

Notes

My notes contain fragments from Paul Hudson's blog: <https://www.hackingwithswift.com/100/swiftui>

Day 1 & 2 Recap

```
var myString = "This is my string"
print(myString.count)
print(myString.uppercased())
```

```
let number = 120
print(number.isMultiple(of: 3))
```

```
let filename = "paris.jpg"
print(filename.hasSuffix(".jpg"))
print(filename.hasPrefix("paris"))
```

```
var gameOver = false
gameOver.toggle()
```

Day 3 Recap

Arrays

Swift makes sure the array only contains one type of data at a time.

```
var beatles = ["John", "Paul", "George", "Ringo"]
beatles.append("Adrian")
```

Starting with an empty array:

```
var scores = Array<Int>() // or var scores = [Int]()
scores.append(100)
scores.append(80)
scores.append(85)
```

```
scores.count
scores.remove(at: 2)
scores.removeAll()
```

```
let bondMovies = ["Casino Royale", "Spectre", "No Time To Die"]
print(bondMovies.contains("Frozen")) //false
```

```
let cities = ["London", "Tokyo", "Rome", "Budapest"]
print(cities.sorted()) // ["Budapest", "London", "Rome", "Tokyo"]
```

Dictionaries

<https://developer.apple.com/documentation/swift/dictionary>

"Subscripting a dictionary with a key returns an optional value, because a dictionary might not hold a value for the key that you use in the subscript."

We can provide a default value to use if the key doesn't exist:

```
print(employee2["name", default: "Unknown"])
```

Creating new dictionary:

```
var heights = [String: Int]()
heights["Yao Ming"] = 229
heights["Shaquille O'Neal"] = 216
heights["LeBron James"] = 206
```

Count and removeAll() both exists for dictionaries, and work just like they do for arrays.

Sets

Sets are similar to arrays, except you can't add duplicate items, and they don't store their items in a particular order.

```
let people = Set(["Denzel Washington", "Tom Cruise", "Nicolas Cage", "Samuel L Jackson"])
```

```
var people = Set<String>()
people.insert("Denzel Washington")
people.insert("Tom Cruise")
people.insert("Nicolas Cage")
people.insert("Samuel L Jackson")
```

Alongside contains(), you'll also find count to read the number of items in a set, and sorted() to return a sorted array containing the the set's items.

Enums

```
enum Weekday {
    case monday
    case tuesday
    case wednesday
    case thursday
    case friday
}
var day = Weekday.monday
day = Weekday.tuesday
day = Weekday.friday
```

Once you assign a value to a variable or constant, its data type becomes fixed – you can't set a variable to a string at first, then an integer later on. Well, for enums this means you can skip the enum name after the first assignment, like this:

```
var day = Weekday.monday
day = .tuesday
day = .friday
```