```
fun main() {
    loopForRange()
    println()
    forLoopWithCollection()
    println()
    forLoopWithCollection2()
    println()
    forLoopWithIndicates()
    println()
    forLoopWithSteps()
    println()
    forLoopWithRangesAndFilters()
    println()
    forLoopWithSteps2()
    println()
    matrixIncreasing()
    println()
    matrixIncreasing2()
    println()
    matrixDecrasing()
}
//for loop osztály
fun loopForRange(){
    for (i : Int in 1..5 ){ //i 1-5 ig kiírja a számokat
        println("Count: $i")
    }
}
//for loop 2 osztály
fun forLoopWithCollection(){
    val fruits : List<String>  = listOf("Apple, Banana, Orange")//lista
    //így egymás mellé írja listOf("Apple, Banana, Orange")
    //for loop
    for (item : String  in fruits){
        println("fruits: $item")
    }
}
//for loop 3 osztály
fun forLoopWithCollection2(){
    val fruits : List<String>  = listOf("Apple", "Banana", "Orange")//lista
    //így egymás allá írja írja istOf("Apple", "Banana", "Orange")
    //for loop
    for (item : String  in fruits){
        println("fruit: $item")
    }
}
fun forLoopWithIndicates(){
    val colors : List<String> = listOf("Red", "Green", "Blue")
    //
    //withIndex() visszaadja az elemek indexét és értékét is egyszerre
    for ((index,color )in colors.withIndex()){
        println("Color at index $index: $color ")
    }
}
fun forLoopWithSteps(){
    for (i in 10 downTo 1 step 2 ){
        //downto 1 lefelé megy 1-ig
        //step 2 minden második értéket irja ki(10,8,6 stb)
        println("Countdown : $i")
    }
}
fun forLoopWithSteps2(){
    for (i in 0..10 step 2  ){
        println("Countdown : $i")
    }
}
fun forLoopWithRangesAndFilters(){
    for (num in 0..20){
        if (num %2 == 0){
        //az aktuális szám osztható-e 2-vel
        //Ha osztható, akkor a maradék 0 -> == 0
            println("Event number : $num")
        }
    }
}
fun matrixIncreasing(){
    for (tetel in 1..4){//sorok száma
      for (darabSzam in 1.. tetel)  {
          print("Ati") //a kiiras
      }
        println()//uj sor
    }
}
fun matrixIncreasing2(){
    val szoveg :String = "Ati"
    for (tetel in 1..4){//sorok száma
        for (darabSzam in 1.. tetel)  {
            print(szoveg)
        }
        println()//uj sor
    }
}
fun matrixDecrasing(){
    for (i in 4 downTo 1){
        for (j in 1..i){
            print("Ati")
        }
        println()//uj sor
    }
}
```

Kimenete:

```
Count: 1
Count: 2
Count: 3
```

```
Count: 4
Count: 5

fruits: Apple, Banana, Orange

fruit: Apple
fruit: Banana
fruit: Orange

Color at index 0: Red
Color at index 1: Green
Color at index 2: Blue

Countdown : 10
Countdown : 8
Countdown : 6
Countdown : 4
Countdown : 2

Event number : 0
Event number : 2
Event number : 4
Event number : 6
Event number : 8
Event number : 10
Event number : 12
Event number : 14
Event number : 16
Event number : 18
Event number : 20

Countdown : 0
Countdown : 2
Countdown : 4
Countdown : 6
Countdown : 8
Countdown : 10

Ati
AtiAti
AtiAtiAti
AtiAtiAtiAti

Ati
AtiAti
AtiAtiAti
AtiAtiAtiAti

AtiAtiAtiAti
AtiAtiAti
AtiAti
Ati

Process finished with exit code 0
```