



[MB-KT-004]

Técnicas de Programação (kotlin)

Sobre o Professor

Guilherme Taveira Berson



Guilherme Taveira **Berson**

25 anos

Palmas - Tocantins

Engenheiro Mobile Sênior
Zup/Itaú

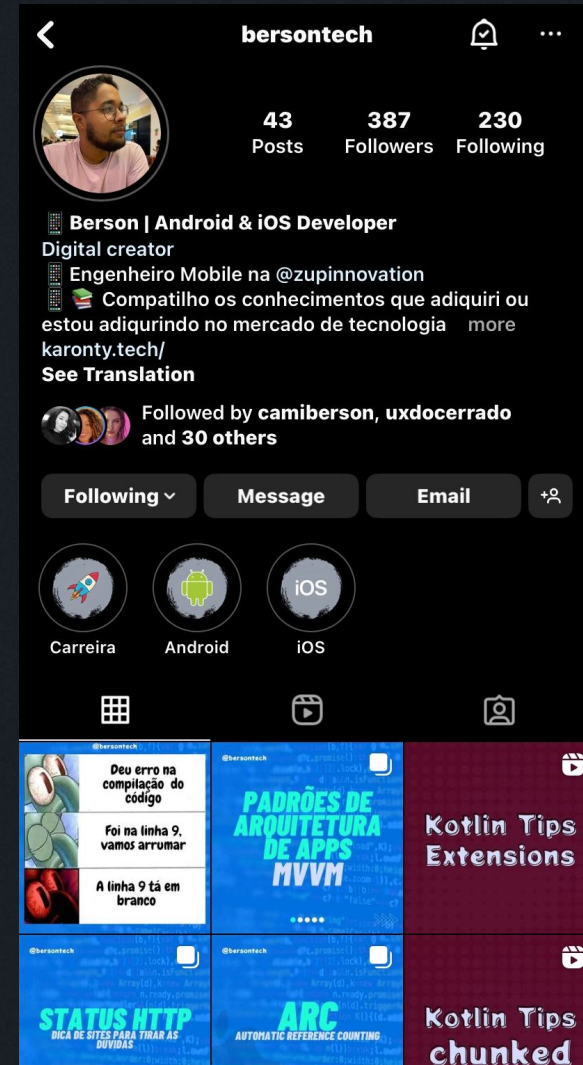


Sobre o Professor

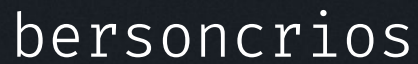
Guilherme Taveira Berson



bersontech



Guilherme Taveira Berson



Sobre o Professor

Victor Magosso



Victor Magosso

26 anos

São Paulo capital

Engenheiro Android Sênior
Invillia/iFood

invillia
a compass.uol company



Sobre o Professor

Victor Magosso



victormagosso



Sobre o Professor

Victor Magosso



VictorMagosso

The screenshot shows the GitHub profile of Victor Magosso. At the top, there are navigation links: Product, Solutions, Open Source, and Pricing. A search bar and links for Sign in and Sign up are on the right. The profile header includes a circular profile picture of a man with a beard, the name Victor Magosso, the username VictorMagosso, and a Follow button. Below the name is a bio: "To solve problems - That's why developers exist for, that's false" and statistics: 14 followers and 18 following. The main content area is titled "Pinned" and displays five repositories:

- ToDoList-JUnit-RoomDB** (Public): Android -> Setup de testes de um ToDo list. Languages: Kotlin.
- DividendMine** (Public): Flutter App to control your dividends. Language: Dart.
- SquareGame** (Public): Training provider with a math game. Language: Dart.
- Ecommerce_JAWAWE** (Public): Spring Boot - Ecommerce loja de peças para carros. Languages: HTML, Kotlin.
- Ecommerce-Magosso** (Public): Backend ecommerce -> Spring Boot + Kotlin. Language: Kotlin.

At the bottom, it shows "35 contributions in the last year" with a calendar icon for 2023.

Vamos conhecer a turma ?



Técnica de Programação - Kotlin

#1	Introdução à Programação Funcional: Porque utilizar Programação Funcional?
#2	Entendimento sobre corrotinas e como implementar (Parte 1): Trabalhando com fluxo de dados
#3	Entendimento sobre corrotinas e como implementar (Parte 2): Trabalhando com fluxo de dados assíncronos
#4	Extension: Entendendo funções de extensão
#5	Regex: Manipulando dados com regex

Técnica de Programação - Kotlin

#6	Boas práticas em Kotlin (Parte 1): Criando código convencional
#7	Boas práticas em Kotlin (Parte 2): Criando código convencional
#8	Revisão de tudo que foi visto nas aulas anteriores
#9	Aula dedicada à devolutiva da avaliação por rubrica / autoavaliação.

O que é paradigma de programação?

Nada mais é do que a **maneira de fazer algo**, ou seja, a forma como se programa.

Existem **diversos paradigmas**, dentre os mais famosos temos:

- Programação orientada a objetos
 - Programação Imperativa
 - **Programação Funcional**

O que é paradigma de
programação?



Programação Orientada a Objetos

É um paradigma de programação baseado no conceito de "objetos", que podem conter dados na forma de campos, também conhecidos como atributos, e códigos, na forma de procedimentos, também conhecidos como métodos.

Programação Imperativa

Na programação **Imperativa**, ou **Procedural**, o foco é em **descrever detalhadamente o passo a passo de um processo que a máquina deve executar**, ou seja, as instruções são passadas por meio de códigos na sequência que devem ser executadas.

Programação Funcional

Ela trabalha com códigos compostos de **múltiplas funções**, funcionando de forma unificada para resolver problemas. Esse paradigma tem base no modelo computacional de Cálculo Lambda, criado em 1930 pelo matemático estadunidense Alonzo Church.

Programação Funcional

É um paradigma de programação no qual a computação é tratada como a validação de uma sequência de **funções puras** — que **criam, manipulam e retornam dados imutáveis** — com o objetivo de resolver um determinado problema. Ele é um estilo de programação declarativa.

Programação declarativa = “o que resolver”

Estilo imperativo = “como resolver”

Uma parte importante da programação Funcional ...

É a **imutabilidade**, e ela significa que os dados **não podem ser alterados após terem sido criados e/ou atribuídos a uma propriedade**. Através da imutabilidade é possível garantir que uma determinada propriedade possua sempre o mesmo valor durante toda a execução do programa.

Uma parte importante da programação Funcional ...

VAL → Imutabilidade referencial, porém, não de valor, ou seja, **não podemos associar um novo valor a propriedade**

VAR → Permite associar um novo valor interno a propriedade

Programação Funcional



#showmethecode



```
fun main() {  
    val name = "guilherme"  
    println(name)  
  
    name = "berson"  
    println(name)  
}
```

esse código compila ?
Qual o resultado?

#showmethecode



```
fun main() {  
    var name = "guilherme"  
    println(name)  
  
    name = "berson"  
    println(name)  
}
```

esse código compila ?
Qual o resultado?

Imutabilidade significa ... previsibilidade

Quando um programa é desenvolvido usando os princípios da PF, ele se torna previsível.

Se eu sei que, ao invocar essa função, eu terei um **número de retorno**, eu sei que **irei lidar com números**. E se eu for passar para um **outra função um número**, mas ela **espera uma string**, eu já logo **acuso um erro**. Isso faz com que seu programa se torne mais inteligente

Funções, Funções e mais funções

Funções Puras

As **funções puras** são funções que **não possuem qualquer tipo de efeito colateral**, e portanto, **dado uma entrada**, ela sempre vai retornar o **mesmo valor para essa mesma entrada**.

Funções, Funções e mais funções

Funções Puras

- Recebe ao menos um parâmetro e trabalha com ele.
- Ela retorna alguma coisa

Funções, Funções e mais funções



**“Talk is
cheap. Show
me the code.”**

Linus Torvalds

Funções, Funções e mais funções

Funções Puras

Vamos calcular quantos kWh um equipamento elétrico gasta usando funções ?

$$\text{consumo(kWh)} = (\text{potência(W)} * \text{horas(h)} * \text{dias}) / 1000$$

Aspirador de Pó	600
Computador	300
Fogão Elétrico de 4 Bocas	6000
Espremedor de Frutas	200
Impressora	45
Televisor 21"	90

Funções, Funções e mais funções

Funções Puras

Vamos calcular quantos é a média de duas notas em uma função e informar se o aluno está aprovado se a mesma for maior que 7 ou reprovado se for menor.

Desafio, usar uma função para calcular a média e uma segunda para pegar o valor da mesma e verificar se passou

Funções, Funções e mais funções

Function Types

Em Kotlin, as funções são **first-class citizens**, isso significa que elas são tratadas como um tipo da linguagem, assim como Int, String, Long, Boolean, etc.

Quando elas são representadas como um tipo, elas são chamadas de ***function types***.

Funções, Funções e mais funções

Function Types



```
val NomeDaPropriedade: (TypeParam, TypeParam ...) -> TypeFunc = {  
  Bloco de código  
}
```

Funções, Funções e mais funções

Function Types

Podemos ter Functions Type:

- Com um parâmetro
- Com diversos parâmetro
- Sem parâmetro
- Sem Retorno (Unit)

Funções, Funções e mais funções

Function Types



```
val discountCode: () -> String = { "500FF" }
```



```
val square: (Int) -> Int = { it * it }
```


Funções, Funções e mais funções

Function Types



```
val sum: (Int, Int) -> Int = { x, y -> x + y }
```



```
val saySomething: (String) -> Unit = { input -> println(input) }
```

Funções, Funções e mais funções

HOF – Funções de primeira Ordem

É aquela em que suas funções podem ter outras funções como parâmetros e retornar ou não, elas como resultado.

Uma das formas mais comuns de HOF são as **funções lambda***

*são blocos de código que geralmente são passados como parâmetros de listeners ou para callbacks do tipo `onError` e `onSuccess`



Funções, Funções e mais funções

HOF – Funções de primeira Ordem

Que tal calcular a média de Duas notas, só que agora usando os conceitos de função e primeira Ordem? Vamos lá ?

Funções, Funções e mais funções

Function Types

Vamos criar uma calculadora? Utilizando FunctionType para as operações (Soma, Subtração, divisão e multiplicação) e a calculadora com high order function

Funções, Funções e mais funções

Inline Function

Instrui o compilador a inserir o corpo completo da função sempre que essa função for usada no código.

Funções, Funções e mais funções

Inline Function

Instrui o compilador a inserir o corpo completo da função sempre que essa função for usada no código.

Links Uteis

- <https://medium.com/@regmoraes/programa%C3%A7%C3%A3o-funcional-em-kotlin-parte-1-dd63fbd42a6e#:~:text=Vamos%20come%C3%A7ar%20com%20uma%20defini%C3%A7%C3%A3o,de%20resolver%20um%20de terminado%20problema.>
- <https://www.materialize.pro/programacao-funcional/#:~:text=Como%20os%20c%C3%B3digos%20da%20Programa%C3%A7%C3%A3o,de%20comprometer%20todo%20o%20c%C3%B3digo.>
- <https://medium.com/trainingcenter/programa%C3%A7%C3%A3o-funcional-para-iniciantes-9e2beddb5b43>
- <https://medium.com/pravaler-digital-team/ol%C3%A1-devs-645c77c6ae9b>
- <https://acervolima.com/funcoes-de-ordem-superior-do-kotlin/>
- <https://www.howtodoandroid.com/kotlin-high-order-functions/>
- <https://medium.com/@regmoraes/programa%C3%A7%C3%A3o-funcional-em-kotlin-parte-1-dd63fbd42a6e#:~:text=Nas%20linguagens%20declarativas%2C%20voc%C3%AA%20diz,atrav%C3%A9s%20da%20sua%20 biblioteca%20padr%C3%A3o.>
- <https://amitshekhar.me/blog/inline-function-in-kotlin>
- <https://kotlinlang.org/docs/inline-functions.html>
- <https://www.geeksforgeeks.org/kotlin-inline-functions/>