| Top-level initialization | → | `run` |

```kotlin
val myObj = run {
    val generator = DateStringGenerator(2008, 9, 23)
    generator.locale = "US"
    generator.localizedDate
}
```

| Null check | → | `T?.let` |

```kotlin
class MyClass {

    fun printExceptionMessage(exception: Exception?) {
        exception?.let {
            println(exception.message)
        }
    }
}
```

Chaining operations

| Object configuration | → | `T.apply` |

```kotlin
val lastWeeksDate: String = Calendar.getInstance().apply {
    add(Calendar.DAY_OF_YEAR, -7)
}
```

| Side-effect | → | `T.also` |

```kotlin
    .also {
        println("Created calendar of type: ${it.calendarType}")
    }
```

| map | → | `T.let` |

```kotlin
    .let { calendar ->
        val format = SimpleDateFormat("dd/M/yyyy hh:mm:ss", Locale.US)
        format.format(calendar.time)
    }
}
```