

11 - Estructura repetitiva for y expresiones de rango

La estructura for tiene algunas variantes en Kotlin, en este concepto veremos la estructura for con expresiones de rango.

Veamos primero como se define y crea un rango.

Un rango define un intervalo que tiene un valor inicial y un valor final, se lo define utilizando el operador ..

Ejemplos de definición de rangos:

```
val unDigito = 1..9
val docena = 1..12
var letras = 'a'..'z'
```

Si necesitamos conocer si un valor se encuentra dentro de un rango debemos emplear el operador in o el !in:

```
val docena = 1..12

if (5 in docena)
    println("el 5 está en el rango docena")

if (18 !in docena)
    println("el 18 no está en el rango docena")
```

Los dos if se verifican como verdadero.

Veamos ahora la estructura repetitiva for empleando un rango para repetir un bloque de comandos.

Problema 1

Realizar un programa que imprima en pantalla los números del 1 al 100.

Proyecto47 - Principal.kt

```
fun main(parametro: Array<String>) {
    for(i in 1..100)
        println(i)
}
```

La variable i se define de tipo Int por inferencia ya que el rango es de 1..100

En la primer ejecución del ciclo repetitivo i almacena el valor inicial del rango es decir el 1. Luego de ejecutar el bloque la variable i toma el valor 2 y así sucesivamente.

Problema 2

Desarrollar un programa que permita la carga de 10 valores por teclado y nos muestre posteriormente la suma de los valores ingresados y su promedio. Este problema ya lo desarrollamos empleando el while, lo resolveremos empleando la estructura repetitiva for.

Proyecto48 - Principal.kt

```

fun main(parametro: Array<String>) {
    var suma = 0
    for(i in 1..10) {
        print("Ingrese un valor:")
        val valor = readln().toInt()
        suma += valor
    }
    println("La suma de los valores ingresados es $suma")
    val promedio = suma / 10
    println("Su promedio es $promedio")
}

```

Como podemos ver la variable `i` dentro del ciclo `for` no se la utiliza dentro del bloque repetitivo y solo nos sirve para que el bloque contenido en el `for` se repita 10 veces.

Cuando sabemos cuantas veces se debe repetir un bloque de instrucciones es más conveniente utilizar el `for` que un `while` donde nosotros debemos definir, inicializar e incrementar un contador.

Problema 3

Escribir un programa que lea 10 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 7 y cuántos menores.

Proyecto49 - Principal.kt

```

fun main(parametro: Array<String>) {
    var aprobados = 0
    var reprobados = 0
    for(i in 1..10) {
        print("Ingrese nota:")
        val nota = readln().toInt()
        if (nota >= 7)
            aprobados++
        else
            reprobados++
    }
    println("Cantidad de alumnos con notas mayores o iguales a 7: $aprobados")
    println("Cantidad de alumnos con notas menores a 7: $reprobados")
}

```

Nuevamente como necesitamos cargar 10 valores por teclado disponemos un `for`:

```
for(i in 1..10) {
```

Problema 4

Desarrollar un programa que cuente cuantos múltiplos de 3, 5 y 9 hay en el rango de 1 a 10000 (No se deben cargar valores por teclado)

Proyecto50 - Principal.kt

```

fun main(parametro: Array<String>) {
    var mult3 = 0
    var mult5 = 0
    var mult9 = 0
    for(i in 1..10000) {
        if (i % 3 == 0)
            mult3++
        if (i % 5 == 0)
            mult5++
        if (i % 8 == 0)
            mult9++
    }
    println("Cantidad de múltiplos de 3: $mult3")
    println("Cantidad de múltiplos de 5: $mult5")
    println("Cantidad de múltiplos de 9: $mult9")
}

```

En este problema recordemos que el contador *i* toma la primer vuelta el valor 1, luego el valor 2 y así sucesivamente hasta el valor 10000.

Si queremos averiguar si un valor es múltiplo de 3 obtenemos el resto de dividirlo por 3 y si dicho resultado es cero luego podemos inferir que el número es múltiplo de 3.

Problema 5

Escribir un programa que lea *n* números enteros y calcule la cantidad de valores pares ingresados.

Este tipo de problemas también se puede resolver empleando la estructura repetitiva *for* ya que cuando expresamos el rango podemos disponer un nombre de variable.

Proyecto51 - Principal.kt

```

fun main(parametros: Array<String>) {
    var cant = 0
    print("Cuantos valores ingresará para analizar:")
    val cantidad = readln().toInt()
    for(i in 1..cantidad) {
        print("Ingrese valor:")
        val valor = readln().toInt()
        if (valor % 2 ==0)
            cant++
    }
    println("Cantidad de pares: $cant")
}

```

Como vemos en la estructura repetitiva *for* el valor final del rango dispusimos la variable *cantidad* en lugar de un valor fijo:

```
for(i in 1..cantidad) {
```

Hasta que no se ejecuta el programa no podemos saber cuantas veces se repetirá el for. La cantidad de repeticiones dependerá del número que cargue el operador.

Variantes del for

Si necesitamos que la variable del for no reciba todos los valores comprendidos en el rango sino que avance de 2 en 2 podemos utilizar la siguiente sintaxis:

```
for(i in 1..10 step 2)
    println(i)
```

Se imprimen los valores 1, 3, 5, 7, 9

Si necesitamos que la variable tome el valor 10, luego el 9 y así sucesivamente, es decir en forma inversa debemos utilizar la siguiente sintaxis:

```
for(i in 10 downTo 1)
    println(i)
```

Se imprimen los valores 10, 9, 8, 7, 6, 5, 4, 3, 2, 1

También podemos utilizar el step con el downTo:

```
for(i in 10 downTo 1 step 2)
    println(i)
```

Se imprimen los valores 10, 8, 6, 4, 2

Podemos utilizar la siguiente variante de for que tendrá sentido cuando recorramos arreglos:

```
for(x in 0 until 10)
    println(x)
```

Se imprimen los valores de 0 a 9 (No incluye el valor 10)

Problemas propuestos

- Confeccionar un programa que lea n pares de datos, cada par de datos corresponde a la medida de la base y la altura de un triángulo. El programa deberá informar:
 - a) De cada triángulo la medida de su base, su altura y su superficie (la superficie se calcula multiplicando la base por la altura y dividiendo por dos).
 - b) La cantidad de triángulos cuya superficie es mayor a 12.
- Desarrollar un programa que solicite la carga de 10 números e imprima la suma de los últimos 5 valores ingresados.
- Desarrollar un programa que muestre la tabla de multiplicar del 5 (del 5 al 50)
- Confeccionar un programa que permita ingresar un valor del 1 al 10 y nos muestre la tabla de multiplicar del mismo (los primeros 12 términos)
Ejemplo: Si ingresó 3 deberá aparecer en pantalla los valores 3, 6, 9, hasta el 36.
- Realizar un programa que lea los lados de n triángulos, e informar:
 - a) De cada uno de ellos, qué tipo de triángulo es: equilátero (tres lados iguales), isósceles (dos lados iguales), o escaleno (ningún lado igual)
 - b) Cantidad de triángulos de cada tipo.
- Escribir un programa que pida ingresar coordenadas (x,y) que representan puntos en el plano.
Informar cuántos puntos se han ingresado en el primer, segundo, tercer y cuarto cuadrante. Al comenzar el programa se pide que se ingrese la cantidad de puntos a procesar.

- Se realiza la carga de 10 valores enteros por teclado. Se desea conocer:
 - a) La cantidad de valores ingresados negativos.
 - b) La cantidad de valores ingresados positivos.
 - c) La cantidad de múltiplos de 15.
 - d) El valor acumulado de los números ingresados que son pares.

Proyecto52

```
fun main(parametro: Array<String>) {
    var cantidad = 0
    print("Cuantos triángulos procesará:")
    val n = readln().toInt()
    for(i in 1..n) {
        print("Ingrese el valor de la base:")
        val base = readln().toInt()
        print("Ingrese el valor de la altura:")
        val altura = readln().toInt()
        val superficie = base * altura / 2
        println("La superficie es de $superficie")
        if (superficie > 12)
            cantidad++
    }
    print("La cantidad de triángulos con superficie superior a 12 son: $cantidad")
}
```

Proyecto53

```
fun main(parametro: Array<String>) {
    var suma = 0
    for(i in 1..10) {
        print("Ingrese un valor entero:")
        val valor = readln().toInt()
        if (i > 5)
            suma += valor
    }
    print("La suma de los últimos 5 valores es: $suma");
}
```

Proyecto54

```
fun main(parametro: Array<String>) {
    for(tabela5 in 5..50 step 5)
        println(tabela5)
}
```

Proyecto55

```
fun main(argumento: Array<String>) {
    print("Ingrese un valor entre 1 y 10:")
    val valor = readln().toInt()
    for(i in valor..valor*12 step valor)
        println(i)
}
```

Proyecto56

```
fun main(argumento: Array<String>) {
    var cant1 = 0
    var cant2 = 0
    var cant3 = 0
    print("Ingrese la cantidad de triángulos:")
    val n = readln().toInt()
    for(i in 1..n) {
```

```

    print("Ingrese lado 1:")
    val lado1 = readln().toInt()
    print("Ingrese lado 2:")
    val lado2 = readln().toInt()
    print("Ingrese lado 3:")
    val lado3 = readln().toInt()
    if (lado1 == lado2 && lado1 == lado3) {
        println("Es un triángulo equilátero.")
        cant1++
    } else
        if (lado1 == lado2 || lado1 == lado3 || lado2 == lado3) {
            println("Es un triángulo isósceles.")
            cant2++
        } else {
            println("Es un triángulo escaleno.")
            cant3++
        }
    }
    println("Cantidad de triángulos equiláteros: $cant1")
    println("Cantidad de triángulos isósceles: $cant2")
    println("Cantidad de triángulos escalenos: $cant3")
}

```

Proyecto57

```

fun main(parametro: Array<String>) {
    var cant1 = 0
    var cant2 = 0
    var cant3 = 0
    var cant4 = 0
    print("Cantidad de puntos a ingresar:")
    val cantidad = readln().toInt()
    for(i in 1..cantidad) {
        print("Ingrese coordenada x:")
        val x = readln().toInt()
        print("Ingrese coordenada y:")
        val y = readln().toInt()
        if (x > 0 && y > 0)
            cant1++
        else
            if (x < 0 && y > 0)
                cant2++;
            else
                if (x < 0 && y < 0)
                    cant3++
                else
                    if (x > 0 && y < 0)
                        cant4++
    }
    println("Cantidad de puntos en el primer cuadrante: $cant1")
    println("Cantidad de puntos en el segundo cuadrante: $cant2")
    println("Cantidad de puntos en el tercer cuadrante: $cant3")
    println("Cantidad de puntos en el cuarto cuadrante: $cant4")
}

```

Proyecto58

```

fun main(parametro: Array<String>) {
    var negativos = 0
    var positivos = 0
    var mult15 = 0
    var sumapares = 0
    for(i in 1..10) {

```

```
print("Ingrese valor:")
val valor = readln().toInt()
if (valor < 0)
    negativos++
else
    if (valor > 0)
        positivos++
    if ( valor % 15 == 0)
        mult15++
    if (valor % 2 == 0)
        sumapares+=valor
}
println("Cantidad de valores negativos: $negativos")
println("Cantidad de valores positivos: $positivos")
println("Cantidad de valores múltiplos de 15: $mult15")
println("Suma de los valores pares: $sumapares")
}
```