

13 - Estructura condicional when con argumento

Tenemos una segunda forma de utilizar la sentencia when en el lenguaje Kotlin pasando un argumento inmediatamente después de la palabra clave when.

Problema 1

Ingresar un valor entero comprendido entre 1 y 5. Mostrar el mismo en castellano.

Proyecto66 - Principal.kt

```
fun main(parameto: Array<String>) {  
    print("Ingrese un valor entero entre 1 y 5:")  
    val valor = readln().toInt()  
    when (valor) {  
        1 -> print("uno")  
        2 -> print("dos")  
        3 -> print("tres")  
        4 -> print("cuatro")  
        5 -> print("cinco")  
        else -> print("valor fuera de rango")  
    }  
}
```

Como vemos disponemos luego de la palabra clave when entre paréntesis una variable. Se verifica el contenido de la variable "valor" con cada uno de los datos indicados.

Por ejemplo si cargamos por teclado el 3 luego son falsos los dos primeros caminos:

```
1 -> print("uno")  
2 -> print("dos")
```

Pero el tercer camino se verifica verdadero y pasa a ejecutar los comandos dispuestos después del operador ->

```
3 -> print("tres")
```

Problema 2

Ingresar un valor entero positivo comprendido entre 1 y 10000. Imprimir un mensaje indicando cuantos dígitos tiene.

Proyecto67 - Principal.kt

```

fun main(parametro: Array<String>){
    print("Ingrese un valor entero positivo comprendido entre 1 y 99999:")
    val valor = readln().toInt()
    when (valor){
        in 1..9 -> print("Tiene 1 dígito")
        in 10..99 -> print("Tiene 2 dígitos")
        in 100..999 -> print("Tiene 3 dígitos")
        in 1000..9999 -> print("Tiene 4 dígitos")
        in 10000..99999 -> print("Tiene 5 dígitos")
        else -> print("No se encuentra comprendido en el rango indicado")
    }
}

```

También en Kotlin podemos comprobar si una variable se encuentra comprendida en un rango determinado utilizando la palabra `in` y el rango respectivo.

Problema 3

Ingresar 10 valores enteros por teclado. Contar cuantos de dichos valores ingresados fueron cero y cuantos 1,5 o 10.

Proyecto68 - Principal.kt

```

fun main(parametro: Array<String>){
    var cant1 = 0
    var cant2 = 0
    for(i in 1..10) {
        print("Ingrese un valor entero:")
        val valor = readln().toInt()
        when (valor){
            0 -> cant1++
            1, 5, 10 -> cant2++
        }
    }
    println("Cantidad de números 0 ingresados: $cant1")
    println("Cantidad de números 1,2 o 3 ingresados: $cant2")
}

```

En este problema disponemos en uno de los caminos del `when` una lista de valores separados por coma, si alguno de ellos coincide con la variable "valor" luego se incrementa "cant2":

```
1, 5, 10 -> cant2++
```

Problema propuesto

- Realizar la carga de la cantidad de hijos de 10 familias. Contar cuantos tienen 0,1,2 o más hijos. Imprimir dichos contadores.

Proyecto69

```
fun main(parametro: Array<String>) {  
    var conta1 = 0  
    var conta2 = 0  
    var conta3 = 0  
    var conta4 = 0  
    for(i in 1..10) {  
        print("Ingrese la cantidad de hijos de la familia:")  
        val cantidad = readln().toInt()  
        when (cantidad) {  
            0 -> conta1++  
            1 -> conta2++  
            2 -> conta3++  
            else -> conta4++  
        }  
    }  
    println("Cantidad de familias con 0 hijos: $conta1")  
    println("Cantidad de familias con 1 hijos: $conta2")  
    println("Cantidad de familias con 2 hijos: $conta3")  
    println("Cantidad de familias con más de 2 hijos: $conta4")  
}
```