



**Cooperative Cancellation**





























































































































































































































































































































































































let's close the shop



this is blocking - so the main  
thread is blocked

we didn't change our  
dispatcher, so we're still on  
the main thread



main suspends here forever

```
fun main() = runBlocking {  
    val job = cashDesk()  
    delay(1000)  
    job.cancelAndJoin()  
}
```

```
fun CoroutineScope.cashDesk() = launch {  
    while (true) {  
        println("How many pretzels")  
        val count = readLine()?.toIntOrNull()  
  
        //receive pretzel  
    }  
}
```



# Cooperative Cancellation

```
fun main() = runBlocking {  
    val job = cashDesk()  
    delay(1000)  
    job.cancelAndJoin()  
}
```

main suspends here forever

let's close the shop

we didn't change our dispatcher, so we're still on the main thread

```
fun CoroutineScope.cashDesk() = launch {  
    while (true) {  
        println("How many pretzels?")  
        val count = readLine()?.toIntOrNull()  
  
        //receive pretzel  
    }  
}
```

this is blocking - so the main thread is blocked

# Cooperative Cancellation

```
fun main() = runBlocking {  
    val job = cashDesk()  
    delay(1000)  
    job.cancelAndJoin()  
}
```

```
fun CoroutineScope.cashDesk() = launch {  
    while (true) {  
        println("How many pretzels?")  
        val count = readLine()?.toIntOrNull()  
        yield()  
        //receive pretzel  
    }  
}
```