



## Criando a classe Transacao

Caso você não tenha o projeto com o conteúdo da aula passada, você pode baixá-lo [neste link \(https://github.com/alura-cursos/financas-kotlin/archive/aula-1.zip\)](https://github.com/alura-cursos/financas-kotlin/archive/aula-1.zip).

Atualmente representamos as nossas transações a partir de uma lista de String, entretanto, cada transação possui campos que representam, por exemplo, valor, categoria, data e ícone...

Pensando justamente em obter essas informações, precisamos criar uma estrutura que nos permita armazenar tais informações de uma maneira objetiva. Ou seja, vamos criar uma classe para representar uma transação.

### Criando a classe para representar uma transação

Para isso crie a classe `Transacao` dentro do pacote `br.com.alura.financask.model`. Em seguida, adicione os atributos:

- `valor` do tipo `BigDecimal`
- `categoria` do tipo `String`
- `data` do tipo `Calendar`

### Recebendo parâmetros via construtor

O Kotlin exige que todas as variáveis ou atributos criados sejam inicializados, ou seja, faça com que todos os campos sejam enviados via construtor, e então, atribua o valor de cada um dos atributos.

**Boa prática:** lembre-se que quando adicionamos mais de um parâmetro via construtor, a boa prática é que pulemos uma linha para cada um deles.