Introduction to Swift

Part 7: Functions

raywenderlich.com

Declaring a Function

func **name** (**param 1 name** : **param 1 type** , …) -> **retval type** {

    return **expression**

}

```
func addNums(x: Int, y: Int) -> Int {
  return x + y
}

addNums(1, 2)
addNums(3, 4)
```

Functions Without Params/Return Values

```
func noParams() {
  println("Look ma, no params!")
}
noParams()
```

External Parameter Names

```
import CoreGraphics
func lerp(#start:CGPoint, #end:CGPoint, #t:CGFloat) -> CGPoint {
  let range = CGPoint(x: end.x - start.x, y: end.y - start.y)
  return CGPoint(x: start.x + range.x*t, y: start.y + range.y*t)
}

lerp(start: CGPoint(x: 100, y: 100), end: CGPoint(x: 200, y: 200), t: 0.2)
```

Demo

Challenge Time!

- From Project Euler:
  - Find the sum of multiples of 3 or 5 below 1000.
- Abstract this!
  - Write a function to find the sum of **any two multiples**, below **any max value** (default 1000)
- Should be able to call it like this:

```
sumOfMultiples(mult1: 3, mult2: 5)
```

- Answer should be: **233,168**