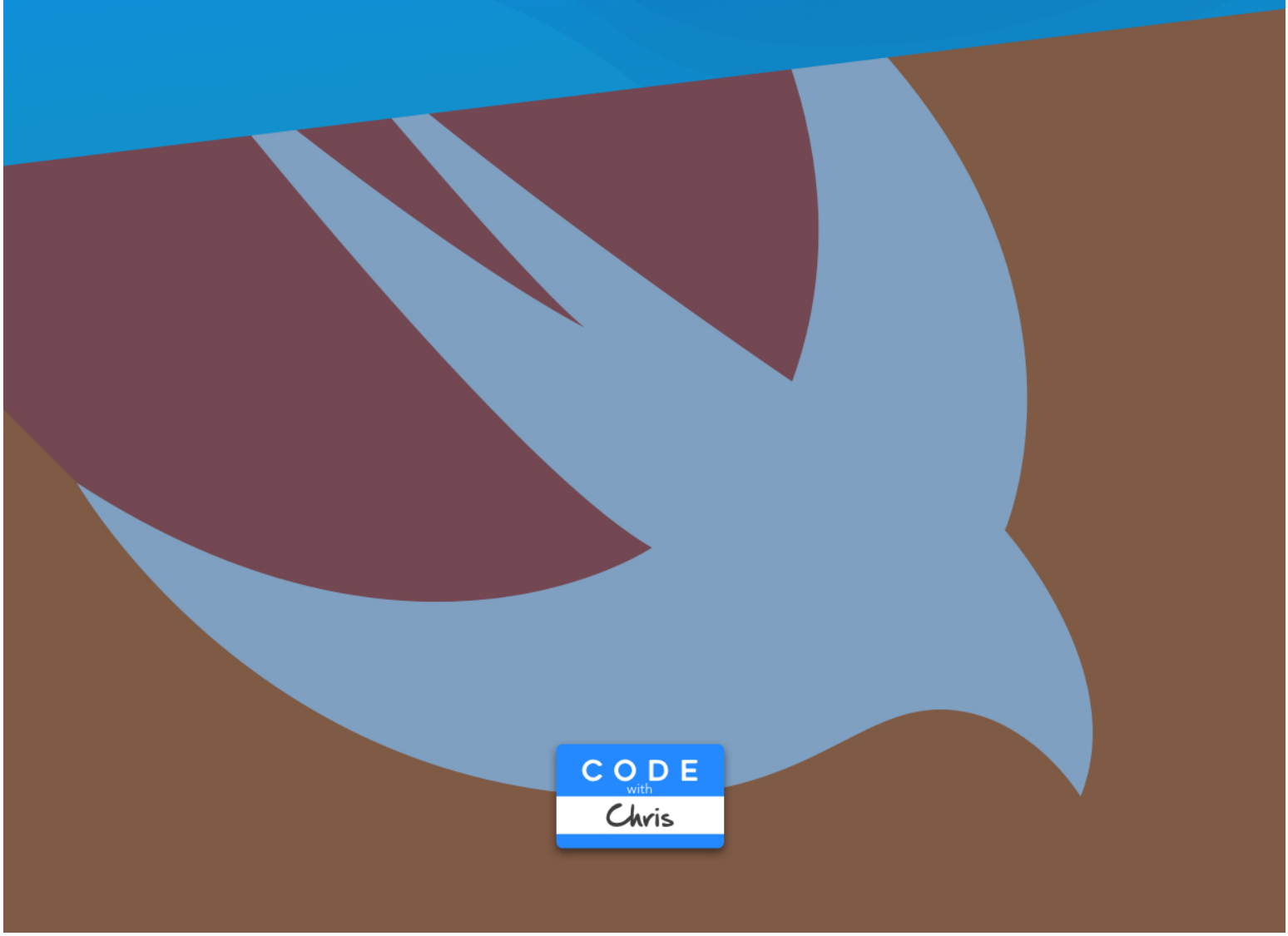
HOW TO MAKE AN APP FOR BEGINNERS

Lesson 7
Swift Basics Part 3

RECAP NOTES AND EXERCISES



CODE with Chris

Key Concepts

1. You can group functions into a class

2. When a function is inside a class, it's called a "method" of the class

3. To declare a new class, use the "class" keyword, followed by the name of the class and then a pair of curly brackets

```
class myClass {

}
```

4. Inside the curly brackets, you put the methods and code for that class

5. Classes are usually declared to represent something. Most often, you'll be declaring classes to represent a type of data or a specific role that is crucial to making the app work.

   For example, if your app was an employee database, you might create an Employee class to represent an employee. In the same app, you might create a DatabaseManager class that is responsible for interacting with the database. You would then put all of the database interaction code inside the DatabaseManager class.

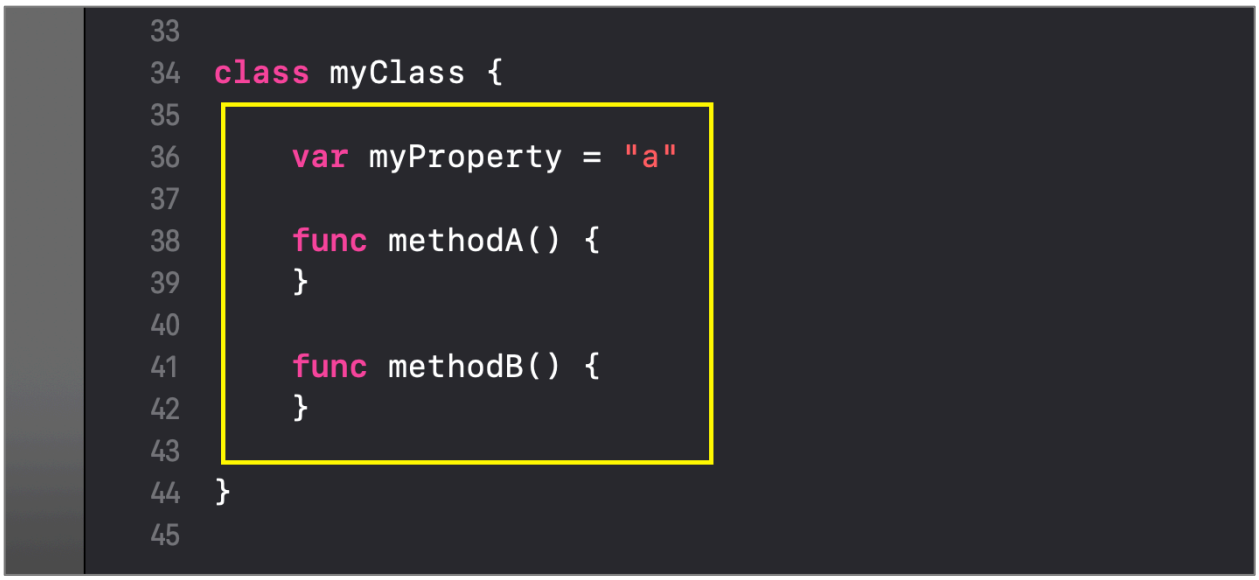
6. When you're designing a class, think of it like a blueprint, recipe or mould.

7. A class by itself is just a "design". It can't actually perform any action or be useful to anyone until you bring it to life and create an instance of it (also called an object of the class).

8. You can create many objects of the class. Each are independent instances of that class.

9. If you declare a variable inside the class but outside of any method, then it's known as a property of that class.

10. This property can be accessed by the code inside any of the class' methods because the property and methods are all within the scope of the class.



```
33
34  class myClass {
35
36      var myProperty = "a"
37
38      func methodA() {
39      }
40
41      func methodB() {
42      }
43
44  }
45
```

11. You create a new object by writing the Class name followed by a pair of rounded brackets

12. You can then call methods and access properties of that object by writing a "." after the object like this:

```
var myClassObject = myClass
myClassObject.methodA()
```

13. This is called dot-notation

Exercises

This exercise continues from the Swift Basics (Part 3) taught in lesson 7.

In this worksheet, you'll continue with the Spaceship example and practice declaring and calling classes and methods using dot notation.

Setup

We'll be doing these exercises in a Swift Playground.

Open Xcode and create a new playground
(File Menu->New->Playground).

From the list of Playground templates, just select "Blank"

Step 1:

Declare a class called "Spaceship".

Inside of it:

- Declare a property of type Int called "fuelLevel" and set it to 50.
- Declare a method called "liftOff"
- Declare a method called "addFuel"
- Declare a method called "thrust"
- Declare a method called "cruise"

Step 2:

Inside the "liftOff" method, write code that will:

- Decrement the fuelLevel property by 50.
- Print to the console "We have lift off!".

- Print the current fuelLevel to the console.
- Use a message like “Current Fuel Level at: X” where X is the actual value of the fuelLevel property.
- Hint: Substitute dynamic values into strings using “\(fuelLevel)” inside the string.

Step 3:

Regarding the “addFuel” method:

- Change the method declaration so that it can accept an Int parameter called “fuel”.
- Inside the function, write code that will increment the fuelLevel property by the value of the parameter.
- Print “Fuel added”.
- Print out the current fuel level to the console (just like in the “liftOff” method).

Step 4:

Inside the “thrust” function, write code that will:

- Decrement the fuelLevel property by 15.
- Print to the console “Rocket is thrusting”.
- Print out the current fuel level (just like in the “liftOff” method).

Step 5:

Inside the “cruise” function, write code that will:

- Decrement the fuelLevel property by 5.
- Print to the console “Rocket is cruising”.

- Print out the current fuel level to the console (just like in the “liftOff” method).

Step 6:

Now execute the following commands:

- Create a new Spaceship object.
- Call the “addFuel” method of your object and pass in 50.
- Call the “liftOff” method.
- Call the “thrust” method.
- Call the “cruise” method.

Solutions

Download the solutions for all the exercises here:
https://codewithchris.com/beginner-youtube/