

Tópicos Abordados



- Exceções em Kotlin
- O bloco try/catch/finally
- O tipo Nothing

Exceções em Kotlin



- Herdam direta ou indiretamente da classe *Throwable*
- A expressão *throw* é usada para lançar uma exceção
- Diferentemente do Java, Kotlin não possui checked exceptions
 - O código que faz a chamada não é obrigado a tratar ou relançar a exceção

throw MyException("ERRO")

Bloco *try/catch/finally*



 Quando uma exceção é lançada, o código que fez a chamada pode tratá-la através de um bloco try/catch

```
fun somarPositivos(a: Int, b: Int): Int {
    if (a < 0 || b < 0) {
        throw ValueException("Valor negativo")
    }
    return a + b
}

try {
    somarPositivos(-5, 7)
    } catch (e: ValueException) {
        print("Erro na soma: ${e.message}")
    }
}</pre>
```

Bloco try/catch/finally

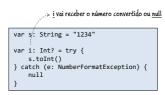


- Um bloco try pode ter 0 ou mais blocos catch associados
- Um bloco *try* pode ter 0 ou 1 bloco *finally* associado
- Um bloco try deve estar sempre associado a, no mínimo, um bloco try ou um bloco catch
- O bloco finally é sempre executado, independentemente do código ter entrado no bloco try ou catch

Bloco trv/catch/finally



- O bloco *try* é uma expressão
 - Retorna um valor



O Tipo Nothing Como throw é uma expressão, ela tem um retorno O retorno é do tipo Nothing Var v = s?.toUpperCase() ?: throw ValueException("A string não pode ser nula") Y vai ser uma String ou Nothing

