

```

fun main() {
    val myCool = cool() //osztály deklarálása
    myCool.basicWhileLoop() //függvény indítása
    println()
    myCool.infinityWhileLoop()
    println()
    myCool.loopwidthConditionAtEnd()
}

```

```

import java.util.*

class cool {
    fun basicWhileLoop() {
        var count = 0
        while (count < 5) {
            println("Count : $count")
            count++ //itt növelni kell az értéket egyel
        }
    }

    fun infinityWhileLoop() {
        var x = 1
        while (true) { //végtelen for loop Kotlinban csak ez van
            println("Value is x : $x")
            x++ //it is növelni kell
            //x+= 2 //akkor 2 vel növelnénk
            //kilépés a végtelen ciklusból
            if (x > 10) {
                break
            }
        }
    }

    fun loopwidthUserInput() {
        val scanner = Scanner(System.`in`) //vagy String.in
        var userInput: String

        while (true) {
            println("Enter a word(or 'Exit' to quit): ")
            userInput = scanner.nextLine()
            if (userInput.toLowerCase() == "exit") { //lowercase az új
                break
            }
        }

        println("Your entered : $userInput ")
    }

    fun loopwidthConditionAtEnd(){
        var countdown=5
        while (countdown > 0){
            println("Countdown : $countdown")
        }
    }
}

```

```

        countdown-- //csökkentem
    }
}
fun Matrixwhile(){
    var row = 1 //row jelentése sor
    while (row <= 4) {           // sorok száma
        var count = 1 //count darabszám számláló
        while (count <= row) {   // adott sorban hány "Ati"
            print("Ati")
            count++
        }
        println()               // új sor
        row++
    }
}
fun Matrixwhile2(){
    var row = 4
    while (row >= 1) {           // sorok száma
        var count = 1
        while (count <= row) {   // adott sorban hány "Ati"
            print("Ati")
            count++
        }
        println()               // új sor
        row--
    }
}
}
}

```

Main fájl és egy másik osztály másik fájlban

```

import java.util.Scanner

class userN {
    fun inputUser(){
        val scanner = Scanner(System.`in`)
        var userInput : String

        while (true) {
            println("Enter a word(or 'Exit' to quit): ")
            userInput = scanner.nextLine()
            if (userInput.lowercase() == "exit") {
                break
            }
            println("Your entered : $userInput ")
        }
    }
}

```

main2 fájl

```
fun main(){  
    val Ccool = cool()  
    //Ccool.loopwidthUserInput()  
    //Ccool.loopwidthConditionAtEnd()  
    Ccool.Matrixwhile()  
    println()  
    Ccool.Matrixwhile2()  
}
```