CSM 3505

NATIVE MOBILE PROGRAMMING

LAB 2

S58101

LOKE JOO TUCK

INTRODUCTION TO PROGRAMMING IN KOTLIN

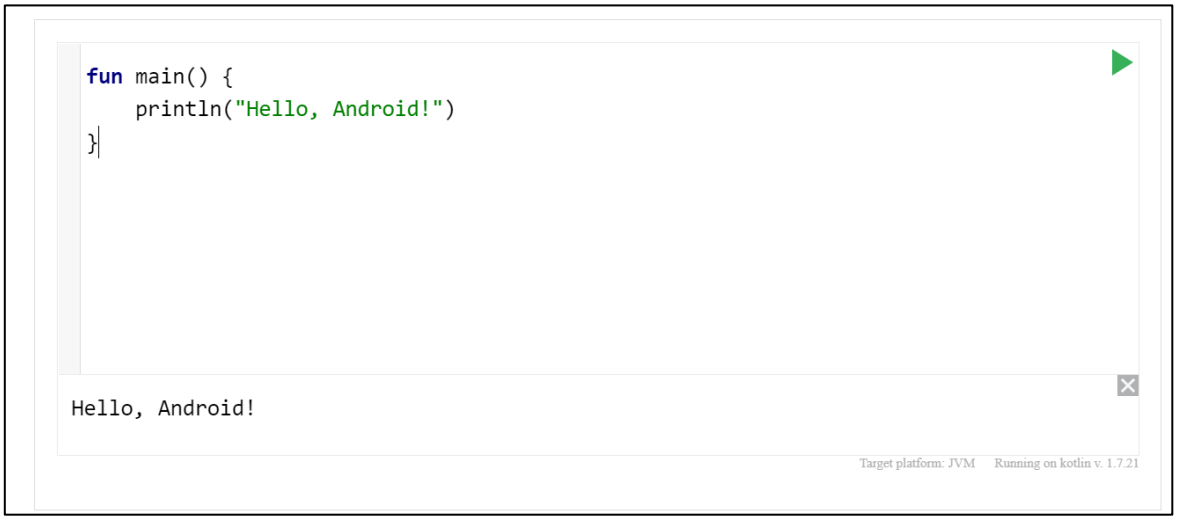
1 Before you begin

[No activities]

2 Welcome to Android Basics with Compose

[No activities]

3 Your First Program in Kotlin



```
fun main() {
    println("Hello, Loke Joo Tuck!")
}
```

```
Hello, Loke Joo Tuck!
```

Target platform: JVM Running on kotlin v. 1.7.21

3.8 Fixing Errors in Your Code

```
fun main() {
    println("Today is sunny!)
}
```

```
Expecting '"'
Expecting ')'
```

Target platform: JVM Running on kotlin v. 1.7.21

Fixed code:

```
fun main() {
    println("Today is sunny!")
}
```

```
Today is sunny!
```

Target platform: JVM Running on kotlin v. 1.7.21

3.9 Exercises

Question:

9. Exercises

1. Can you read the code in this program and guess what the output is (without running it in Kotlin Playground)?

```
fun main() {
    println("1")
    println("2")
    println("3")
}
```

Answer:

1

2

3

Question:

2. Use the Kotlin Playground to create a program that outputs the following messages:

```
I'm
learning
Kotlin!
```

Answer:

```
fun main() {
    println("I'm")
    println("learning")
    println("Kotlin!")
}
```

```
I'm
learning
Kotlin!
```

Target platform: JVM Running on kotlin v. 1.7.21

Question:

3. Copy and paste this program into the Kotlin Playground.

```
fun main() {
    println("Tuesday")
    println("Thursday")
    println("Wednesday")
    println("Friday")
    println("Monday")
}
```

Fix the program so that it prints this output:

Answer:

```
fun main() {
    println("Monday")
    println("Tuesday")
    println("Wednesday")
    println("Thursday")
    println("Friday")
}
```

```
Monday
Tuesday
Wednesday
Thursday
Friday
```

Question:

4. Fix the error in this program, so that it produces the desired output.

```
fun main() {
    println("Tomorrow is rainy")
```

Desired output:

```
Tomorrow is rainy
```

Answer:

```
fun main() {
    println("Tomorrow is rainy")
}
```

```
Tomorrow is rainy
```

Question:

5. Fix the error in this program, so that it produces the desired output.

```
fun main() {
    printLine("There is a chance of snow")
}
```

Desired output:

```
There is a chance of snow
```

Answer:

```
fun main() {
    println("There is a chance of snow")
}
```

```
There is a chance of snow
```

Question:

6. Fix the error in this program, so that it produces the desired output.

```
fun main() {
    println("Cloudy") println("Partly Cloudy") println("Windy")
}
```

Desired output:

```
Cloudy
Partly Cloudy
Windy
```

Answer:

```
fun main() {
    println("Cloudy")
    println("Partly Cloudy")
    println("Windy")
}
```

```
Cloudy
Partly Cloudy
Windy
```

Question:

7. Fix the error in this program, so that it produces the desired output.

```
fun main() (
    println("How's the weather today?")
)
```

Desired output:

```
How's the weather today?
```

Answer:

```
fun main() {
    println("How's the weather today?")
}
```

```
How's the weather today?
```

4 Create and Use variables in Kotlin

Try it

1. Open your favorite app on your phone.
2. Identify where you think variables are used in the app on that particular screen.
3. Guess what data type those variables are.
4. Share your answers on social media with a screenshot of the app, an explanation of where you think variables are used, and the hashtag #AndroidBasics.

Answer:

Telegram App Main Page

Data Type:

Chat/Group name – String

Mute notification – Boolean

Messages – Int

4.3 Define and use variables

```
fun main() {
    val count: Int = 2
    println(count)
}
```

```
2
```

Without dollar sign for variable “count”:

```
fun main() {
    val count: Int = 2
    println("You have count unread messages.")
}
```

```
You have count unread messages.
```

With dollar sign for variable “count”:

```
fun main() {
    val count: Int = 2
    println("You have $count unread messages.")
}
```

```
You have 2 unread messages.
```

Assigning different value for the variable:

```
fun main() {
    val count: Int = 10
    println("You have $count unread messages.")
}
```

```
You have 10 unread messages.
```

```kotlin
fun main() {
    val unreadCount = 7
    val readCount = 150
    println("You have ${unreadCount + readCount} total messages in your inbox.")
}
```

```
You have 157 total messages in your inbox.
```

```kotlin
fun main() {
    val unreadCount = 7
    val readCount = 150
    println("You have $unreadCount + readCount total messages in your inbox.")
}
```

```
You have 7 + readCount total messages in your inbox.
```

For example, modify your program to print this out:

```
100 photos
10 photos deleted
90 photos left
```

Answer:

```
fun main() {
    val photoNo = 100
    val delPhoto = 10
    println("$photoNo photos")
    println("$delPhoto photos deleted")
    println("${photoNo - delPhoto} photos left")
}
```

```
100 photos
10 photos deleted
90 photos left
```

4.4 Update variables

```
fun main() {
    var cartTotal = 0
    println("Total: $cartTotal")

    cartTotal = 20
    println("Total: $cartTotal")
}
```

```
Total: 0
Total: 20
```

5 Create and Use Functions in Kotlin

```kotlin
fun main() {
    birthdayGreeting()
}

fun birthdayGreeting() {
    println("Happy Birthday, Rover!")
    println("You are now 5 years old!")
}
```

```
Happy Birthday, Rover!
You are now 5 years old!
```

```kotlin
fun main() {
    birthdayGreeting()
}

fun birthdayGreeting(): Unit {
    println("Happy Birthday, Rover!")
    println("You are now 5 years old!")
}
```

```
Happy Birthday, Rover!
You are now 5 years old!
```

```
fun main() {
    println(birthdayGreeting())
}

fun birthdayGreeting(): String {
    val nameGreeting = "Happy Birthday, Rover!"
    val ageGreeting = "You are now 5 years old!"
    return "$nameGreeting\n$ageGreeting"
}
```

```
Happy Birthday, Rover!
You are now 5 years old!
```

```
fun main() {
    println(birthdayGreeting("Elex"))
    println(birthdayGreeting("Eric"))
}

fun birthdayGreeting(name: String): String {
    val nameGreeting = "Happy Birthday, $name!"
    val ageGreeting = "You are now 5 years old!"
    return "$nameGreeting\n$ageGreeting"
}
```

```
Happy Birthday, Elex!
You are now 5 years old!
Happy Birthday, Eric!
You are now 5 years old!
```

```kotlin
fun main() {
    println(birthdayGreeting("Elex", 7))
    println(birthdayGreeting("Eric", 22))
}

fun birthdayGreeting(name: String, age: Int): String {
    val nameGreeting = "Happy Birthday, $name!"
    val ageGreeting = "You are now $age years old!"
    return "$nameGreeting\n$ageGreeting"
}
```

```
Happy Birthday, Elex!
You are now 7 years old!
Happy Birthday, Eric!
You are now 22 years old!
```

```kotlin
fun main() {
    println(birthdayGreeting(name = "Elex", age = 7))
    println(birthdayGreeting(name = "Eric", age = 22)) //Named argument
}

fun birthdayGreeting(name: String, age: Int): String {
    val nameGreeting = "Happy Birthday, $name!"
    val ageGreeting = "You are now $age years old!"
    return "$nameGreeting\n$ageGreeting"
}
```

```
Happy Birthday, Elex!
You are now 7 years old!
Happy Birthday, Eric!
You are now 22 years old!
```

Target platform: JVM

```kotlin
fun main() {
    println(birthdayGreeting(age = 7, name = "Elex")) //Order of arguments changed
    println(birthdayGreeting(name = "Eric", age = 22)) //Named argument
}

fun birthdayGreeting(name: String, age: Int): String {
    val nameGreeting = "Happy Birthday, $name!"
    val ageGreeting = "You are now $age years old!"
    return "$nameGreeting\n$ageGreeting"
}
```

```
Happy Birthday, Elex!
You are now 7 years old!
Happy Birthday, Eric!
You are now 22 years old!
```

Target platform: JVM Running on kotlin v

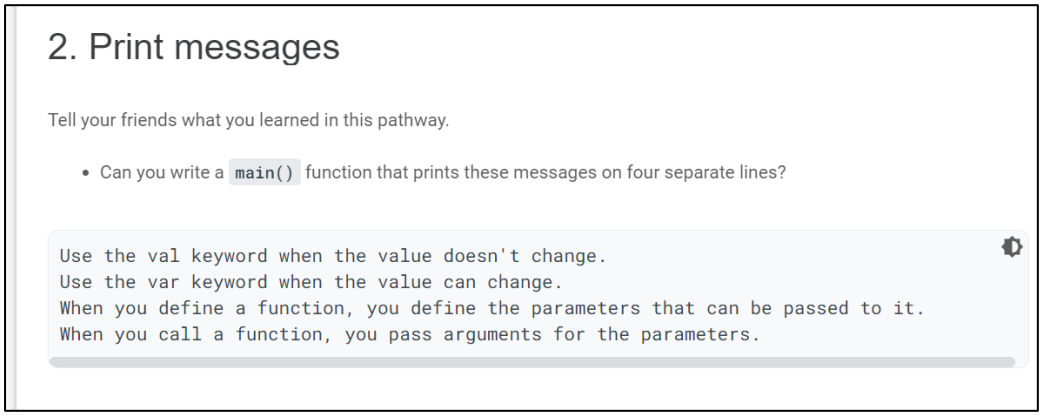
```kotlin
fun main() {
    println(birthdayGreeting(age = 5))
    println(birthdayGreeting("Rex", 2))
}

fun birthdayGreeting(name: String = "Rover", age: Int): String {
    return "Happy Birthday, $name! You are now $age years old!"
}
```

```
Happy Birthday, Rover! You are now 5 years old!
Happy Birthday, Rex! You are now 2 years old!
```

Targe

6 Practice Problems: Kotlin Basics

Question:



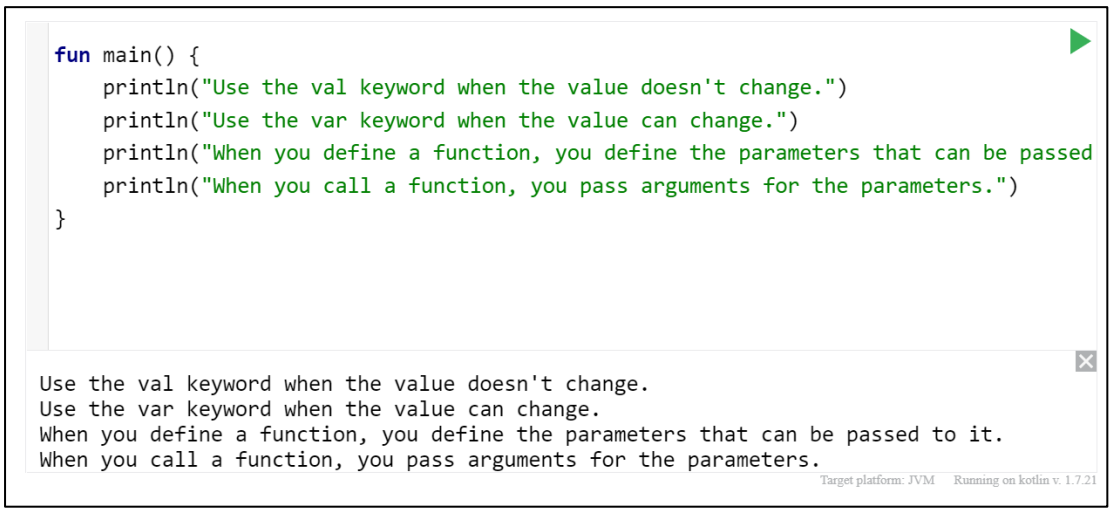
2. Print messages

Tell your friends what you learned in this pathway.

- Can you write a `main()` function that prints these messages on four separate lines?

```
Use the val keyword when the value doesn't change.
Use the var keyword when the value can change.
When you define a function, you define the parameters that can be passed to it.
When you call a function, you pass arguments for the parameters.
```

Answer:



```
fun main() {
    println("Use the val keyword when the value doesn't change.")
    println("Use the var keyword when the value can change.")
    println("When you define a function, you define the parameters that can be passed
    println("When you call a function, you pass arguments for the parameters.")
}
```

```
Use the val keyword when the value doesn't change.
Use the var keyword when the value can change.
When you define a function, you define the parameters that can be passed to it.
When you call a function, you pass arguments for the parameters.
```

Target platform: JVM Running on kotlin v. 1.7.21

Question:

3. Fix compile error

This program prints a message that notifies the user that they received a chat message from a friend.

```
fun main() {
    println("New chat message from a friend'}
}
```

1. Can you figure out the root cause of the compile errors in this program and fix them?
2. Does the code use appropriate symbols to indicate the open and close of the string and function argument?

Hint: You can use Kotlin Playground to run the code and view the compilation errors.

After you fix the errors, the program should compile without errors and print this output:

```
New chat message from a friend
```

Answer:

```
fun main() {
    println("New chat message from a friend")
}
```

```
New chat message from a friend
```

The open and close of the string argument are different, double quote at the start should pair with the same double quote at the end of the String, and also the println function argument should be closed with bracket, not curly bracket.

Question:

4. String templates

This program informs users about the upcoming promotional sale on a particular item. It has a string template, which relies on the `discountPercentage` variable for the percent discount and the `item` variable for the item on sale. However, there are compilation errors in the code.

```
fun main() {
    val discountPercentage: Int = 0
    val offer: String = ""
    val item = "Google Chromecast"
    discountPercentage = 20
    offer = "Sale - Up to $discountPercentage% discount on $item! Hurry up!"

    println(offer)
}
```

1. Can you figure out the root cause of the errors and fix them?
2. Can you determine the output of this program before you run the code in Kotlin Playground?

Hint: Can you re-assign a value to a read-only variable?

After you fix the errors, the program should compile without errors and print this output:

```
Sale - Up to 20% discount on Google Chromecast! Hurry up!
```

Answer:

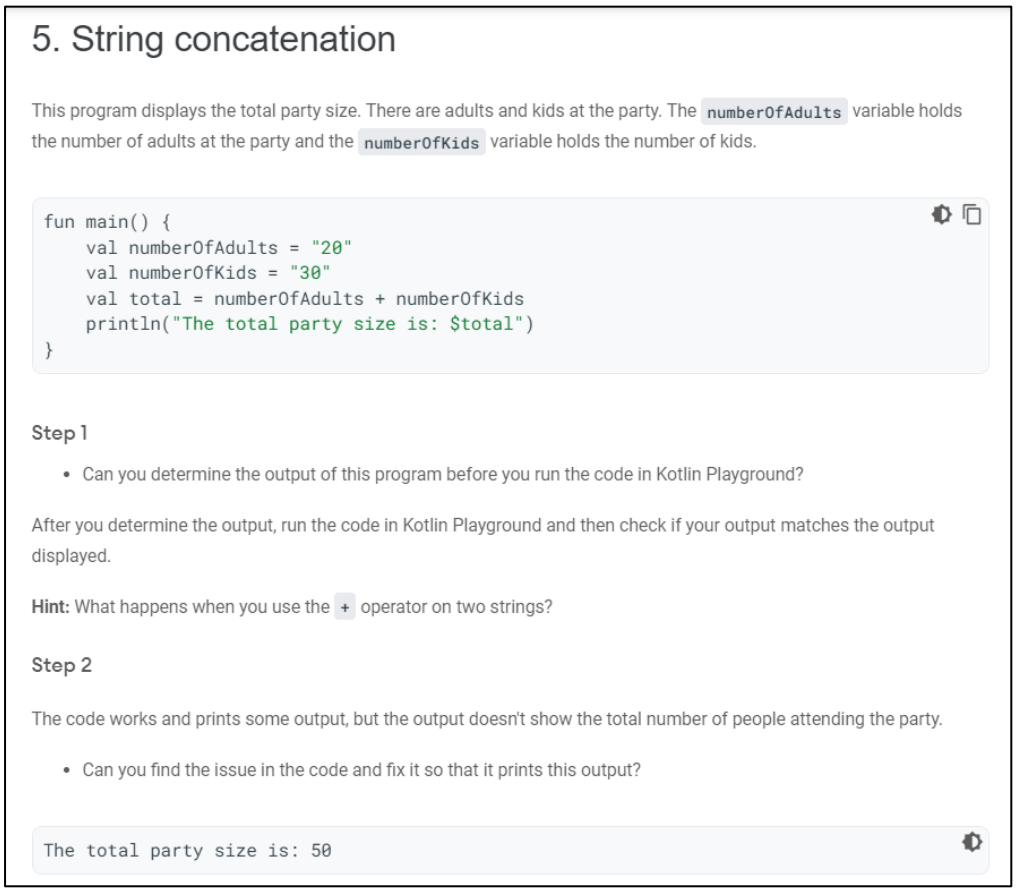
Use “var” keyword instead of “val” if the variable value will be changed later in the code. In this question, the value of the variable “offer” and “discountPercentage” are changed from null and 0 to another value, hence the “val” keyword is not suitable.

```
fun main() {
    var discountPercentage: Int = 0
    var offer: String = ""
    val item = "Google Chromecast"
    discountPercentage = 20
    offer = "Sale - Up to $discountPercentage% discount on $item! Hurry up!"

    println(offer)
}
```

```
Sale - Up to 20% discount on Google Chromecast! Hurry up!
```

Target platform: JVM Running

Question:



5. String concatenation

This program displays the total party size. There are adults and kids at the party. The `numberOfAdults` variable holds the number of adults at the party and the `numberOfKids` variable holds the number of kids.

```
fun main() {
    val numberOfAdults = "20"
    val numberOfKids = "30"
    val total = numberOfAdults + numberOfKids
    println("The total party size is: $total")
}
```

Step 1

- Can you determine the output of this program before you run the code in Kotlin Playground?

After you determine the output, run the code in Kotlin Playground and then check if your output matches the output displayed.

Hint: What happens when you use the `+` operator on two strings?

Step 2

The code works and prints some output, but the output doesn't show the total number of people attending the party.

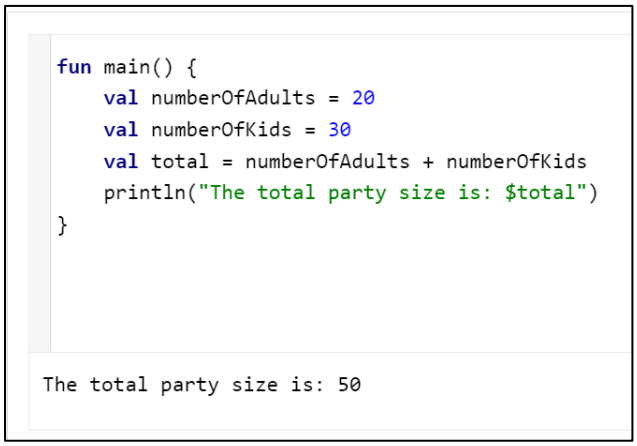
- Can you find the issue in the code and fix it so that it prints this output?

```
The total party size is: 50
```

Answer:

The output should be

The total party size is: 2030



```
fun main() {
    val numberOfAdults = 20
    val numberOfKids = 30
    val total = numberOfAdults + numberOfKids
    println("The total party size is: $total")
}
```

```
The total party size is: 50
```

Remove the double quotes for numberOfAdults and numberOfKids to change their data types to integers instead of using String.

Question:

6. Message formatting

This program displays the total salary that an employee receives this month. The total salary is divided in two parts: the `baseSalary` variable, which the employee receives every month, and the `bonusAmount` variable, which is an additional bonus awarded to the employee.

```
fun main() {
    val baseSalary = 5000
    val bonusAmount = 1000
    val totalSalary = "$baseSalary + $bonusAmount"
    println("Congratulations for your bonus! You will receive a total of $totalSalary(ad
}
```

1. Can you figure out the output of this code before you run it in Kotlin Playground?
2. When you run the code in Kotlin Playground, does it print the output that you expected?

Answer:

Congratulations for your bonus! You will receive a total of 5000 + 1000(additional bonus).

```
fun main() {
    val baseSalary = 5000
    val bonusAmount = 1000
    val totalSalary = "$baseSalary + $bonusAmount"
    println("Congratulations for your bonus! You will receive a total of $totalSalary(
}
```

```
Congratulations for your bonus! You will receive a total of 5000 + 1000(additional bonu
```

Target platform: JVM Running on kotlin v. 1.7.21

Question:

7. Implement basic math operations

In this exercise, you write a program that performs basic math operations and prints the output.

Step 1

This `main()` function contains one compile error:

```
fun main() {
    val firstNumber = 10
    val secondNumber = 5

    println("$firstNumber + $secondNumber = $result")
}
```

- Can you fix the error so that the program prints this output?

```
10 + 5 = 15
```

Step 2

The code works, but the logic for adding two numbers is located within the result variable, making your code less flexible to reuse. Instead, you can extract the addition operation into an `add()` function so that the code is reusable. To do this, update your code with the code listed below. Notice that the code now introduces a new `val` called `thirdNumber` and prints the result of this new variable with `firstNumber`.

```
fun main() {
    val firstNumber = 10
    val secondNumber = 5
    val thirdNumber = 8

    val result = add(firstNumber, secondNumber)
    val anotherResult = add(firstNumber, thirdNumber)

    println("$firstNumber + $secondNumber = $result")
    println("$firstNumber + $thirdNumber = $anotherResult")
}

// Define add() function below this line
```

- Can you define the `add()` function so that the program prints this output?

```
10 + 5 = 15
10 + 8 = 18
```

Step 3

Now you have a reusable function to add two numbers.

- Can you implement the `subtract()` function the same way you implemented the `add()` function? Modify the `main()` function as well to use the `subtract()` function so you can verify that it works as expected.

Hint: Think about the difference between addition, subtraction and other math operations. Start work on the solution code from there.

Answer:

(Step 1)

```
fun main() {
    val firstNumber = 10
    val secondNumber = 5
    val result = firstNumber + secondNumber

    println("$firstNumber + $secondNumber = $result")
}
```

```
10 + 5 = 15
```

(Step 2)

```
fun main() {
    val firstNumber = 10
    val secondNumber = 5
    val thirdNumber = 8

    val result = add(firstNumber, secondNumber)
    val anotherResult = add(firstNumber, thirdNumber)

    println("$firstNumber + $secondNumber = $result")
    println("$firstNumber + $thirdNumber = $anotherResult")
}

// Define add() function below this line
fun add(firstNumber : Int, secondNumber : Int): Int {
    return firstNumber + secondNumber
}
```

```
10 + 5 = 15
10 + 8 = 18
```

(Step 3)

```
fun main() {
    val firstNumber = 10
    val secondNumber = 5
    val thirdNumber = 8

    val result = add(firstNumber, secondNumber)
    val anotherResult = add(firstNumber, thirdNumber)
    val thirdResult = subtract(firstNumber, secondNumber)

    println("$firstNumber + $secondNumber = $result")
    println("$firstNumber + $thirdNumber = $anotherResult")
    println("$firstNumber - $secondNumber = $thirdResult")
}

// Define add() function below this line
fun add(firstNo : Int, secondNo : Int): Int {
    return firstNo + secondNo
}

//subtract
fun subtract(firstNo : Int, secondNo : Int): Int {
    return firstNo - secondNo
}
```

```
10 + 5 = 15
10 + 8 = 18
10 - 5 = 5
```

Question:

8. Default parameters

Gmail has a feature that sends a notification to the user whenever a login attempt is made on a new device.

In this exercise, you write a program that displays a message to users with this message template:

```
There's a new sign-in request on operatingSystem for your Google Account emailId.
```

You need to implement a function that accepts an `operatingSystem` parameter and an `emailId` parameter, constructs a message in the given format, and returns the message.

For example, If the function was called with "`Chrome OS`" for the `operatingSystem` and "`sample@gmail.com`" for the `emailID`, it should return this string:

```
There's a new sign-in request on Chrome OS for your Google Account sample@gmail.com.
```

Step 1

1. Can you implement the `displayAlertMessage()` function in this program so that it prints the output displayed?

```
fun main() {
    val operatingSystem = "Chrome OS"
    val emailId = "sample@gmail.com"

    println(displayAlertMessage(operatingSystem, emailId))
}

// Define your displayAlertMessage() below this line.
```

2. Does your program print this output?

```
There's a new sign-in request on Chrome OS for your Google Account sample@gmail.com.
```

Step 2

Great job! You displayed the message. However, in some cases, you discover that you can't determine the user's operating system. In such cases, you need to specify the operating system name as `Unknown OS`. You can further optimize the code so that you don't need to pass the `Unknown OS` argument each time that the function is called.

1. Can you find a way to optimize the code with this information so that it prints this output?

```
There's a new sign-in request on Unknown OS for your Google Account user_one@gmail.com.

There's a new sign-in request on Windows for your Google Account user_two@gmail.com.

There's a new sign-in request on Mac OS for your Google Account user_three@gmail.com.
```

2. Replace the `main()` function implementation with this one:

```
fun main() {
    val firstUserEmailId = "user_one@gmail.com"

    // The following line of code assumes that you named your parameter as emailId.
    // If you named it differently, feel free to update the name.
    println(displayAlertMessage(emailId = firstUserEmailId))
    println()

    val secondUserOperatingSystem = "Windows"
    val secondUserEmailId = "user_two@gmail.com"

    println(displayAlertMessage(secondUserOperatingSystem, secondUserEmailId))
    println()

    val thirdUserOperatingSystem = "Mac OS"
    val thirdUserEmailId = "user_three@gmail.com"

    println(displayAlertMessage(thirdUserOperatingSystem, thirdUserEmailId))
    println()
}
```

Answer:

```
fun main() {
    val operatingSystem = "Chrome OS"
    val emailId = "sample@gmail.com"

    println(displayAlertMessage(operatingSystem, emailId))
}

// Define your displayAlertMessage() below this line.
fun displayAlertMessage(oS : String, emailId : String): String {
    return "There's a new sign-in request on $oS for your Google Account $emailId."
}
```

```
There's a new sign-in request on Chrome OS for your Google Account sample@gmail.com.
```

Target platform: JVM Running on kotlin v. 1.7.21

```kotlin
fun main() {
    println(displayAlertMessage(emailId = "user_one@gmail.com"))
    println(displayAlertMessage("Windows", "user_two@gmail.com"))
    println(displayAlertMessage("Mac OS", "user_three@gmail.com"))
}

// Define your displayAlertMessage() below this line.
fun displayAlertMessage(oS : String = "Unknown OS", emailId : String): String {
    return "There's a new sign-in request on $oS for your Google Account $emailId."
}
```

```
There's a new sign-in request on Unknown OS for your Google Account user_one@gmail.com.
There's a new sign-in request on Windows for your Google Account user_two@gmail.com.
There's a new sign-in request on Mac OS for your Google Account user_three@gmail.com.
```

Target platform: JVM Running on kotlin v. 1.7.21

```kotlin
fun main() {
    val firstUserEmailId = "user_one@gmail.com"

    // The following line of code assumes that you named your parameter as emailId.
    // If you named it differently, feel free to update the name.
    println(displayAlertMessage(emailId = firstUserEmailId))
    println()

    val secondUserOperatingSystem = "Windows"
    val secondUserEmailId = "user_two@gmail.com"

    println(displayAlertMessage(secondUserOperatingSystem, secondUserEmailId))
    println()

    val thirdUserOperatingSystem = "Mac OS"
    val thirdUserEmailId = "user_three@gmail.com"

    println(displayAlertMessage(thirdUserOperatingSystem, thirdUserEmailId))
    println()
}

// Define your displayAlertMessage() below this line.
fun displayAlertMessage(oS : String = "Unknown OS", emailId : String): String {
    return "There's a new sign-in request on $oS for your Google Account $emailId."
}
```

```
There's a new sign-in request on Unknown OS for your Google Account user_one@gmail.c

There's a new sign-in request on Windows for your Google Account user_two@gmail.com.

There's a new sign-in request on Mac OS for your Google Account user_three@gmail.com
```

Question:

9. Pedometer

The pedometer is an electronic device that counts the number of steps taken. Nowadays, almost all mobile phones, smart watches, and fitness gear come with pedometers built into them. The health and fitness app uses built-in pedometers to calculate the number of steps taken. This function calculates the number of calories that the user burns based on the user's number of steps.

- Can you rename the functions, function parameters, and variables in this program based on best practices?

```
fun main() {
    val Steps = 4000
    val caloriesBurned = PEDOMETERstepsTOcalories(Steps);
    println("Walking $Steps steps burns $caloriesBurned calories")
}

fun PEDOMETERstepsTOcalories(NumberOFStepS: Int): Double {
    val CaloriesBURNEDforEachStep = 0.04
    val TotalCALORIESburned = NumberOFStepS * CaloriesBURNEDforEachStep
    return TotalCALORIESburned
}
```

Answer:

```
fun main() {
    val steps = 4000
    val caloriesBurned = pedometerStepsToCalories(steps);
    println("Walking $steps steps burns $caloriesBurned calories")
}

fun pedometerStepsToCalories(numberOfSteps: Int): Double {
    val caloriesBurnedForEachStep = 0.04
    val totalCaloriesBurned = numberOfSteps * caloriesBurnedForEachStep
    return totalCaloriesBurned
}
```

```
Walking 4000 steps burns 160.0 calories
```

Question:

10. Compare two numbers

Modern mobile phones have a built-in feature that tracks screen time, or the time you spend on your phone each day.

In this exercise, you implement a function that compares the time in minutes that you spent on your phone today versus the time spent yesterday. The function accepts two integer parameters and returns a boolean value.

The first parameter holds the number of minutes that you spent today and the second parameter holds the number of minutes that you spent yesterday. The function returns a `true` value if you spent more time on the phone today compared to yesterday. Otherwise, it returns a `false` value.

For example, if you called the function with these named arguments:

- `timeSpentToday = 300` and `timeSpentYesterday = 250`, the function returns a `true` value.
- `timeSpentToday = 300` and `timeSpentYesterday = 300`, the function returns a `false` value.
- `timeSpentToday = 200` and `timeSpentYesterday = 220`, the function returns a `false` value.

Hint: The `>` comparison operator returns a `true` value if the value before the operator is greater than the value after it. Otherwise, it returns a `false` value.

Answer:

```
fun main() {
    println(timeSpentComparison(timeSpentToday = 300, timeSpentYesterday = 250))
    println(timeSpentComparison(timeSpentToday = 300, timeSpentYesterday = 300))
    println(timeSpentComparison(timeSpentToday = 200, timeSpentYesterday = 220))
}

fun timeSpentComparison(timeSpentToday: Int, timeSpentYesterday: Int): Boolean {
    return timeSpentToday > timeSpentYesterday
}
```

```
true
false
false
```

Target platform: JVM Running on kotlin v

Question:

11. Move duplicate code into a function

This program displays the weather for different cities. It includes the city name, the high and low temperature for the day, and the chance of rain.

```
fun main() {
    println("City: Ankara")
    println("Low temperature: 27, High temperature: 31")
    println("Chance of rain: 82%")
    println()

    println("City: Tokyo")
    println("Low temperature: 32, High temperature: 36")
    println("Chance of rain: 10%")
    println()

    println("City: Cape Town")
    println("Low temperature: 59, High temperature: 64")
    println("Chance of rain: 2%")
    println()

    println("City: Guatemala City")
    println("Low temperature: 50, High temperature: 55")
    println("Chance of rain: 7%")
    println()
}
```

There are many similarities in the code that prints the weather for each city. For example, there are phrases that are repeated multiple times, such as `"City:"` and `"Low temperature:"`. Similar, repeated code creates the risk of errors in your program. For one of the cities, you may have a typo or you may forget one of the weather details.

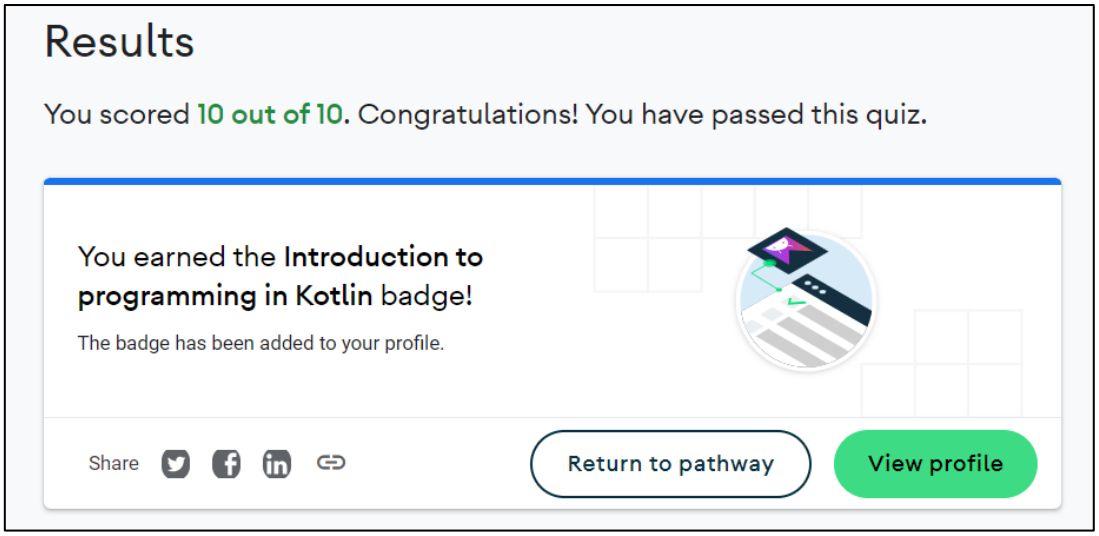
1. Can you create a function that prints the weather details for a single city to reduce the repetition in the `main()` function and then do the same for the remaining cities?
2. Can you update the `main()` function to call the function that you created for each city and pass in the appropriate weather details as arguments?

Answer:

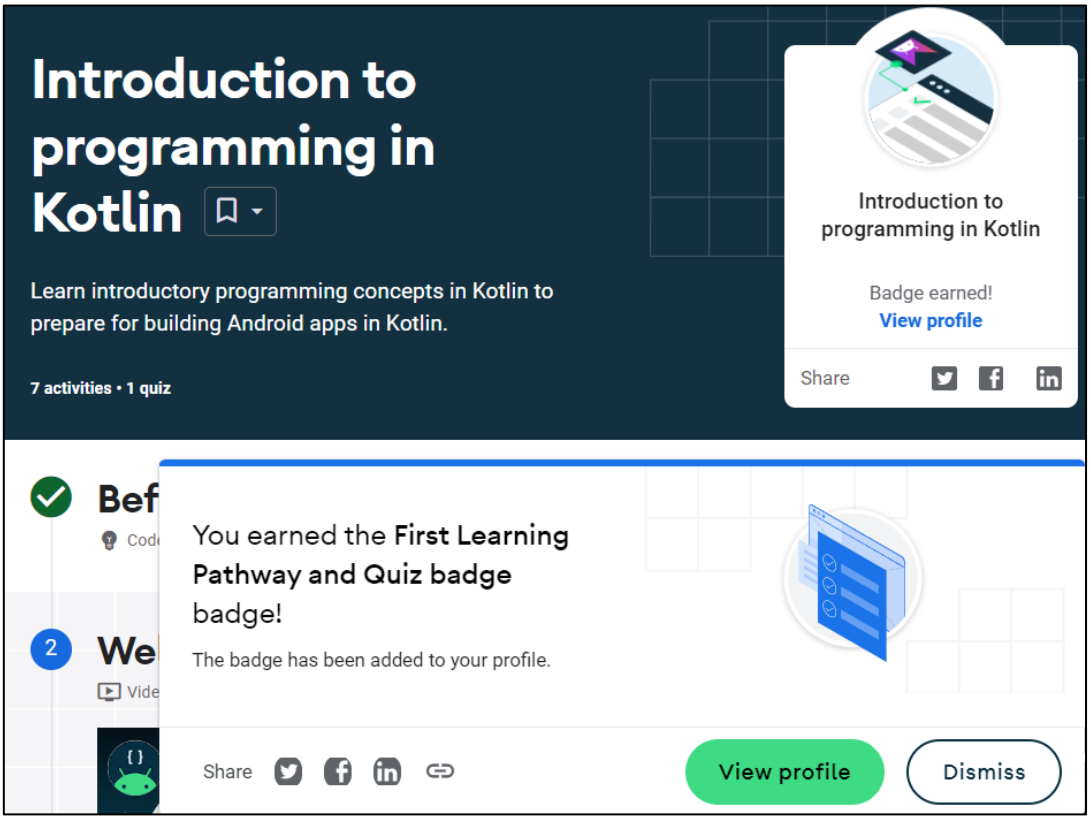
```
fun main() {
    cityDetails("Ankara", 27, 31, 82)

    cityDetails("Tokyo", 32, 36, 10)

    cityDetails("Cape Town", 59, 64, 2)

    cityDetails("Guatemala City", 50, 55, 7)
}


fun cityDetails(city: String, lowTemp: Int, highTemp: Int, chanceOfRain: Int){
    println("City: $city")
    println("Low Temperature: $lowTemp, High Temperature: $highTemp")
    println("Chance of rain: $chanceOfRain%")
    println()
}
```

```
City: Ankara
Low Temperature: 27, High Temperature: 31
Chance of rain: 82%

City: Tokyo
Low Temperature: 32, High Temperature: 36
Chance of rain: 10%

City: Cape Town
Low Temperature: 59, High Temperature: 64
Chance of rain: 2%

City: Guatemala City
Low Temperature: 50, High Temperature: 55
Chance of rain: 7%
```

Quiz


Results
You scored 10 out of 10. Congratulations! You have passed this quiz.
You earned the Introduction to programming in Kotlin badge!
The badge has been added to your profile.
Share
Return to pathway
View profile

Badge


Introduction to programming in Kotlin
Learn introductory programming concepts in Kotlin to prepare for building Android apps in Kotlin.
7 activities • 1 quiz
Introduction to programming in Kotlin
Badge earned!
View profile
Share
You earned the First Learning Pathway and Quiz badge badge!
The badge has been added to your profile.
Share
View profile
Dismiss