



Aplicando a inicialização atrasada

Caso você precisar do projeto com os ajustes realizados na aula anterior, você pode baixá-lo por meio [deste link](https://github.com/alura-cursos/financas-kotlin-parte3/archive/aula-2.zip) (<https://github.com/alura-cursos/financas-kotlin-parte3/archive/aula-2.zip>).

Vimos que quando lidamos com variáveis que podem receber valores `null`, temos diversos detalhes que precisamos nos atentar. Pensando justamente em evitar esse tipo de situação, vamos aplicar uma técnica na qual permite inicializarmos uma property em um outro momento sem a necessidade de atribuir como valor padrão o `null`.

Inicialização atrasada com `lateinit`

Pensando justamente em permitir esse tipo de possibilidade, o Kotlin nos fornece um recurso conhecido como inicialização atrasada e, para realizar esse tipo de operação, adicione a keyword `lateinit` na `viewDaActivity`. Repare que não é possível utilizar tipos que podem receber `null`, portanto, remova a possibilidade da variável receber valor `null`.

Em seguida, realize a inicialização da property. A inicialização pode ser feita em qualquer momento logo depois de chamar a função `setContentView()`, porém, antes de usar a property.

Removendo todo o tratamento de `null`

Depois da modificação, modifique o código para que não precise mais lidar com valores que podem ser `null`, em específico, dentro da classe `ResumoView`.

Por fim, teste a App e veja se funciona como esperado.

Observação: É válido lembrar que se a inicialização não for feita e a property for utilizada, ainda ocorre um erro! Porém, não um NPE e sim um detalhe de falta de inicialização.