

Kotlin Standard Scoping Functions

input	receiver		parameter
binding in lambda			
receiver	apply	run	with
parameter	also	let	
output	same object	result of lambda	

Definitions

```
fun <T, R> T.run(block: T.() -> R): R = block()
fun <T, R> with(receiver: T, block: T.() -> R): R = receiver.block()
fun <T> T.apply(block: T.() -> Unit): T { block(); return this }
fun <T> T.also(block: (T) -> Unit): T { block(this); return this }
fun <T, R> T.let(block: (T) -> R): R = block(this)
```

Examples

```
val r: R = T().run { this.foo(); this.toR() }
val r: R = with(T()) { this.foo(); this.toR() }
val t: T = T().apply { this.foo() }
val t: T = T().also { it.foo() }
val r: R = T().let { it.foo(); it.toR() }
```