

Kotlin - różnice w porównaniu do poznanych języków

Mateusz Wiśniewski

Deklaracje funkcji

```
fun main() {  
    println("Mateusz Wiśniewski")  
}
```

Mateusz Wiśniewski

Deklaracje zmiennych – Kotlin automatycznie wykrywa co mamy w zmiennej, więc nie zawsze musimy podać typ zmiennej

```
val a: Int = 10  
val b = 20  
val c = "Mateusz Wiśniewski"  
val d: String = "Mateusz Wiśniewski"  
val e = 0.11111111111111111111111111111111;  
val f: Double = 0.11111111111111111111111111111111;  
fun main() {  
    println("${a}, ${b}\n ${c}\n ${d}\n ${e}\n ${f}")  
}
```

10, 20
Mateusz Wiśniewski
Mateusz Wiśniewski
0.11111111111111111111111111111111
0.11111111111111111111111111111111

If jest wyrażeniem, więc możemy przypisać jego wartość

```
val a: Int = 10  
val b = 20  
fun main() {  
    val max = if (a > b) a else b  
    println(max)  
}
```

20

```
val a: Int = 10  
val b = 20  
fun main() {  
    val max = if (a > b) 1 else 2  
    println(max)  
}
```

2

Pętle for, można się odwoływać po indexach, w drugim przykładzie pętla wyświetla index wraz z wartością.

```
6 fun main() {
7   val arr = arrayOf(101, 102, 103, 104, 105)
8   for (i in arr) {
9     println(i)
10  }
11  for (i in arr.indices) {
12    print("$i, ")
13  }
14  for ((i, v) in arr.withIndex()) {
15    println("$i => $v")
16  }
17 }
18 }
```

Compilation completed successfully

```
101
102
103
104
105
0, 1, 2, 3, 4, 0 => 101
1 => 102
2 => 103
3 => 104
4 => 105
1 2 3 4 5 5 4 3 2 1 1 3 5 1 2 3 4
```

Odwrotna pętla for, pętla odliczająca co 2 kroki oraz od 1 „przed” 5

```
9
10 fun main() {
11   for (i in 1..5) {
12     print("$i ")
13   }
14   print("\n")
15   for (i in 5 downTo 1) {
16     print("$i ")
17   }
18   print("\n")
19   for (i in 1..5 step 2) {
20     print("$i ")
21   }
22   print("\n")
23   for (i in 1 until 5) {
24     print("$i ")
25   }
26 }
```

Compilation completed successfully

```
1 2 3 4 5
5 4 3 2 1
1 3 5
1 2 3 4
```

Pętle foreach

```
10 fun main() {
11   val arr = arrayOf(101, 102, 103, 104, 105)
12   arr.forEach {
13     print(it)
14     print("\n")
15   }
16
17   repeat(5) {
18     print("$it ")
19   }
20
21 }
```

Compilation completed successfully

```
101
102
103
104
105
0 1 2 3 4
```

When wykonuje zadanie odpowiednie dla Switch

```
9
10 fun main() {
11     val number = 4
12     when(number) {
13         0 -> println("Invalid number")
14         1, 2 -> println("Number too low")
15         3 -> println("Number correct")
16         4 -> println("Number too high, but acceptable")
17         else -> println("Number too high")
18     }
19 }
```

Compilation completed successfully

Number too high, but acceptable