SE 4020: Mobile Application Design and Development

Lecture 02 – Basics of Swift

Topics

- Calculations
- Collections
  - Arrays
  - Dictionaries
  - Set
- Control Structures
  - If, Switch
  - For, While, Repeat
- Ranges
- Optionals

Calculations

- You can't mix and match whitespace styles...

- ...you have to pick one style and stick with it. So your expressions and operators can either be like 8/4 or be like 8 / 4, but they can't be like 8 /4 or be like 8/ 4.

Arrays

- Value Type
- Methods
  - append()
  - remove()
  - insert()
  - max()
  - min()
- Properties
  - count
  - isEmpty

If statement

- Same conditional operators as other C based languages
- Have to use explicit { } for if and else parts

```
var planet = "Jupiter"
if(planet == "Jupiter") {
    print("Jupiter is named after the Roman king of the gods.")
} else if(planet == "Neptune") {
    print("Neptune is inhospitable to life as we know it.")
} else if(planet == "Mars") {
    print("Mars has a lot of Earth-made rovers on it.")
} else if(planet == "Earth") {
    print("Earth is infested with something called 'humans'.")
} else {
    print("All the planets are pretty cool.")
}
```

Switch

- No break statement needed to exit a condition

```
var number = 6

switch(number) {
case 6:
    print("6 means easy and smooth, all the way!")
case 8:
    print("8 means sudden fortune!")
case 99:
    print("99 means eternal!")
default:
    print("I've told you everything I know about lucky numbers.")
}
```

Exercise

- Calculation of grades using if

>= 80 A

>= 75 A-

>=70 B+

>= 65 B

>= 60 B-

>= 55 C+

>= 45 C

>= 40 C-

Dictionaries

```
var studentGPAs = ["IT21123456": 3.9, "IT21234567": 3.6,
                    "IT21345678": 3.2, "IT21456789": 2.8]

// Accessing and updating values in the dictionary
let gpa1 = studentGPAs["IT21123456"] // gpa1 = 3.9
studentGPAs["IT21234567"] = 3.8
```
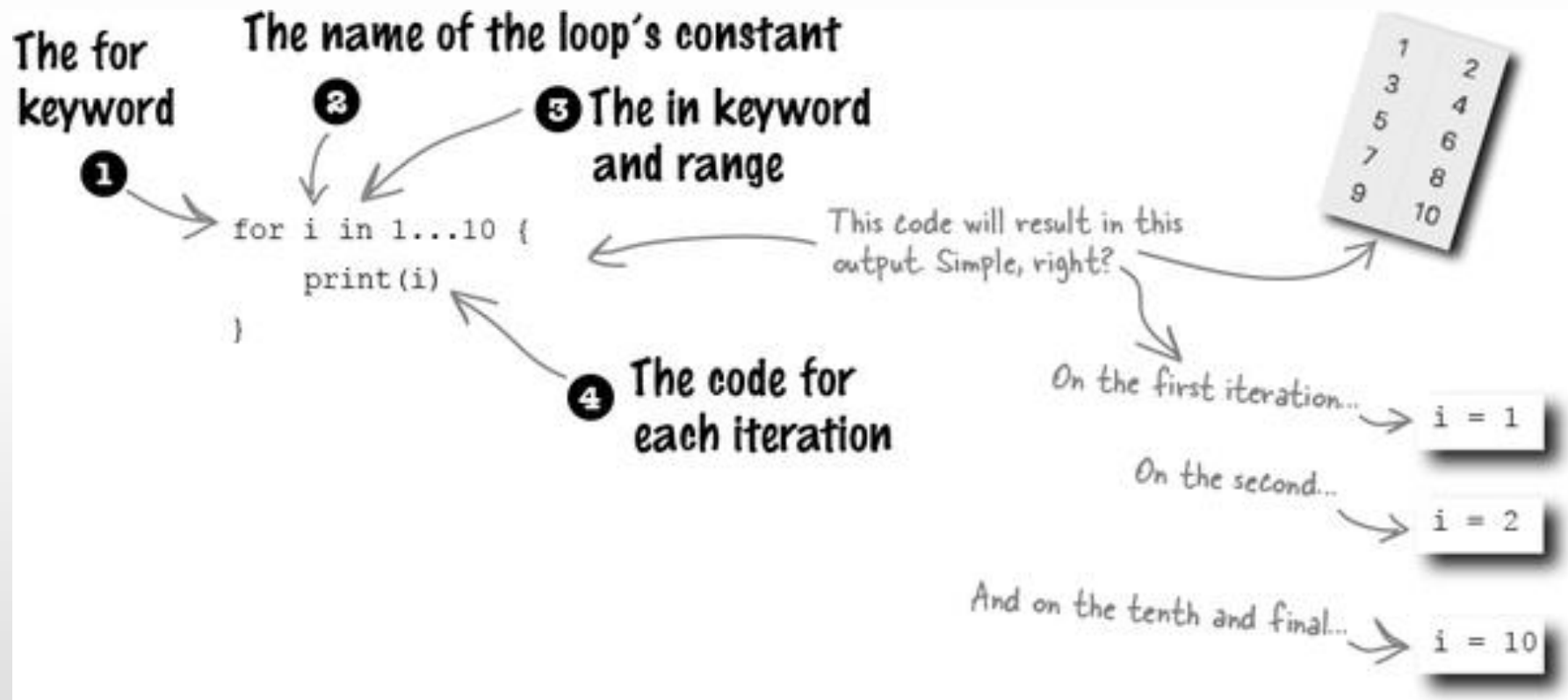
Sets

- Sets are similar to arrays and must also contain just one type, but **sets are unordered, and <u>there can only be one of each distinct value in a set</u>**.

- You can create a set like this:

- `var evenNumbers = Set([2, 4, 6, 8])`

- Or like this:

- `var oddNumbers: Set = [1,3,5,7]`

- `oddNumbers.insert(9)`

- `oddNumbers.remove(5)`

Exercise

1. Write a program to store details of your friends phone numbers.

2. Write a program to store unique towns a group of people come from

For

For

Our drinks array.

Apparently our café is licensed...

```
let drinks = ["Coffee", "Tea", "Water", "Whisky"]
```

Iterating through all the items in the drinks array.

```
for drink in drinks {
    print("\(drink) is still available!")
}
```
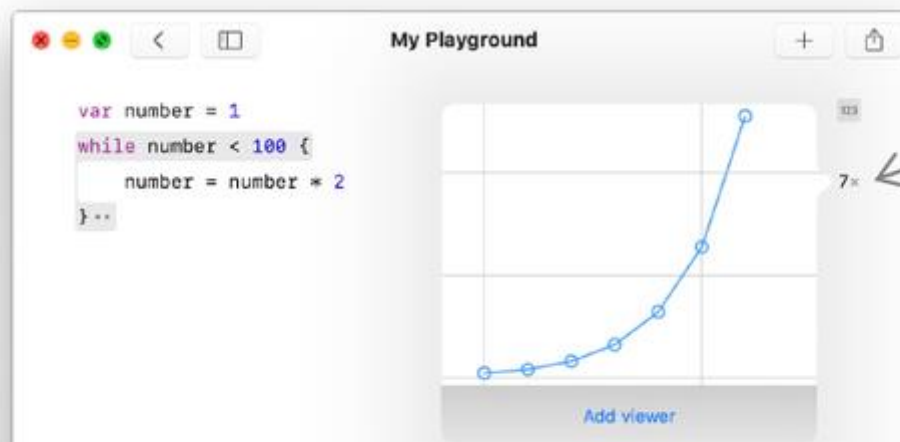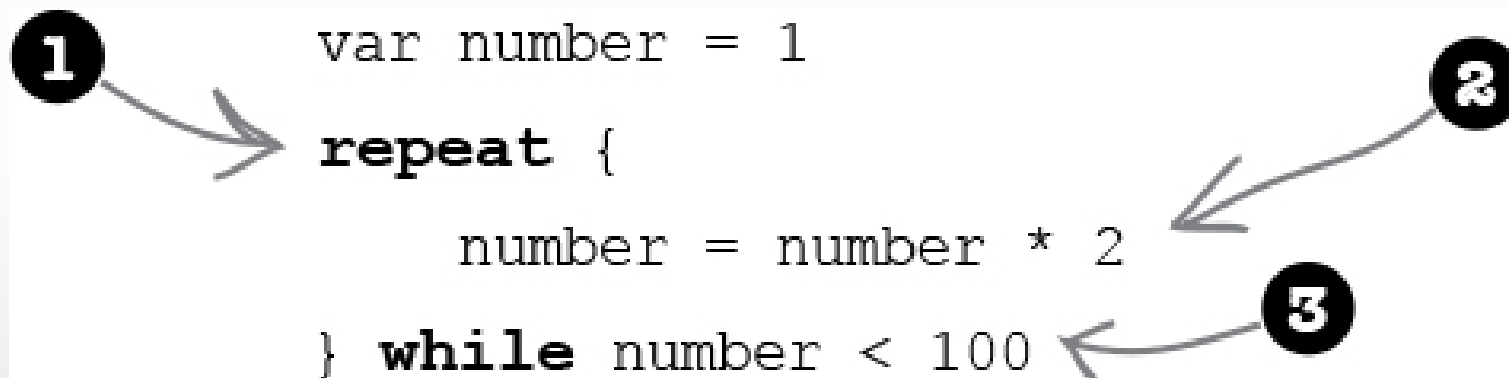
While

Repeat

```
var number = 1

repeat {

    number = number * 2

} while number < 100
```

Exercise

1. Store marks of students in an array

Calculate and display if they have passed or failed the module. Assume the pass mark is >= 50

Ranges

- Closed Range Operator: a...b
  - Represents a range that includes both a and b.
- Half-Open Range Operator: a..<b
  - Represents a range that includes a but not b.
- One-Sided Ranges: a... or ...b
  - Represents a range from a to the end or from the beginning to b.
- Countable Range: a..<b or a...b where a and b are integers
  - Represents a range of integers that can be enumerated using a for-in loop.
- Strideable Range: stride(from: a, to: b, by: s) or stride(from: a, through: b, by: s)
  - Represents a range of values that can be stepped through with a certain stride or step size, where a is the starting value, b is the end value (exclusive or inclusive), and s is the step size.

Exercise

- Calculation of grades using ranges and switch

>= 80 A

>= 75 A-

>=70 B+

>= 65 B

>= 60 B-

>= 55 C+

>= 45 C

>= 40 C-

Optionals

- An optional value either contains a value or contains nil to indicate that a value is missing. Write a question mark (?) after the type of a value to mark the value as optional.