



Programación multimedia y dispositivos móviles

Ejercicios Kotlin

Francisco José García Cutillas | 2FPGS_DAM

Índice

Ejercicio 1 3

Ejercicio 1.2 5

Ejercicio 2 6

Ejercicio 3 7

Ejercicio 1

Escribe un programa que te diga si un coche de Uber puede iniciar su recorrido, para esto se necesitan dos cosas, que el conductor esté cerca y que esté disponible, el programa te pedirá dos valores, la distancia del conductor en kilómetros y su disponibilidad, donde false = no disponible y true= disponible, según los datos que insertes imprime lo siguiente:

- Si la distancia es menor o igual a 0.5 km y el conductor está disponible, imprime:
 - “Listo para iniciar recorrido”.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12\" ; if ($?) { kotlinc Ejercicio1.kt -include-runtime -d Ejercicio1.jar } ; if ($?) { java -jar Ejercicio1.jar }
Introduce la distancia del conductor:
0.5
¿Está disponible? (s/n):
s
Listo para iniciar recorrido
```

- Si la distancia es menor o igual a 0.5 km y el conductor está NO disponible, imprime:
 - “Conductor cercano pero no disponible.”.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12\" ; if ($?) { kotlinc Ejercicio1.kt -include-runtime -d Ejercicio1.jar } ; if ($?) { java -jar Ejercicio1.jar }
Introduce la distancia del conductor:
0.3
¿Está disponible? (s/n):
n
Conductor cercano pero no disponible
```

- Si la distancia es mayor a 0.5 km y el conductor está disponible, imprime:
 - “Conductor disponible pero muy lejos, tarifas especiales.”.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12\" ; if ($?) { kotlinc Ejercicio1.kt -include-runtime -d Ejercicio1.jar } ; if ($?) { java -jar Ejercicio1.jar }
Introduce la distancia del conductor:
0.6
¿Está disponible? (s/n):
s
Conductor disponible pero muy lejos, tarifas especiales
```

- Si la distancia es mayor a 0.5 km y el conductor NO está disponible, imprime:
 - “No hay conductores disponibles”.

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12\" ; if ($?) { kotlinc Ejercicio1.kt -include-runtime -d Ejercicio1.jar } ; if ($?) { java -jar Ejercicio1.jar }
Introduce la distancia del conductor:
0.9
¿Está disponible? (s/n):
n
No hay conductores disponibles
```

```

fun llamaUber(distanciaConductor: Double, disponibilidad: Boolean): String {

    var salida: String = "Error"

    if (distanciaConductor <= 0.5 && disponibilidad) {

        salida = "Listo para iniciar recorrido"
    } else if (distanciaConductor <= 0.5 && !disponibilidad) {

        salida = "Conductor cercano pero no disponible"
    } else if (distanciaConductor > 0.5 && disponibilidad) {

        salida = "Conductor disponible pero muy lejos, tarifas especiales"
    } else if (distanciaConductor > 0.5 && !disponibilidad) {

        salida = "No hay conductores disponibles"
    }

    return salida
}

```

Ejercicio1.kt > main

```

1  fun main() {
2
3      ejercicio1()
4      //ejercicio12()
5
6  }
7
8  fun ejercicio1(){
9
10     var distanciaConductor: Double
11     var disponibilidad: Boolean = false
12     var salida: String
13
14     println("Introduce la distancia del conductor:")
15     distanciaConductor = readln().toDouble()
16
17     println("¿Está disponible? (s/n):")
18     var disponibilidadCadena = readln().toString()
19
20     if (disponibilidadCadena.equals("s")) {
21
22         disponibilidad = true
23     } else if (disponibilidadCadena.equals("n")) {
24
25         disponibilidad = false
26     }
27
28     salida = llamaUber(distanciaConductor, disponibilidad)
29     println(salida)
30
31 }
32

```

Ejercicio 1.2

El mismo enunciado de antes pero ahora añadir que si no se cumplen las condiciones de “Listo para iniciar el recorrido” vuelve a pedir los datos de distancia y disponibilidad.

```
fun ejercicio12(){
    var distanciaConductor: Double
    var disponibilidad: Boolean = false
    var salida: String

    do{
        println("Introduce la distancia del conductor:")
        distanciaConductor = readln().toDouble()

        println("¿Está disponible? (s/n):")
        var disponibilidadCadena = readln().toString()

        if (disponibilidadCadena.equals("s")) {
            disponibilidad = true
        } else if (disponibilidadCadena.equals("n")) {
            disponibilidad = false
        }

        salida = llamaUber(distanciaConductor, disponibilidad)

    } while (salida != "Listo para iniciar recorrido")

    println(salida)
}
```

```
P5 C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12> cd "C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Temal\Ejercicios_kotlin\Ejercicio1_12\" ; if ($?) { kotlinc Ejercicio1.kt -include-runtime -d Ejercicio1.jar } ; if ($?) { java -jar Ejercicio1.jar }
Introduce la distancia del conductor:
0.5
¿Está disponible? (s/n):
n
Introduce la distancia del conductor:
0.6
¿Está disponible? (s/n):
s
Introduce la distancia del conductor:
0.5
¿Está disponible? (s/n):
s
Listo para iniciar recorrido
```

Ejercicio 2

Haz un ciclo for y while que obtenga la sumatoria de los números hasta n, ejemplo, para 5 debes obtener 15 (1+2+3+4+5), para 3 debes obtener 6.

Imprime el resultado así: “La suma es 15” usando formatos de String.

```
Ejercicio2 > Ejercicio2.kt > main > numero
1 fun main() {
2
3     var numero: Int = 5
4     var sumatoriaFor: Int = 0
5
6     for(i in 1..numero){
7
8         sumatoriaFor += i
9
10    }
11
12    println("La sumatoria con el bucle for es: $sumatoriaFor")
13
14    var contador: Int = 1
15    var sumatoriaWhile: Int = 0
16
17    while(contador <= numero){
18
19        sumatoriaWhile += contador
20        contador++
21
22    }
23
24    println("La sumatoria con el bucle while es: $sumatoriaWhile")
25
26 }
27
28
```

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio2> cd "C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio2\" ; if ($?) { kotlinc Ejercicio2.kt -include-runtime -d Ejercicio2.jar } ; if ($?) { java -jar Ejercicio2.jar }
La sumatoria con el bucle for es: 15
La sumatoria con el bucle while es: 15
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio2>
```

Ejercicio 3

Escribe un programa que imprima ¿Cómo es el clima de hoy?, dependiendo del número que insertes te imprima los siguientes valores.

1 = "Soleado"

2 = "Nublado"

3 = "Lluvioso"

4 = "Tormentoso"

5 = "Nevado"

Si insertas cualquier otro valor debe imprimir "Pregúntale a Google".

```
Ejercicio3 > Ejercicio3.kt > main
1 fun main(){
2
3     var numEntrada: Int
4
5     println("¿Cómo es el clima de hoy?")
6     numEntrada = readln().toInt()
7
8     when(numEntrada){
9
10        1 -> println("Soleado")
11        2 -> println("Nublado")
12        3 -> println("Lluvioso")
13        4 -> println("Tormentoso")
14        5 -> println("Nevado")
15        else -> println("Pregúntale a Google")
16
17    }
18
19 }
```

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
1
Soleado
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "C:\Users\Fran\OneDrive\Documents\2FPGS_DAM\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
2
Nublado
```

```
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
3
Lluvioso
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
4
Tormentoso
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
5
Nevado
PS C:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3> cd "c:\Users\Fran\OneDrive\Documents\2FPGS_DAW\Programacion_Multimedia\Tareas\Tema1\Ejercicios_kotlin\Ejercicio3\" ; if ($?) { kotlinc Ejercicio3.kt -include-runtime -d Ejercicio3.jar } ; if ($?) { java -jar Ejercicio3.jar }
¿Cómo es el clima de hoy?
7
Pregúntale a Google
```