

Kotlin - async

Para operações onde existe a necessidade de ter um resultado ao final da execução existe a função *async*. Assim como o *launch*, o *async* é uma *coroutine* em *background* que possui o mesmo ciclo de vida, pode ser executada em um contexto diferente, mas que produz um resultado ao final da execução. O tipo de retorno da função *async* é o tipo *Deferred*, e este possui uma função chamada *await()* que retorna o resultado produzido ao final da execução da função *async*.

Exemplo de *async*:

```
fun main() {  
  
    GlobalScope.launch {  
  
        val pedidos = fetchOrders().await()  
  
        println(pedidos)  
  
    }  
  
}  
  
suspend fun fetchOrders() = GlobalScope.async {  
  
    delay(2000) // simula uma busca de dados externos  
  
    listOf(  
  
        Pedido(  
  
            id = 1, items = listOf(  
  
                ItemdoPedido(id = 1, descricao = "notebook", preco = 4000.0),  
  
                ItemdoPedido(id = 2, descricao = "smartphone", preco = 2000.0)  
  
            )  
  
        ),  
  
    ),  
  
}
```

Pedido(

id = 2, items = listOf(

ItemdoPedido(id = 3, descricao = "playstation 5", price = 6000.0),

ItemdoPedido(id = 4, descricao = "SmartTV", price = 5000.0)

)

)

)

}

Por hoje é só pessoal, sucesso nos estudos.