

Desenvolvimento de Aplicativos – Tarefa 1_04

Prof. Dr. Aparecido V. de Freitas – Ciência da Computação

Kotlin não é muito diferente de **Java**. **Kotlin** introduziu algumas novas funcionalidades de linguagem, mas há mais semelhanças do que diferenças entre **Kotlin** e **Java**. Para o programador **Java**, a curva de aprendizado para **Kotlin** não é íngreme.

Em **Kotlin**, variáveis são declaradas por meio de um identificador usando a palavra-chave **var** seguida pelo seu tipo. Variáveis também podem ser declaradas com a palavra-chave **val**. Neste caso podem ser inicializadas apenas uma vez dentro do bloco de execução onde foram definidas. Isso as torna efetivamente constantes e equivalem à palavra-chave **final** em **Java**.

Uma **expressão** em **Kotlin** combina operadores, funções, valores literais, variáveis ou constantes e sempre é avaliada para um valor. Expressões também podem fazer parte de uma expressão mais complexa. Exemplo: $A = (5 + 3)$. A expressão $(5 + 3)$ resulta no valor 8.

Declarações podem conter **expressões**, mas por si só, **não** são avaliadas para um valor. **Declarações** são instruções que definem **variáveis**, **funções**, **classes** ou outros elementos do código. Diferentemente de **expressões**, que sempre retornam um valor, as **declarações** servem para estruturar o programa e **não produzem um valor** diretamente. O código abaixo representa a declaração de uma variável chamada idade:

```
var idade = 30;
```

Em **Kotlin**, atribuições são operações onde um **valor** é atribuído a uma variável. A atribuição é fundamental em qualquer linguagem de programação, pois permite que se armazene e se manipule dados em variáveis. O operador de atribuição em **Kotlin** é o sinal de igual (=).

Em **Java**, as atribuições podem retornar um valor. Quando se realiza uma atribuição, o valor da atribuição é o valor que foi atribuído à variável. Isso permite que atribuições sejam usadas em expressões maiores ou mesmo em outras atribuições.

Empregando a **IDE Eclipse**, codifique o código abaixo em **Java**:

```
package br.uscs;

public class Main {

    public static void main(String[] args) {
        int x = 4 , a = 15, b = 2, y = 8;

        while ( ( x = a % b ) != 0 ) {
            y = y + 1;
            b = y + x;
            a = a - 1;
        }
        System.out.println ("y = " + y);
    }
}
```

Ao executar o programa acima em Java, o que será exibido na console ?

Resposta: y = 23

Usando a **IDE VSCode**, reescreva o código acima em **Kotlin**, conforme abaixo:

```
fun main() {
    var x = 4
    var a = 15
    var b = 2
    var y = 8

    while ( (x = a % b ) != 0) {
        y += 1
        b = y + x
        a = a - 1
    }

    println("y = $y")
}
```

Explique abaixo a razão pela qual o código **Kotlin** apresenta erros de compilação:

O código Kotlin irá apresentar erros de compilação pois por padrão, o Kotlin não retorna valor em atribuições como o Java retorna. Dessa forma, o trecho "`(x = a % b) != 0`" impede que o código seja executado corretamente.