@Louis_CAD

# Writing your own: parallel composition

```kotlin
/**
 * Returns when meal is ready.
 */
suspend fun prepareLouisFavoriteMeal() = coroutineScope {
    launch {
        cookSoftBoiledEggs()
    }
    buyIceCream(
        brand = "Häagen Dazs",
        flavor = "Vanilla Caramel Brownie"
    )
}
```

**throws an IllegalStateException**

**cancels parent** (and children, if any)

cancels children
& <u>waits for</u>
<u>their completion</u>

finally re-throws

# Writing your own: parallel composition

```
/**
 * Returns when meal is ready.
 */
suspend fun prepareLouisFavoriteMeal() = coroutineScope {
    launch {
        cookSoftBoiledEggs()
    }
    buyIceCream(
        brand = "Häagen Dazs",
        flavor = "Vanilla Caramel Brownie"
    )
}
```

finally re-throws

cancels parent (and children, if any)

throws an `IllegalStateException`

cancels children & <u>waits for their completion</u>

What if eggs are stolen?

@Louis_CAD

# Writing your own: parallel composition

```kotlin
/**
 * Returns when meal is ready.
 */
suspend fun prepareLouisFavoriteMeal() = coroutineScope {
    launch {
        cookSoftBoiledEggs()
    }
    buyIceCream(
        brand = "Häagen Dazs",
        flavor = "Vanilla Caramel Brownie"
    )
}
```