Declarando Variáveis em Kotlin

Variável é um mecanismo das linguagens de programação que facilita colocar, remover, acessar e modificar um ou mais dados na memória do computador.

val

Essa é a forma mais comum e recomendada para declarar uma variável. Com val a variável recebe valor uma única vez, podendo ser utilizada somente para a leitura de valores.

Exemplo de variável do tipo somente leitura:

```
val precoproduto Float = 4.99f
```

A sintaxe para declaração de variáveis usa pascal notation, onde informamos `nome : tipo`.
No **`Exemplo1`** `precoproduto` é o nome da variável e `Float` o seu tipo.

Toda variável declarada com val deve ser iniciada com um valor, que não poderá ser modificado. Caso se tente alterar o valor de uma variável declarada com val, como no Exemplo**2**, um erro de compilação será emitido.

```
val precoproduto: Float = 4.99f

preco = 5.49f
```

Erro de compilação porque o valor de val não pode ser reatribuído

Ao tentar modificar o valor de uma variável declarada com val receberemos um erro de compilação com a mensagem "`Val cannot be reassigned`."

var

Caso uma variável deva mudar de valor, usamos var em sua declaração, como mostra o Exemplo**3**.

```
var preco: Float = 1.99f
```

Ao contrário de val, var não requer que a variável seja iniciada com um valor. O Exemplo**4** contém um exemplo dessa sintaxe.

```
var preco: Float
```

Exemplo 4. Declaração de uma variável com var com omissão de valor

Embora possamos omitir o valor quando usamos var, o tipo da variável ainda é obrigatório nesse caso. Um erro de compilação é emitido caso tipo e valor estejam ausentes na declaração de uma variável (Exemplo **5**).

```
var preco
```

Exemplo 5. A omissão de tipo e valor gera um erro de compilação

O primeiro erro possui a mensagem This variable must either have a type annotation or be initialized, que significa a ausência de valor e tipo de valor. A segunda mensagem de erro é Variable 'preco' must be initialized, que diz respeito a ausência de um valor especificamente, porque se um valor está presente o compilador consegue inferir o tipo de uma variável.

Inferência de tipo

Os tipos de dados em Kotlin serão detalhados em um outro momento, mas veremos aqui uma regra de sintaxe que possui conexão com esse assunto.

O tipo de uma variável pode ser omitido quando tivermos declaração e atribuição de valor em uma mesma instrução, como no Exemplo **6**.

```
val precoproduto = 6.99f
```

Exemplo 6. Declaração de variável com inferência de tipo

O mesmo é válido para var (Exemplo **7**):

```
var precoproduto = 2.99f
```

Exemplo 7. Declaração de variável com inferência de tipo

Note que só foi possível omitir o tipo de dado porque a atribuição de valor foi feita junto com a declaração da variável. Nesse caso, o tipo da variável é inferido de acordo com o valor que ela recebeu.

Nível elevado

A linguagem Kotlin suporta a declaração de variáveis de nível elevado (top-level), imediatamente dentro de um pacote. Pacotes são um assunto que veremos futuramente, mas por enquanto podemos dizer que em Kotlin é possível declarar uma variável fora de uma função ou classe, dessa forma:

```
val precoproduto = 2.99f

fun main() {
    val desconto = 0.3f

    println(precoproduto - precoproduto * desconto)
}
```

Exemplo 8. Declarações de variáveis de nível elevado

Nesse exemplo, a variável precoproduto está no nível mais elevado e a variável desconto é local porque está dentro da função main. A variável de nível elevado precoproduto pode ser utilizada em qualquer lugar no projeto, inclusive em outros arquivos, enquanto a variável local desconto pode ser utilizada somente dentro da função onde foi declarada.

Caso uma variável local, declarada dentro de uma função, tenha o mesmo nome de uma variável de nível elevado, a variável de nível elevado será ofuscada dentro da função.

No Exemplo **9** declaramos uma variável de nível elevado chamada precoproduto e depois uma variável local de mesmo nome dentro da função main.

```
val precoproduto = 2.99f

fun main() {
    val precoproduto = 4.99f
    val desconto = 0.5f

    println(precoproduto - precoproduto * desconto)
}
```

Exemplo 9. Variável de nível elevado ofuscada por uma variável local

Nesse exemplo, o valor exibido será 2.495, uma vez que para a função main a variável preco de nível elevado e valor 2.99f não existe, pois ela foi ofuscada pela variável preco local de valor 4.99f.

Por Hoje é isso aí pessoal, continuem firmes nos estudos, pois nada pode impedir uma pessoa realmente focada e dedicada de alcançar seus objetivos.

Sucesso nos estudos!