

## 27. Kotlin (Introduction)

- Обработка данных на языке Kotlin
- Обработка коллекций
- Использование Lambda
- Ключевое слово data

### Использование элементов ListOf, MutableListOf:

```
fun loop() {
    val items = listOf("apple", "banana", "orange")
    for (item in items)
        println(item)
}

//-----

fun loop2() {
    val items = mutableListOf<Int>(10, 20, 30)
    var index = 0
    while (index < items.size) {
        println("Item at ${index} is ${items[index]}")
        index++
    }
}
```

Диапазоны, тип vararg в качестве аргумента функции, аргументы по умолчанию:

```
fun range() {
    for (i in 0..100 step 20)
        println(i)
}

fun moreArgs(name:String, surname:String = "Bobkins",
toUpperCase:Boolean=false) {
    if (toUpperCase)
        println(name.toUpperCase() + " " + surname.toUpperCase())
}

fun printEven(vararg operands:Int) {
    operands.forEach { operand -> println(operand * 10) }
}
```

### Обработка NullableTypes объектов:

```
fun elvisAndNullPointer() {
    val a:String?
    a = if ((0..10).random() > 5) "String" else null
    val l = a?.length ?: -1
    val t = a!!.length
    println(l)
    println(t)
}
```

## WhenExpression (аналог SwitchCase):

```
fun whenUsing(argument:Any) {  
    when(argument) {  
        is Int -> println("Введено число " + argument)  
        is String -> println("Введена строка " + argument + " длиной  
${argument.length} символов")  
        else -> print("Введен другой аргумент" + argument)  
    }  
}
```

Примеры обработки коллекций: (forEach(), map(), filter(), reduce(), sorted(), partition(), groupBy())

```
fun collection() {  
    val a = listOf(1,2,3,4,5,6,7,8,9,10)  
    a.forEach {elem -> println(elem)}  
    println(a.map{ elem -> elem*2})  
    println(a.filter { elem -> elem%2 ==0})  
    println(a.reduce{sum, elem -> sum + elem})  
    println(a.sortedByDescending { it })  
    println(a.any{it>10})  
    println(a.all{it<10})  
    println(a.sum())  
  
    val numbers = listOf(1,3,-6,8,-10,22)  
    val (positive, negative) = numbers.partition { it > 0 }  
    println(positive)  
    println(negative)  
  
    val items = listOf("a","b","ba", "ccc", "ad").groupBy { it.length }  
    println(items)  
}
```

Ключевое слово data (автоматическое создание сеттеров, геттеров, equals(), hashCode(), toString() и др.):

```
data class Human(val name: String, val surname: String, var age:Int ) {  
    init {  
        println("Human is Created: ${name}")  
    }  
}
```