

Aula 5 – Introdução ao Kotlin

Programação Mobile
Professor Bruno Maion

Kotlin

- Linguagem de programação que combina os paradigmas de Orientação a Objetos e Programação Funcional
- Foco em interoperabilidade, segurança, clareza e suporte a ferramenta
- Surgiu em 2011 durante o JVM Language Summit em que a JetBrains revelou que estava trabalhando havia quase um ano no projeto Kotlin.
- A ideia era criar uma nova linguagem estaticamente tipada para a JVM.
- INTEROPERABILIDADE com Java,
 - significa que podemos utilizar códigos que foram escritos em Java no nosso código em Kotlin

JAVA

```
Button bt_login = (Button) findViewById(R.id.bt_login);  
bt_login.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        efetuarLogin();  
    }  
});
```

KOTLIN

```
bt_login.setOnClickListener{  
    efetuarLogin()  
}
```

Kotlin – Tipos de dados

Double	Tipo numérico para valores de ponto flutuante, com Double é possível guardar valores com precisão bem grande pois ele ocupa 64 bits
Float	Tipo numérico para valores de ponto flutuante, com a diferença de que tem uma precisão menor que o Double pois ele ocupa 32 bits
Long	Usado para valores inteiros, a diferença dele para o Int é que, como ele ocupa 64 bits, você consegue representar números muito maiores
Int	Talvez um dos tipos numéricos mais comuns, é para guardar números inteiros com uma precisão de 32 bits
Short	Também guarda valores inteiros, com capacidade máxima de metade de um Int , pois ocupa somente 16 bits
Byte	Também guarda números inteiros, porém seu número máximo é 127 pois ele ocupa somente 8 bits
String	É um tipo para guardar valores em texto, uma frase, um nome ou qualquer valor que seja um texto
Char	É um tipo para guardar um único caractere, diferente da String , em que você pode guardar textos
Boolean	É usado para valores booleanos, ou seja, valores que são verdadeiro (True) ou falso (False)

Kotlin

- Em **Kotlin**, tudo é um objeto, diferentemente de Java, em que existia uma diferenciação de tipo de dados primitivos e objetos.
- Na prática, isso quer dizer que todas as variáveis e tipos que usamos em Kotlin possuem **propriedades** e **métodos**, isto é:
 - todo objeto possui características e comportamentos específicos.
- Um exemplo são os métodos de conversão de tipos.
 - Imagine que você tenha uma variável do tipo **Int** e precise convertê-la para **Double** , ou **Float** , ou até mesmo para **String** .
 - `valor.toDouble()` , `valor.toFloat()`, `valor.toString()`

Kotlin

O Kotlin consegue inferir o tipo automaticamente com base no valor atribuído.

- `val` → cria uma variável imutável (não pode ser alterada depois de atribuída).
- `var` → cria uma variável mutável (pode ser alterada depois).

```
fun main() {  
    val idade = 20      // Kotlin entende que é Int  
    var nome = "Bruno"  // Kotlin entende que é String  
  
    println(idade)  
    println(nome)  
}
```

Kotlin

Se você quiser deixar explícito o tipo da variável, usa-se : depois do nome da variável: `val` → cria uma variável imutável (não pode ser alterada depois de atribuída).

```
fun main() {  
    val idade = 20      // Kotlin entende que é Int  
    var nome = "Bruno"  // Kotlin entende que é String  
  
    println(idade)  
    println(nome)  
}
```

Teste em Kotlin

- Acesse o TryKotlin para realizar experimentos;
- <https://try.kotlinlang.org>



Obrigado!