

Kotlin – Null Safety

Referências nulas e suas `NullPointerExceptions` (NPEs) não são coisas boas – tanto que são chamadas de “O erro de 1 bilhão de dólares”. O seguinte trecho de código é completamente **ilegal** no Kotlin:

```
fun main(args: Array<String>){  
    var nome:String = null  
}
```

Isso acontece porque por padrão **objetos não são nullable**, não recebem **null**. O mesmo acontece para argumentos de função:

```
fun nullFunction(nome: String){  
    // nome nunca será null. Dentro do escopo desta função  
    // não há a necessidade de checar se nome == null  
    println(nome.length)  
}
```

Você pode estar se perguntando: “mas e se eu quiser valores nulos?”. Você pode explicitamente informar que uma variável é *optional* (que pode ser nula) durante sua declaração usando o operador `?`, como no exemplo abaixo:

```
fun minhaFunction(){  
    var nome:String? = null  
}
```

Por hoje é só pessoal, sucesso nos estudos.