
FOR INSTRUCTOR PURPOSES ONLY

MATERIALS

- + [Lab](#)
- + [Lesson](#)
- + [Lab Solution Code](#)

SWIFT FUNCTIONS PART 1

Wellington Moreno

Lead iOS Instructor, General Assembly



QUESTION

**WHAT HAPPENS IF WE KEEP
WRITING CODE THIS WAY?**

ACTIVITY: THINK-PAIR-SHARE



5 mins

DIRECTIONS

What problems do you foresee if I continue writing code like this?

1. Write your thoughts down (1 min)
2. Pair up and share (2 min)
3. Discuss (1 min)

DELIVERABLE

Discuss your answers with the person next to you.

LESSON

LEARNING OBJECTIVES

- + **Identify** the purpose of a function
- + **Identify** the different parts of a function
- + **Write** functions that accept parameters and return values
- + **Call** a function

INTRODUCTION

FUNCTIONS

LET'S GET FUNCY!

DEFINITION

**“FUNCTIONS ARE SELF-CONTAINED
CHUNKS OF CODE THAT PERFORM A
SPECIFIC TASK.”**

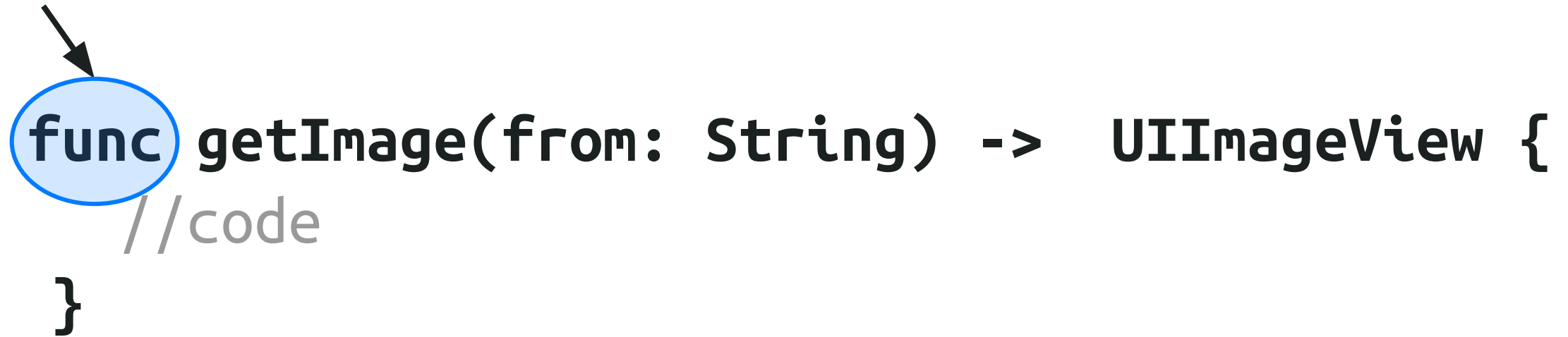
- SWIFT REFERENCE

THE ANATOMY OF A FUNCTION

```
func getImage(from: String) -> UIImageView {  
    //code  
}
```


THE ANATOMY OF A FUNCTION

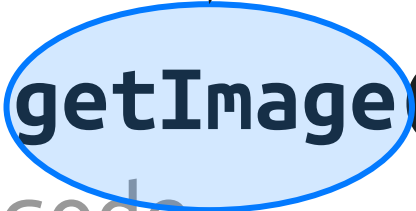
Function
Keyword



```
func getImage(from: String) -> UIImageView {  
    //code  
}
```

THE ANATOMY OF A **FUNCTION**

Function Name



```
func getImage(from: String) -> UIImageView {  
    //code  
}
```

THE ANATOMY OF A **FUNCTION**

Parameters



```
func getImage(from: String) -> UIImageView {  
    //code  
}
```

THE ANATOMY OF A FUNCTION

Return Symbol



```
func getImage(from: String) -> UIImageView {  
    //code  
}
```

The diagram illustrates the components of a Swift function signature. The text 'Return Symbol' is positioned above a blue circle containing a black arrow pointing to the right. This circle highlights the return type 'UIImageView' in the function signature 'func getImage(from: String) -> UIImageView {'. The function signature is followed by a block of code enclosed in curly braces, with a comment '//code' on the first line.

THE ANATOMY OF A **FUNCTION**

Return Type

```
func getImage(from: String) -> UIImageView {  
    //code  
}
```

A diagram illustrating the return type of a Swift function. The text 'UIImageView' is enclosed in a light blue oval with a blue border. A black arrow points from the text 'Return Type' to the top of this oval.

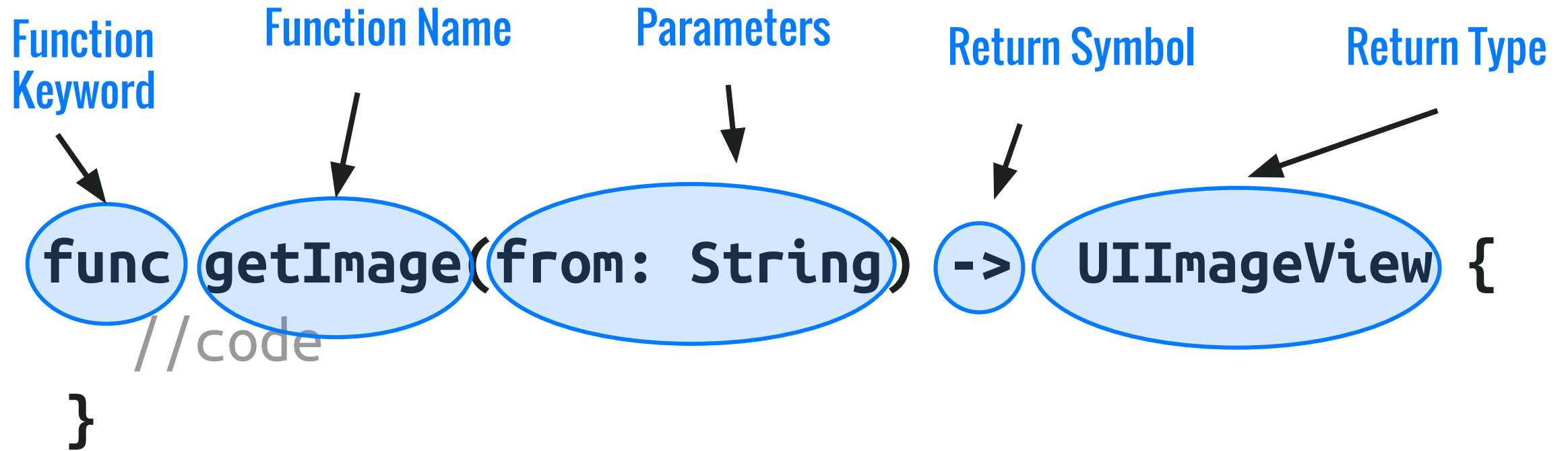
THE ANATOMY OF A **FUNCTION**

```
func getImage(from: String) -> UIImageView {  
    return image  
}
```



Return Statement

THE ANATOMY OF A **FUNCTION**



ACTIVITY: WHITEBOARDING - GET FUNCY



CODE

5 mins

DIRECTIONS

On your desks or on the whiteboard, write a function that takes a speed (m/s) parameter and a time-traveled (minutes) parameter, and returns the total distance traveled.

DELIVERABLES

Your function

ACTIVITY: THINK-PAIR-SHARE



5 mins

DIRECTIONS

Think of an analogy for a function. If you could make a function for anything, what would you make?

1. Think! (1 min)
2. Pair up and share (2 min)
3. Discuss (1 min)

DELIVERABLE

Discuss your answers with the person next to you.

DEMO

COLLABO TIME

- + Write a function that takes a String URL and returns an Image

QUESTION

WHAT'S WRONG HERE?

WHAT'S WRONG?

```
func isEmpty(string: String) {  
    return string.characters.count == 0  
}
```

WHAT'S WRONG?

```
func save(username: String) -> String {  
    settings.save(username)  
}
```

WHAT'S WRONG?

```
func save(_ username: String) {  
    settings.save(username)  
}
```

WHAT'S WRONG?

```
func sayHello() -> String {  
    print("Hello \ (name)")  
}
```

WHAT'S WRONG?

```
func canWatch(movie: String, age: Int) -> Bool {  
    if age >= 18 {  
        return true  
    }  
  
    print("Sorry kid. You're too young.")  
}
```


WHAT'S WRONG?

```
func drinkWater(liters: Int) {  
    let cup = Cup()  
    cup.fillWithWater(liters)  
    cup.drink()  
}
```

```
func drinkMilk(liters: Int) {  
    cup.fillWithMilk(liters)  
    cup.drink()  
}
```

INDEPENDENT PRACTICE: PLAYGROUNDS



CODE

15 mins

DIRECTIONS

Write a function that:

1. Takes a name and prints "Hello {name}"
2. Takes 4 numbers and tells me if they are different
3. Takes a person's birthdate (year, month, day) and tells me how many days they have been alive.
4. Takes an integer and tells me whether it is a perfect square
5. Take an integer and tells me whether it is a perfect number

DELIVERABLES

A Playgrounds file with your answers

ACTIVITY: THINK-PAIR-SHARE



EXERCISE

5 mins

DIRECTIONS

Think of an analogy for a function. If you could make a function for anything, what would you make?

1. Think! (1 min)
2. Pair up and share (2 min)
3. Discuss (1 min)

DELIVERABLE

Discuss your answers with the person next to you.

LESSON

Q & A

CONCLUSION

- + Functions help write cleaner and more organized code
- + Functions help reduce code duplication by allowing you to reuse functionality

BEST PRACTICES

**WRITE DESCRIPTIVE
FUNCTION NAMES USING
VERBY WORDS**

BEST PRACTICES

**NAME YOUR PARAMETERS
WELL AND CLEARLY**

BEST PRACTICES

**FUNCTIONS SHOULD
CONCEPTUALLY DO ONE
THING**

BEST PRACTICES

**CHECK YOUR SHIT AT THE
DOOR**

BEST PRACTICES

KEEP IT TIGHT

BEST PRACTICES

KEEP FUNCTIONS AT THEIR LEVEL

LESSON

Q & A

LESSON

BEFORE NEXT CLASS

+ Read up on "Variadic Parameters"

THANKS!

WELLINGTON MORENO

- + GitHub/Slack: @SirWellington
- + Twitter: @SirWellingtonZ
- + Email: wellington.moreno@ga.co