



<https://jagstakk.page.link/examples>

```
fun main() = runBlocking {  
    val orders = listOf(...)  
    val ordersFlow: Flow<Menu.Cappuccino> = orders.asFlow()  
  
    val espressoMachine = EspressoMachine(this)  
    val time = measureTimeMillis {  
        coroutineScope {  
            launch(CoroutineName("barista-1")) {  
                processOrders(ordersChannel, espressoMachine)  
            }  
            launch(CoroutineName("barista-2")) {  
                processOrders(ordersChannel, espressoMachine)  
            }  
        }  
    }  
    log("time: $time ms")  
    espressoMachine.destroy()  
}
```





environmental scope }



val pressoMachine = EspressoMachine(this)



val orders = listof(.

launch(CourtineName("barista-1")) {

```
val ordersFlow: Flow<Menu.Cappuccino> = orders.asFlow()
```

fun main() = runBlocking {



expressionsMachine.destroy()



validation = measureTimeMin {







log('time:\$time ms')

launch(CourtineNane("barista-2")) {



valid time = measured time Miss {

$\text{find\_main\_id}() \equiv \text{find\_loc\_king}()$

val ordersFlow:Flow<Menu.Capacitor> = orders.asFlow()



`val orders = listOf( . . . )`

val espreMachine = EspresMachine(this)





esspressoMachine.destroy()



log('time: \$time ms')