

# **Unit 2—Lesson 2:**

## **Functions**

# Le Bar de Greg

## Problème

```
print(".: Le Bar de Greg :.")  
print("_____")  
print("Résumé du ticket de caisse 02 Mai 2019")
```



# Functions

## Defining a function

```
func functionName(parameters) -> ReturnType {  
    // Body of the function  
}
```

# Functions

## Defining a function

```
func displayBarGreg() {  
    print(".: Le Bar de Greg :.")  
    print("_____")  
    print("Résumé du ticket de caisse 02 Mai 2019")  
}
```

# Parameters

```
func displayBarGreg() {  
    print(".: Le Bar de Greg :.")  
    print("_____")  
    print("Résumé du ticket de caisse 02 Mai 2019")  
}
```

displayBarGreg()

.: Le Bar de Greg :.

\_\_\_\_\_

Résumé du ticket de caisse 02 Mai 2019



displayBarGreg()



displayBarGreg()



displayBarGreg()



# Parameters

```
func displayBarGreg() {  
    print(".: Le Bar de Greg :.")  
    print("_____")  
    print("Résumé du ticket de caisse 02 Mai 2019")  
}
```

displayBarGreg()

.: Le Bar de Greg :.

\_\_\_\_\_

Résumé du ticket de caisse 02 Mai 2019

# Parameters

```
func displayBarGreg(day: Int) {  
    print(".: Le Bar de Greg :.")  
    print("_____")  
    print("Résumé du ticket de caisse \(day) Mai 2019")  
}
```

```
displayBarGreg(day: 10)
```

.: Le Bar de Greg :.

\_\_\_\_\_

Résumé du ticket de caisse 10 Mai 2019

# Argument labels

```
func sayHello(firstName: String) {  
    print("Hello, \(firstName)!")  
}
```

```
sayHello(firstName: "Amy")
```

# Argument labels

```
func sayHello(to: String, and: String) {  
    print("Hello \(to) and \(and)")  
}
```

```
sayHello(to: "Luke", and: "Dave")
```

# Argument labels

## External names

```
func sayHello(to person: String, and anotherPerson: String) {  
    print("Hello \(person) and \(anotherPerson)")  
}
```

```
sayHello(to: "Luke", and: "Dave")
```

# Argument labels

## Omitting labels

```
print("Hello, world!")
```

```
func add(_ firstNumber: Int, to secondNumber: Int) -> Int {  
    return firstNumber + secondNumber  
}
```

```
let total = add(14, to: 6)
```

# Parameters

```
func triple(value: Int) {  
    let result = value * 3  
    print("If you multiply \(value) by 3, you'll get \(result).")  
}
```

```
triple(value: 10)
```

If you multiply 10 by 3, you'll get 30.

# Parameters

## Multiple parameters

```
func multiply(firstNumber: Int, secondNumber: Int) {  
    let result = firstNumber * secondNumber  
    print("The result is \(result).")  
}
```

```
multiply(firstNumber: 10, secondNumber: 5)
```

The result is 50.



# Return values

```
func multiply(firstNumber: Int, secondNumber: Int) -> Int {  
    let result = firstNumber * secondNumber  
    return result  
}
```

# Return values

```
func multiply(firstNumber: Int, secondNumber: Int) -> Int {  
    return firstNumber * secondNumber  
}
```

```
let myResult = multiply(firstNumber: 10, secondNumber: 5)  
print("10 * 5 is \ (myResult)")
```

# Unit 2—Lesson 2

## Lab: Functions



Open and complete the exercises in Lab – `Functions.playground`

