devolva o String Template que adiciona os `..."` e, depois do `if`, retorne o `this`.

Também, modifique o número mágico `14` pelo parâmetro `caracteres`, pois é a partir dele que a função vai limitar a `String`.

### Extraindo a property para indicar o limite de caracteres

Após a adaptação, no momento que for atribuir o valor do componente `transacao_categoria` atribua com a extension function `String.limitaEmAte()` enviando o valor de limite de caracteres, no caso, `14`.

Inclusive, aproveite e já crie a property apenas de leitura (`val`) e privada (`private`) `limiteDaCategoria` e a initialize com o valor `14`, então, a envie via parâmetro no momento da chamada.

### Isolando a extension function

Após a adaptação crie o arquivo Kotlin `StringExtension` no pacote `extension` e mova a função para ele, em seguida, no Adapter, faça o import da função.

Por fim, execute a App e veja se apresenta o mesmo aspecto.

