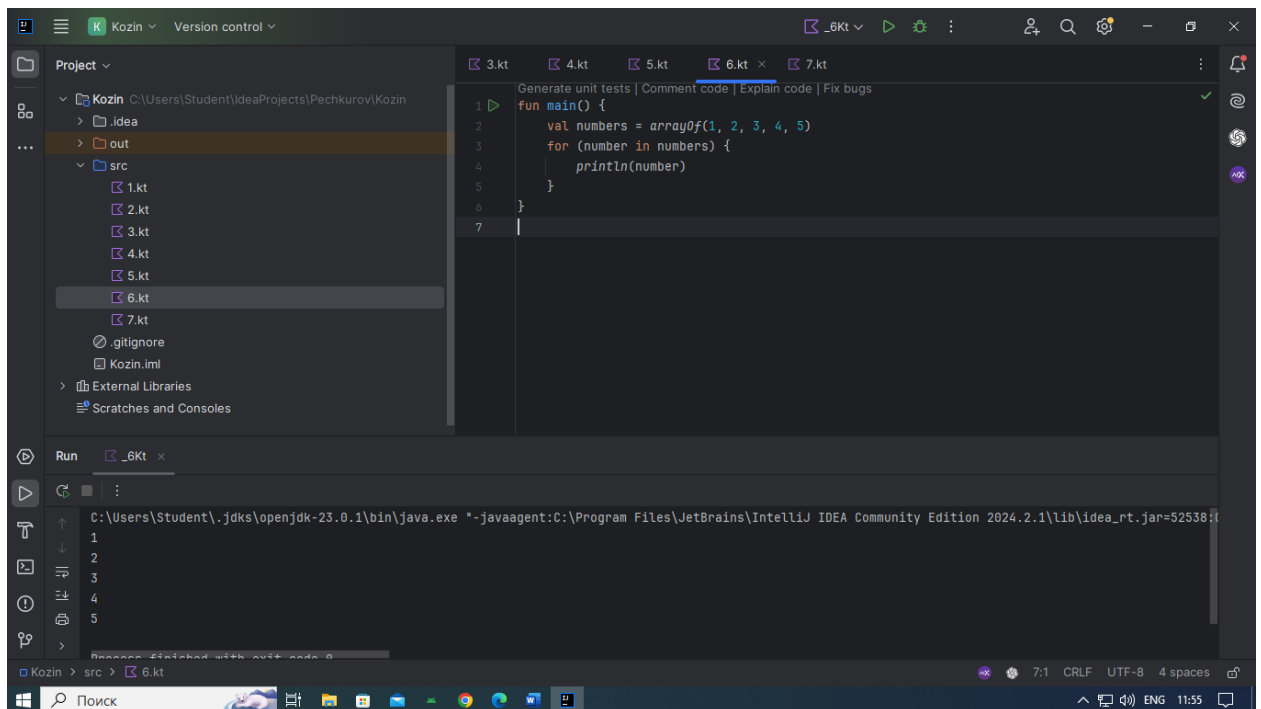


## Практическая работа 6

### Массивы

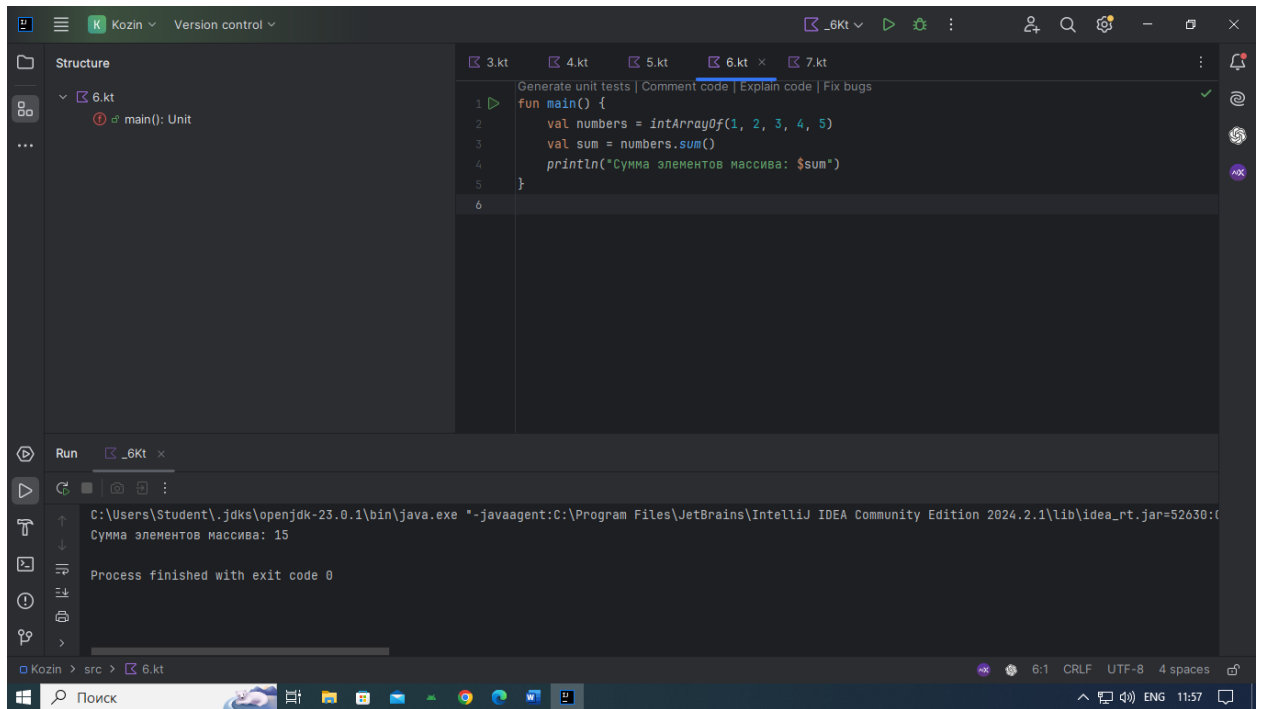
1. Создание и вывод элементов: Создайте массив из 5 целых чисел и выведите их на экран

```
fun main() {  
    val numbers = arrayOf(1, 2, 3, 4, 5)  
    for (number in numbers) {  
        println(number)  
    }  
}
```



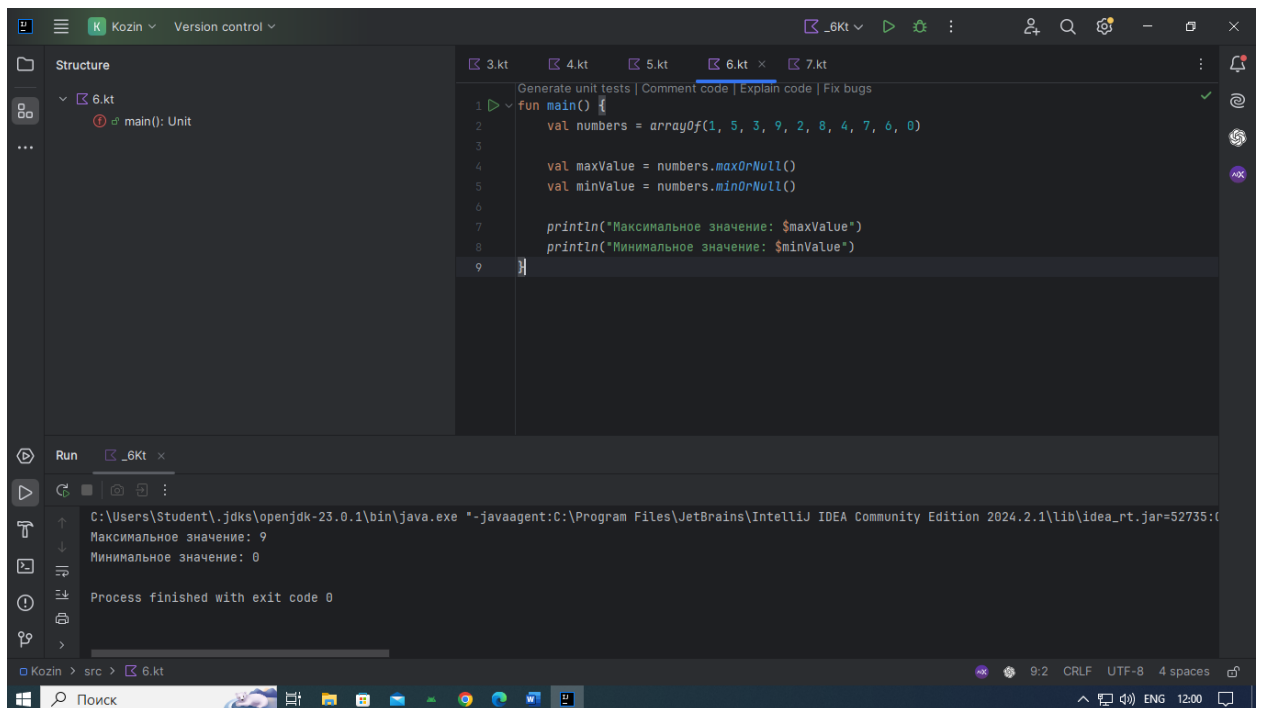
2. Сумма элементов массива: Напишите программу, которая находит сумму всех элементов массива чисел.

```
fun main() {  
    val numbers = intArrayOf(1, 2, 3, 4, 5)  
    val sum = numbers.sum()  
    println("Сумма элементов массива: $sum")  
}
```



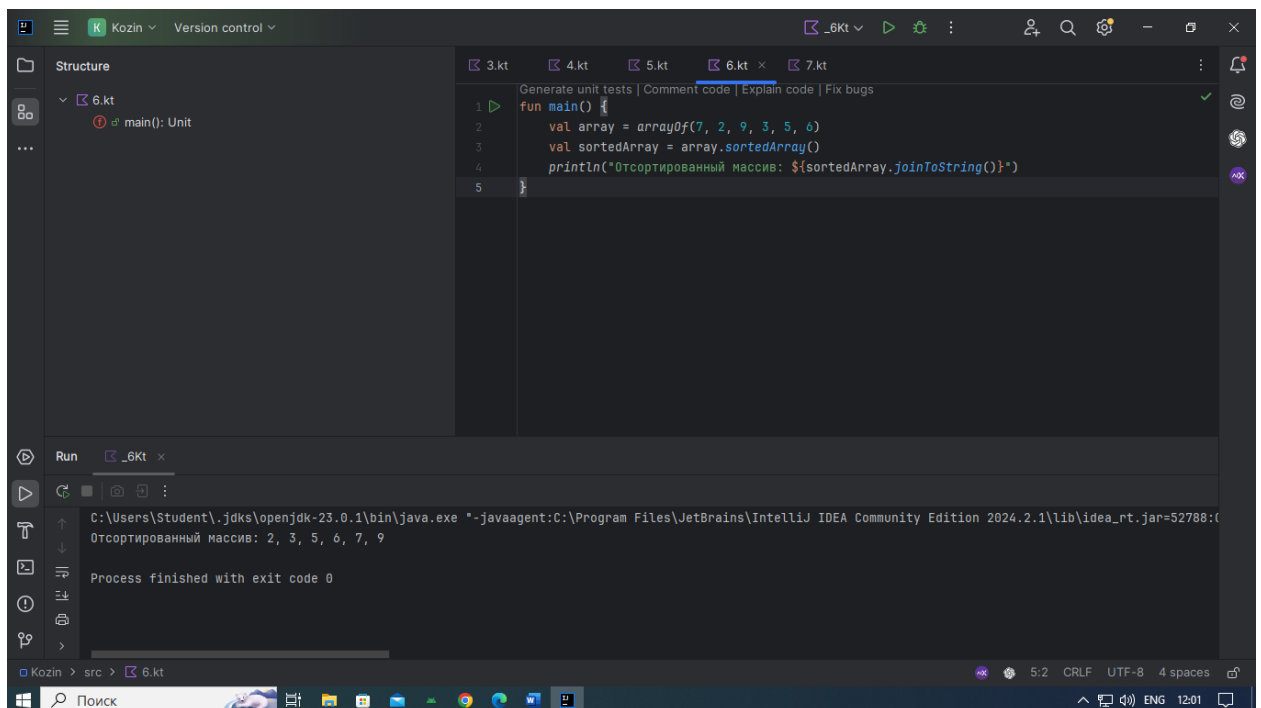
3. Максимальное и минимальное значение: Создайте массив из 10 чисел, найдите и выведите максимальное и минимальное значение.

```
fun main() {  
    val numbers = arrayOf(1, 5, 3, 9, 2, 8, 4, 7, 6, 0)  
  
    val maxValue = numbers.maxOrNull()  
    val minValue = numbers.minOrNull()  
  
    println("Максимальное значение: $maxValue")  
    println("Минимальное значение: $minValue")  
}
```



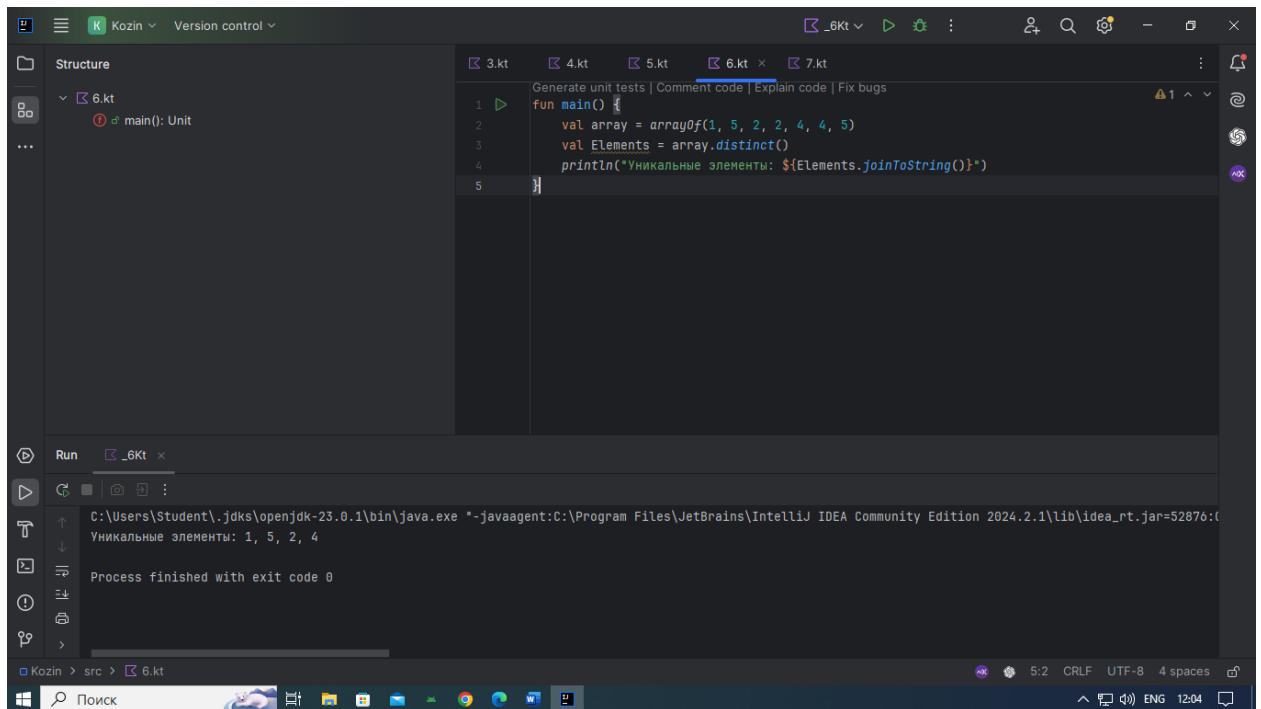
4. Сортировка массива: Реализуйте алгоритм сортировки для массива чисел и выведите отсортированный массив.

```
fun main() {  
    val array = arrayOf(7, 2, 9, 3, 5, 6)  
    val sortedArray = array.sortedArray()  
    println("Отсортированный массив: ${sortedArray.joinToString()}")  
}
```



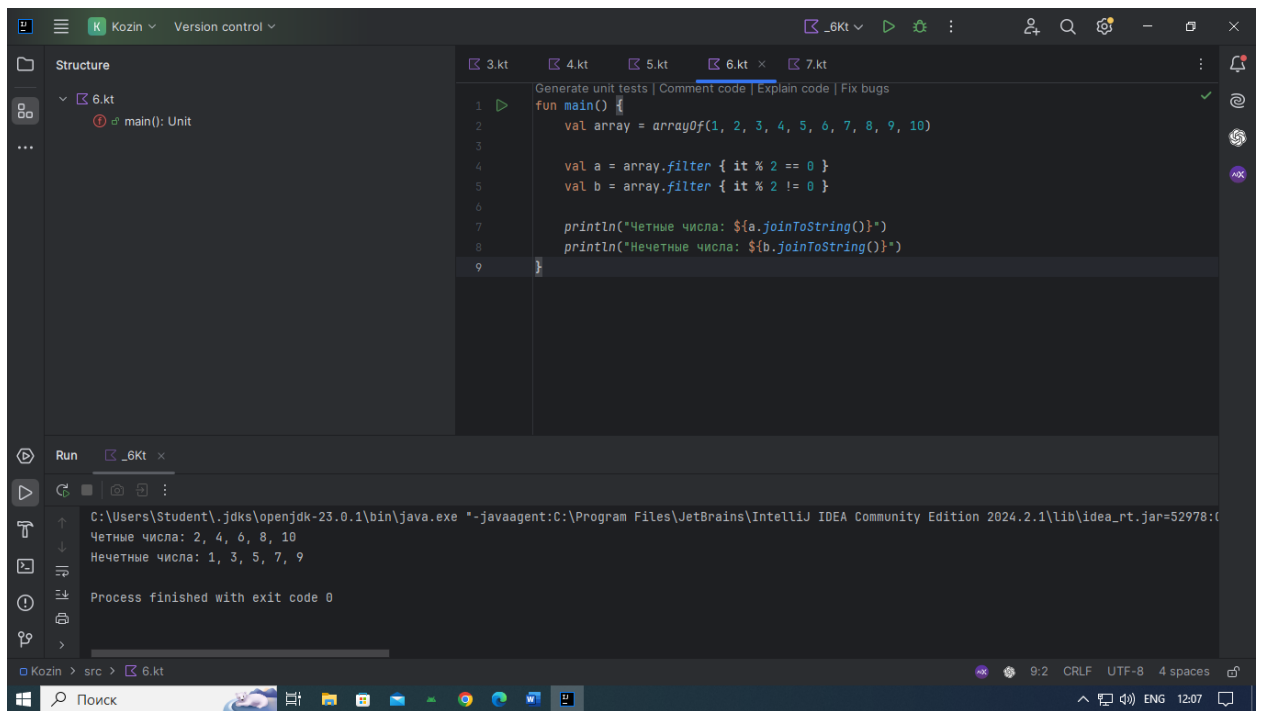
5. Уникальные элементы: Напишите программу, которая выводит уникальные элементы из массива.

```
fun main() {  
    val array = arrayOf(1, 5, 2, 2, 4, 4, 5)  
    val Elements = array.distinct()  
    println("Уникальные элементы: ${Elements.joinToString()}")  
}
```



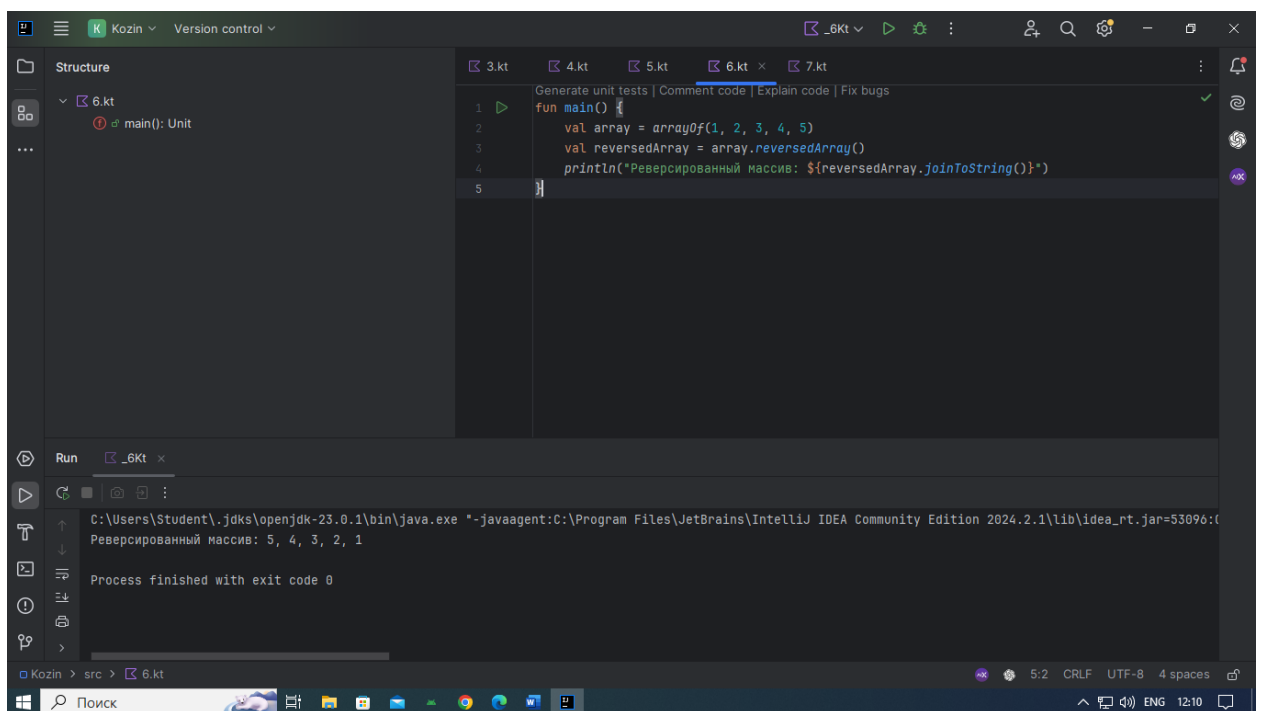
6. Четные и нечетные числа: Создайте массив и разделите его на четные и нечетные числа, сохранив их в разные массивы.

```
fun main() {  
    val array = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)  
  
    val a = array.filter { it % 2 == 0 }  
    val b = array.filter { it % 2 != 0 }  
  
    println("Четные числа: ${a.joinToString()}")  
    println("Нечетные числа: ${b.joinToString()}")  
}
```



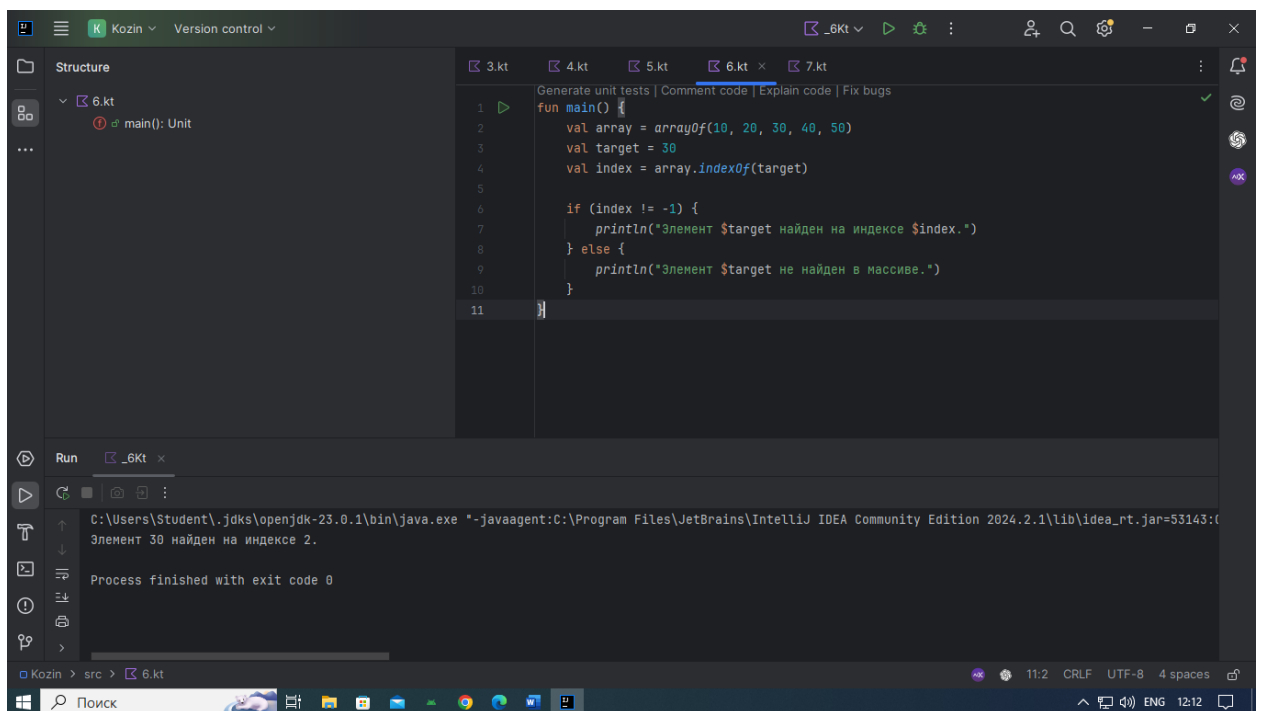
7. Реверс массива: Напишите программу, которая реверсирует массив чисел.

```
fun main() {  
    val array = arrayOf(1, 2, 3, 4, 5)  
    val reversedArray = array.reversedArray()  
    println("Реверсированный массив: ${reversedArray.joinToString()}")  
}
```



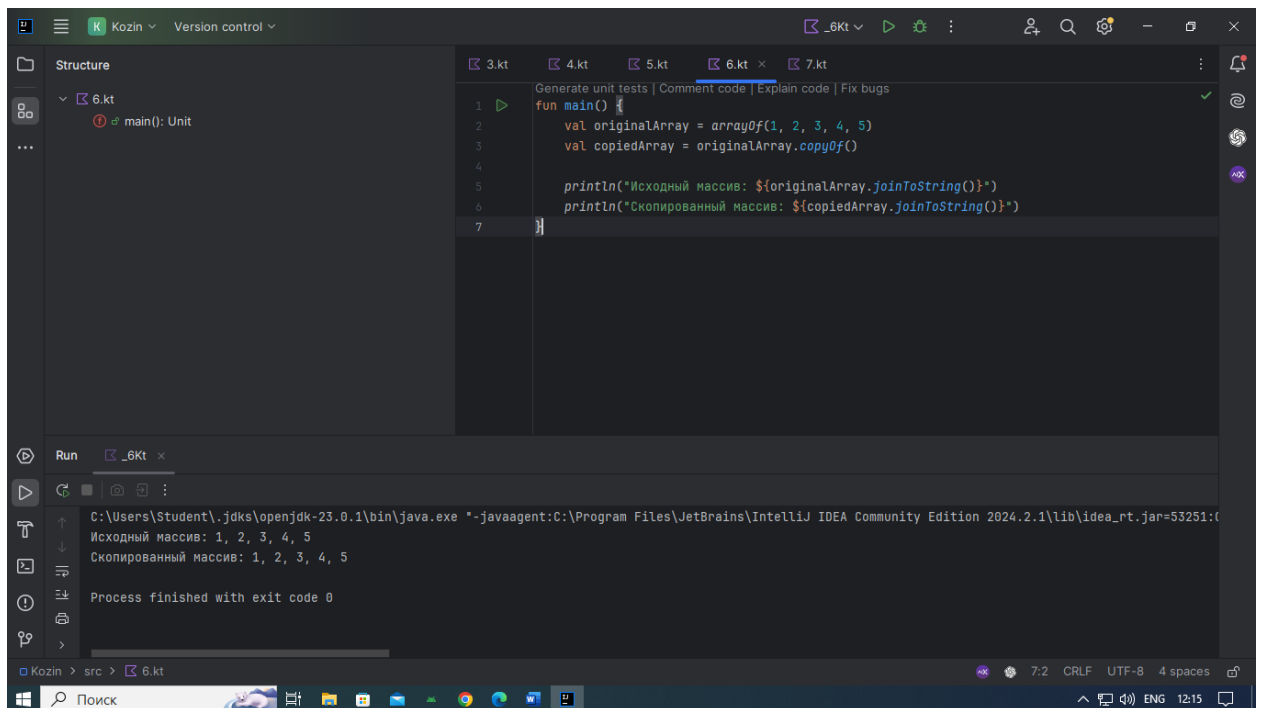
## 8. Поиск элемента: Реализуйте поиск элемента в массиве и выводите его индекс

```
fun main() {  
    val array = arrayOf(10, 20, 30, 40, 50)  
    val target = 30  
    val index = array.indexOf(target)  
  
    if (index != -1) {  
        println("Элемент $target найден на индексе $index.")  
    } else {  
        println("Элемент $target не найден в массиве.")  
    }  
}
```



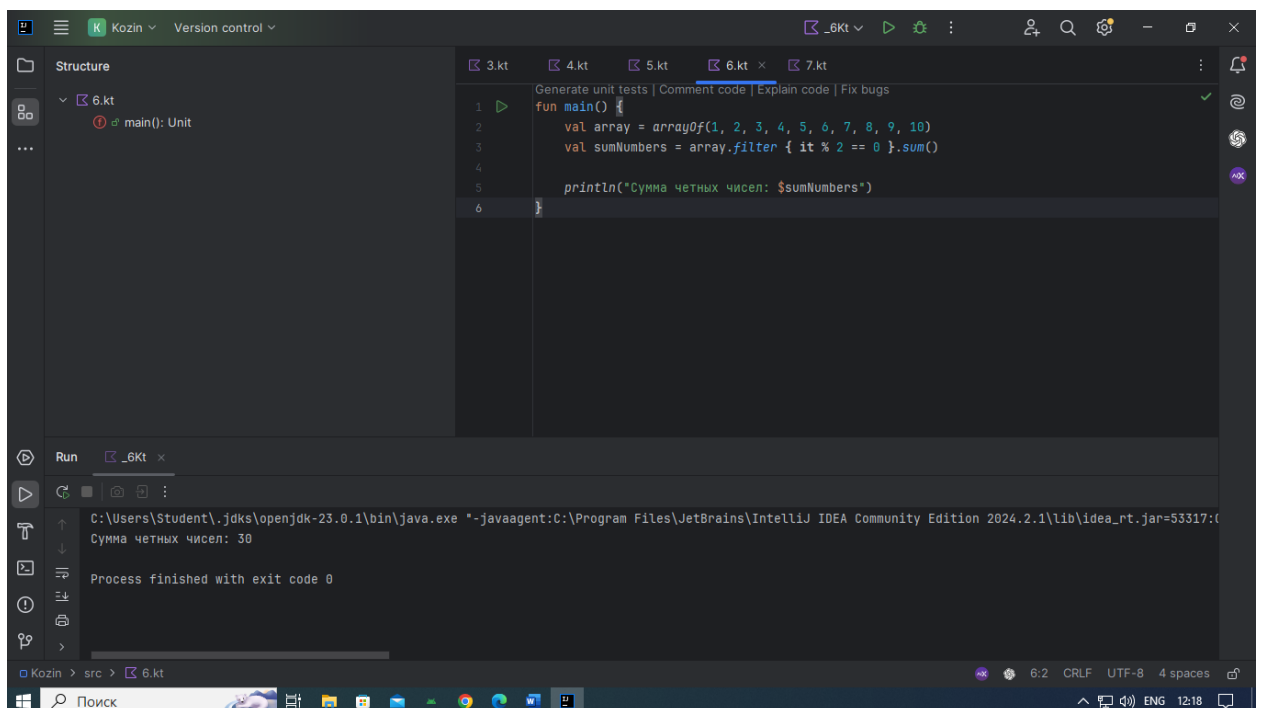
## 9. Копирование массива: Создайте новый массив, скопировав в него элементы из другого массива.

```
fun main() {  
    val originalArray = arrayOf(1, 2, 3, 4, 5)  
    val copiedArray = originalArray.copyOf()  
  
    println("Исходный массив: ${originalArray.joinToString()}")  
    println("Скопированный массив: ${copiedArray.joinToString()}")  
}
```



10. Сумма четных чисел: Напишите программу, которая находит сумму всех четных чисел в массиве

```
fun main() {  
    val array = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)  
    val sumNumbers = array.filter { it % 2 == 0 }.sum()  
  
    println("Сумма четных чисел: $sumNumbers")  
}
```



11. Пересечение массивов: Напишите программу, которая находит пересечение двух массивов и выводит результат

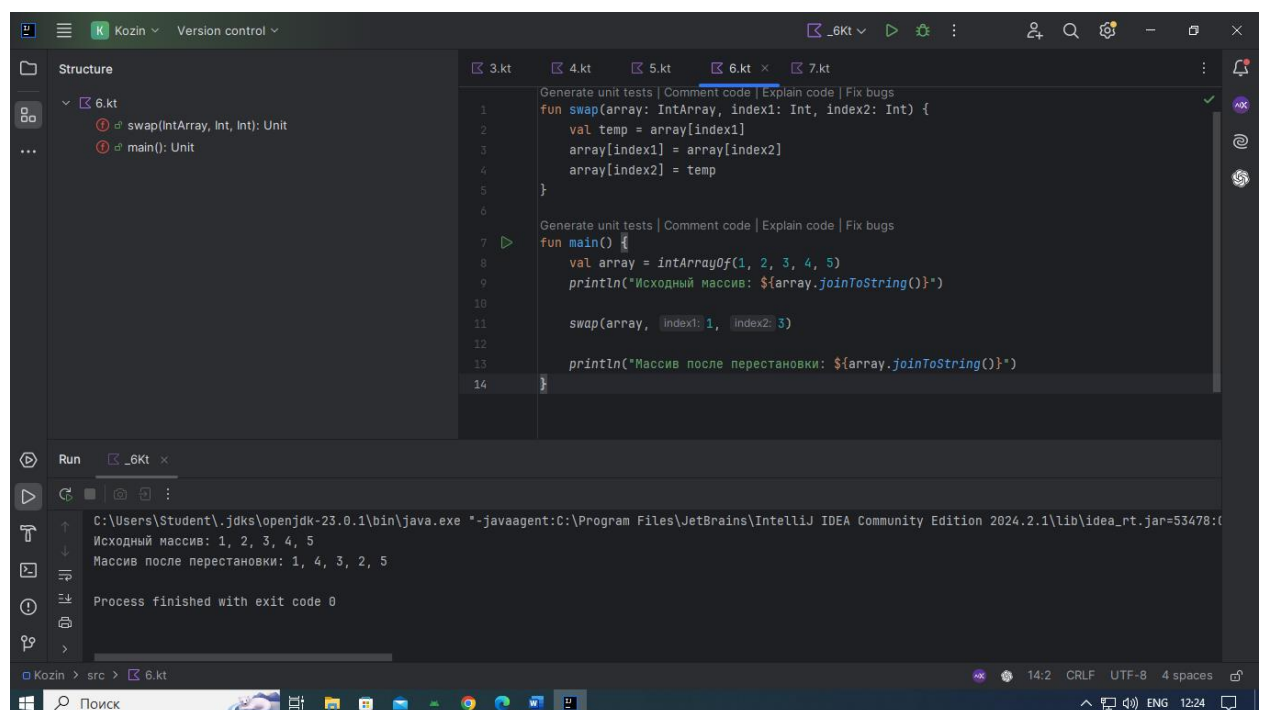
12. Перестановка элементов: Реализуйте функцию, которая меняет местами два элемента в массиве.

```
val temp = array[index1]
array[index1] = array[index2]
array[index2] = temp
}

fun main() {
    val array = intArrayOf(1, 2, 3, 4, 5)
    println("Исходный массив: ${array.joinToString()}")

    swap(array, 1, 3)

    println("Массив после перестановки: ${array.joinToString()}")
}
```



13. Заполнение случайными числами: Создайте массив из 20 случайных чисел от 1 до 100 и выведите его на экран

```
import kotlin.random.Random

fun main() {
```

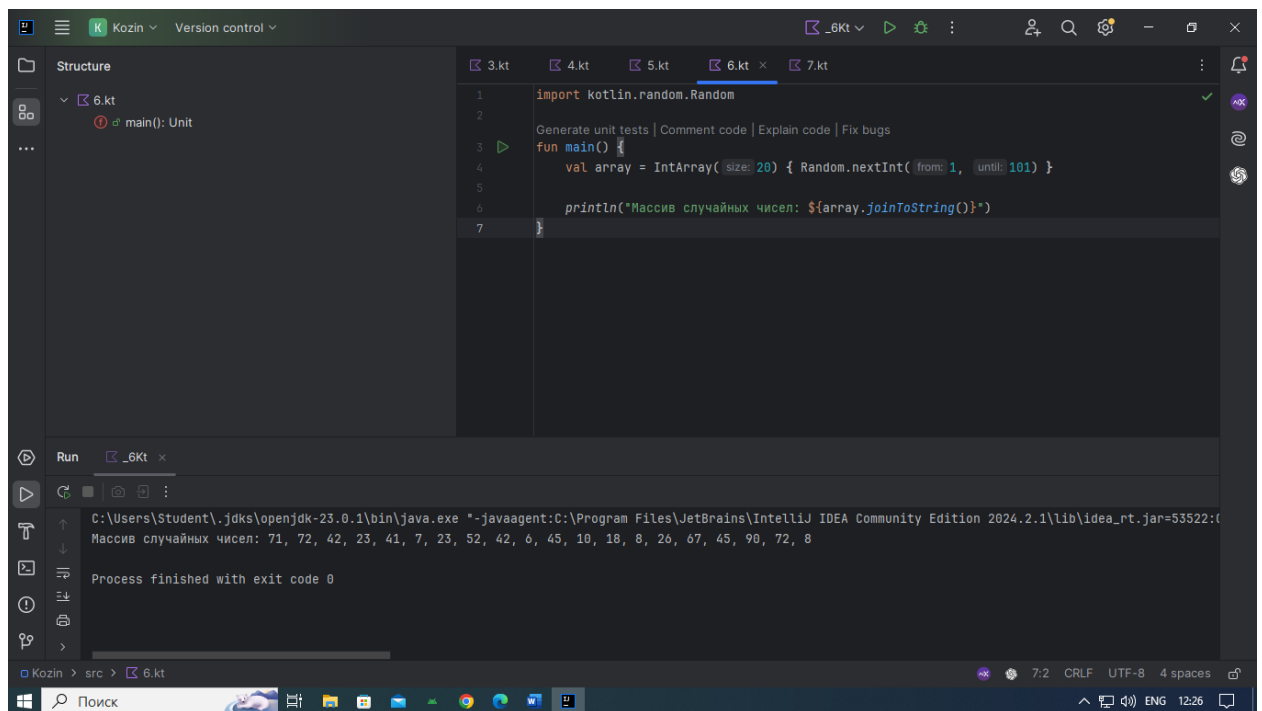


```

val array = IntArray(20) { Random.nextInt(1, 101) }

println("Массив случайных чисел: ${array.joinToString()}")
}

```



14. Числа Прокопенко: Напишите программу, которая выводит все числа в массиве, делящиеся на 3

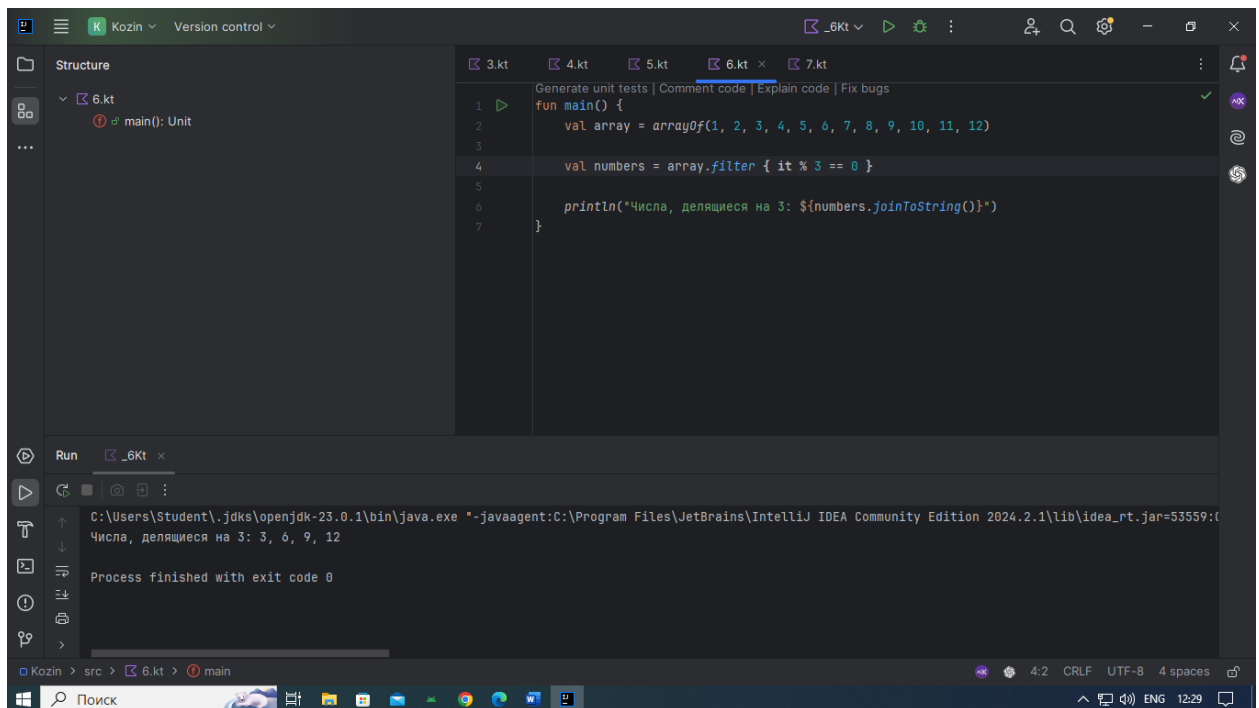
```

fun main() {
    val array = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)

    val numbers = array.filter { it % 3 == 0 }

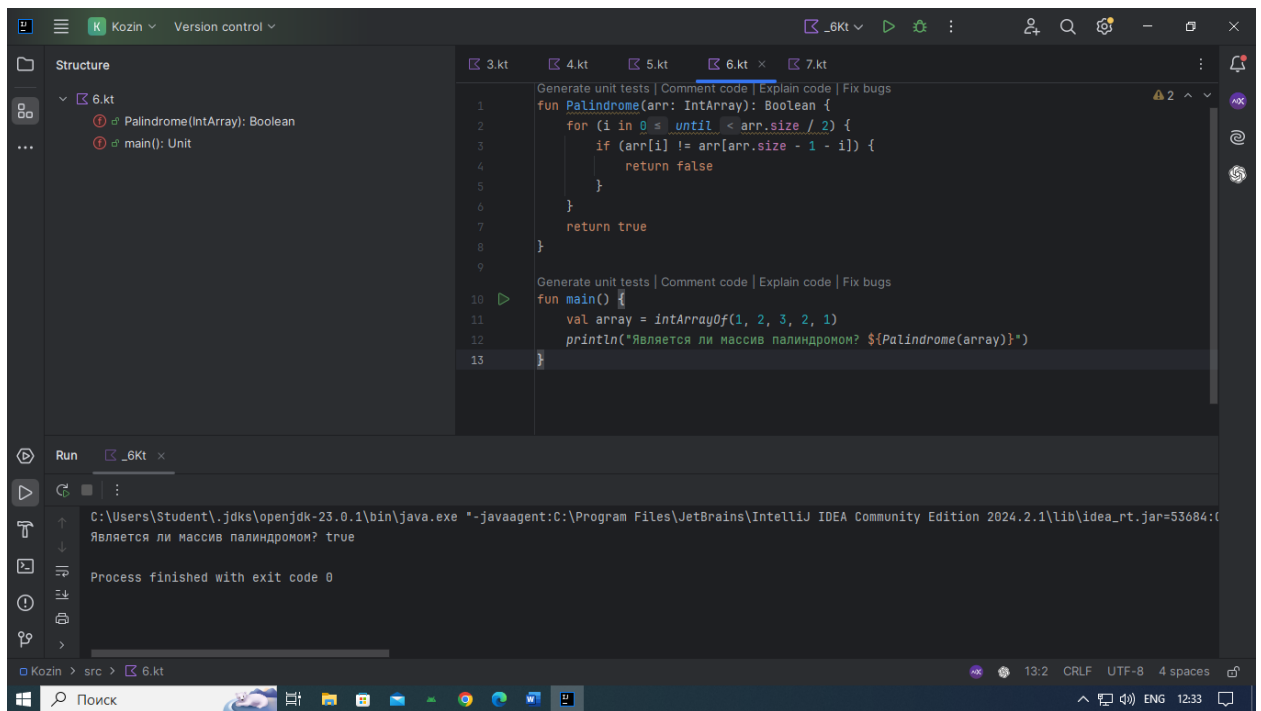
    println("Числа, делящиеся на 3: ${numbers.joinToString()}")
}

```



15. Проверка на палиндром: Напишите программу, которая проверяет, является ли массив палиндромом.

```
fun Palindrome(arr: IntArray): Boolean {  
    for (i in 0 until arr.size / 2) {  
        if (arr[i] != arr[arr.size - 1 - i]) {  
            return false  
        }  
    }  
    return true  
}  
  
fun main() {  
    val array = intArrayOf(1, 2, 3, 2, 1)  
    println("Является ли массив палиндромом? ${Palindrome(array)}")  
}
```

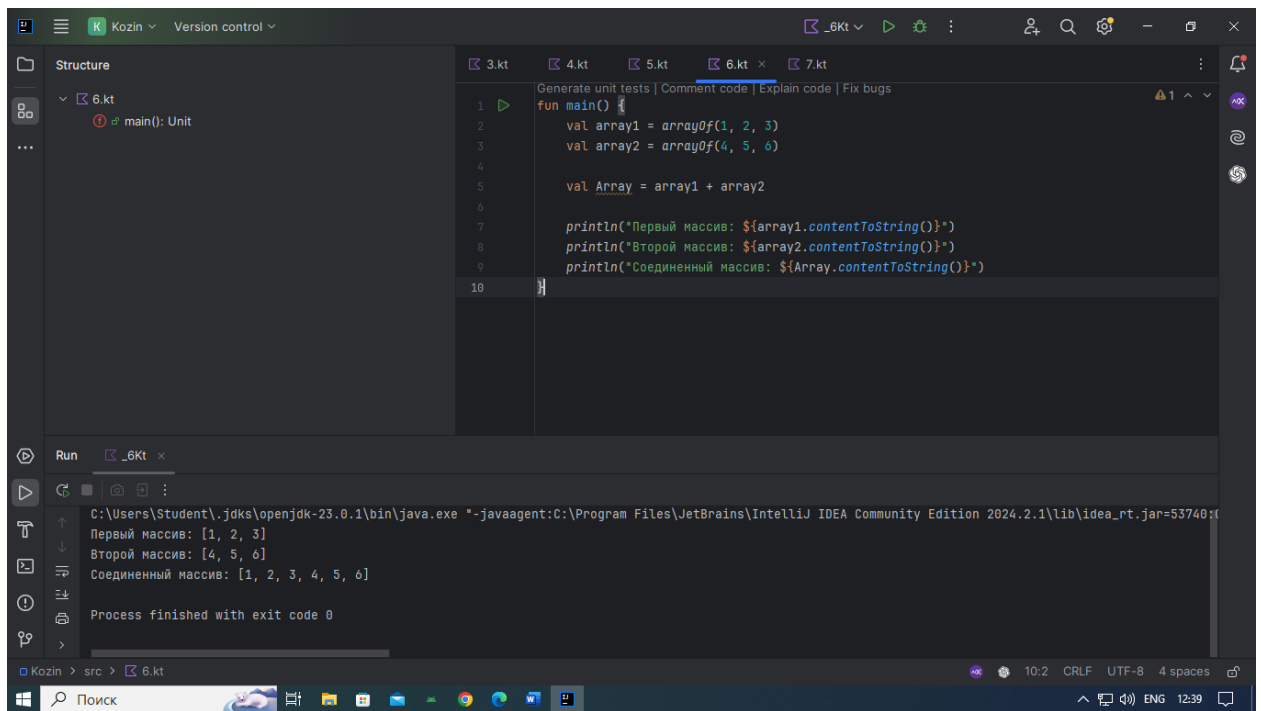


16. Конкатенация двух массивов: Создайте два массива и соедините их в один.

```
fun main() {
    val array1 = arrayOf(1, 2, 3)
    val array2 = arrayOf(4, 5, 6)

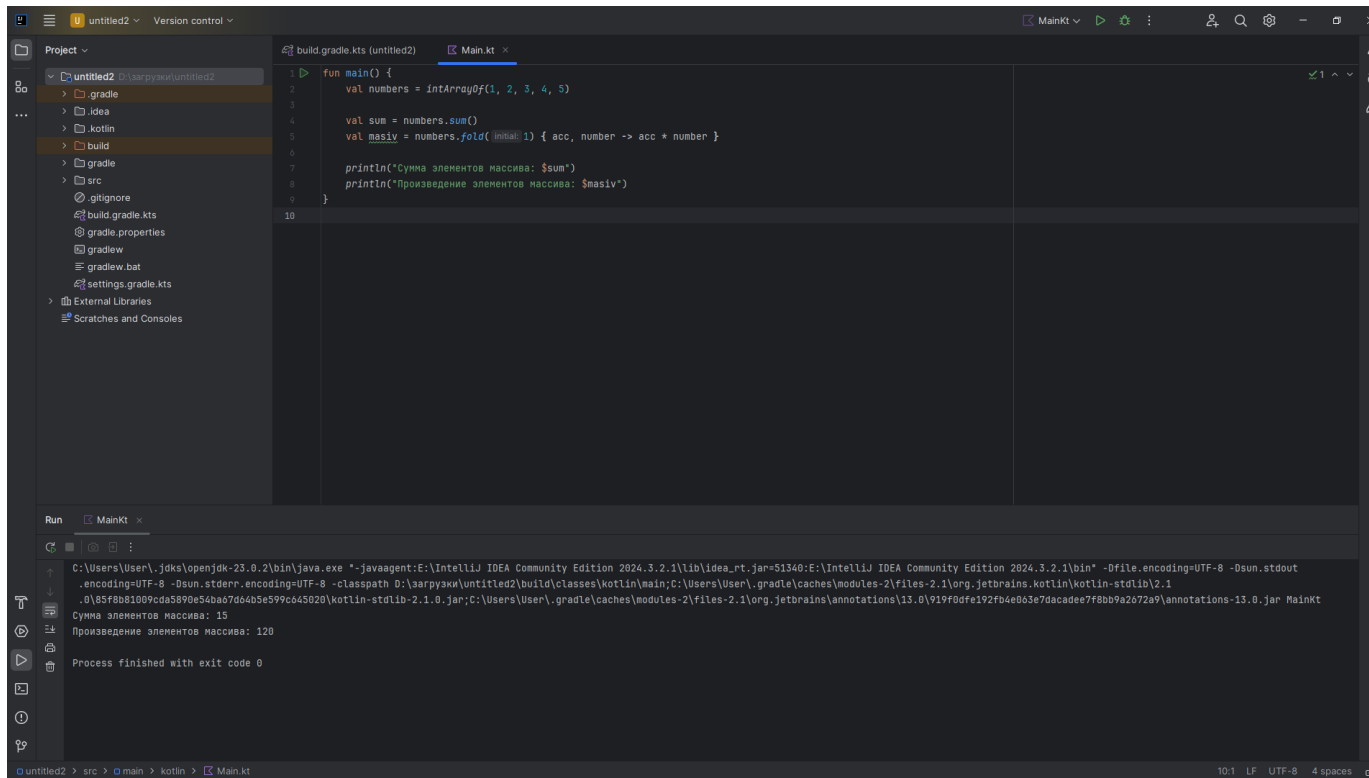
    val Array = array1 + array2

    println("Первый массив: ${array1.contentToString()}")
    println("Второй массив: ${array2.contentToString()}")
    println("Соединенный массив: ${Array.contentToString()}")
}
```



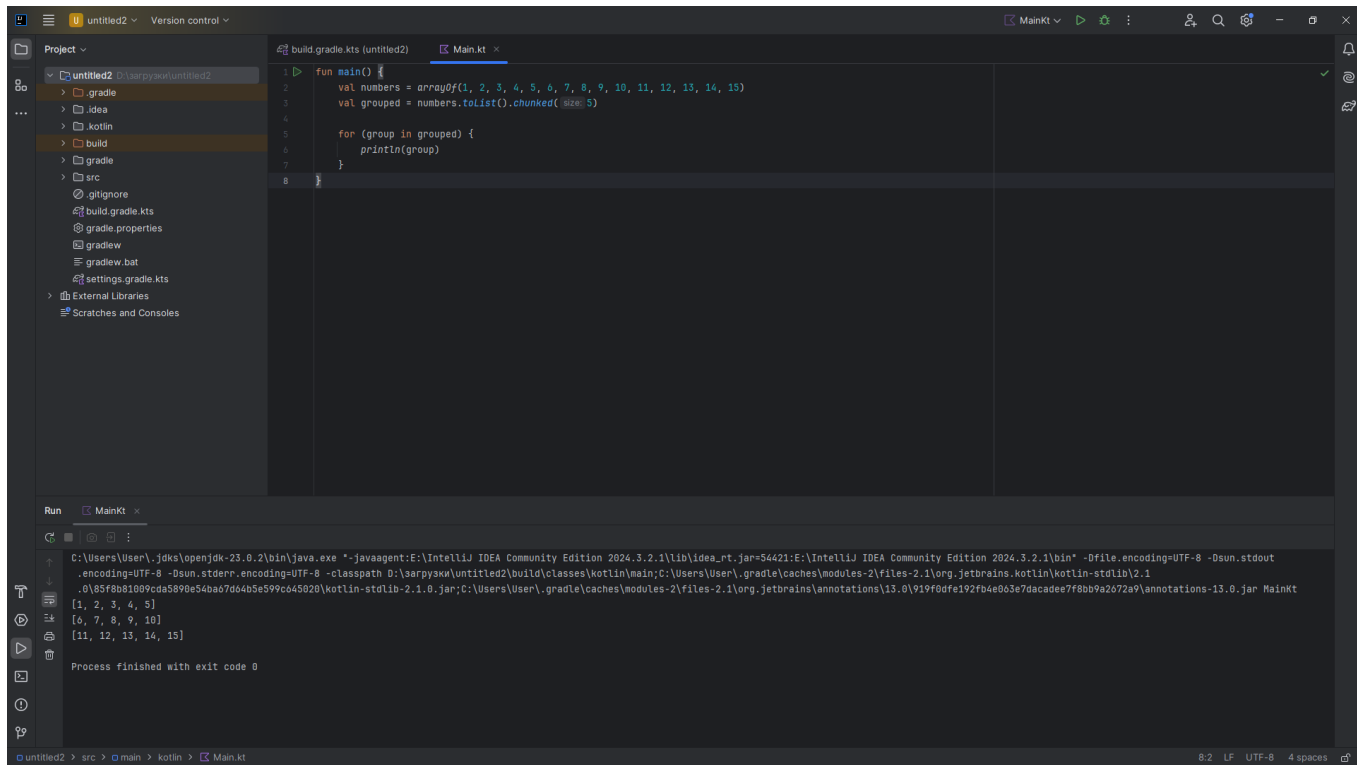
17. Сумма и произведение: Напишите программу, которая находит и выводит сумму и произведение всех элементов массива.

```
fun main() {  
    val numbers = intArrayOf(1, 2, 3, 4, 5)  
  
    val sum = numbers.sum()  
    val masiv = numbers.fold(1) { acc, number -> acc * number }  
  
    println("Сумма элементов массива: $sum")  
    println("Произведение элементов массива: $masiv")  
}
```



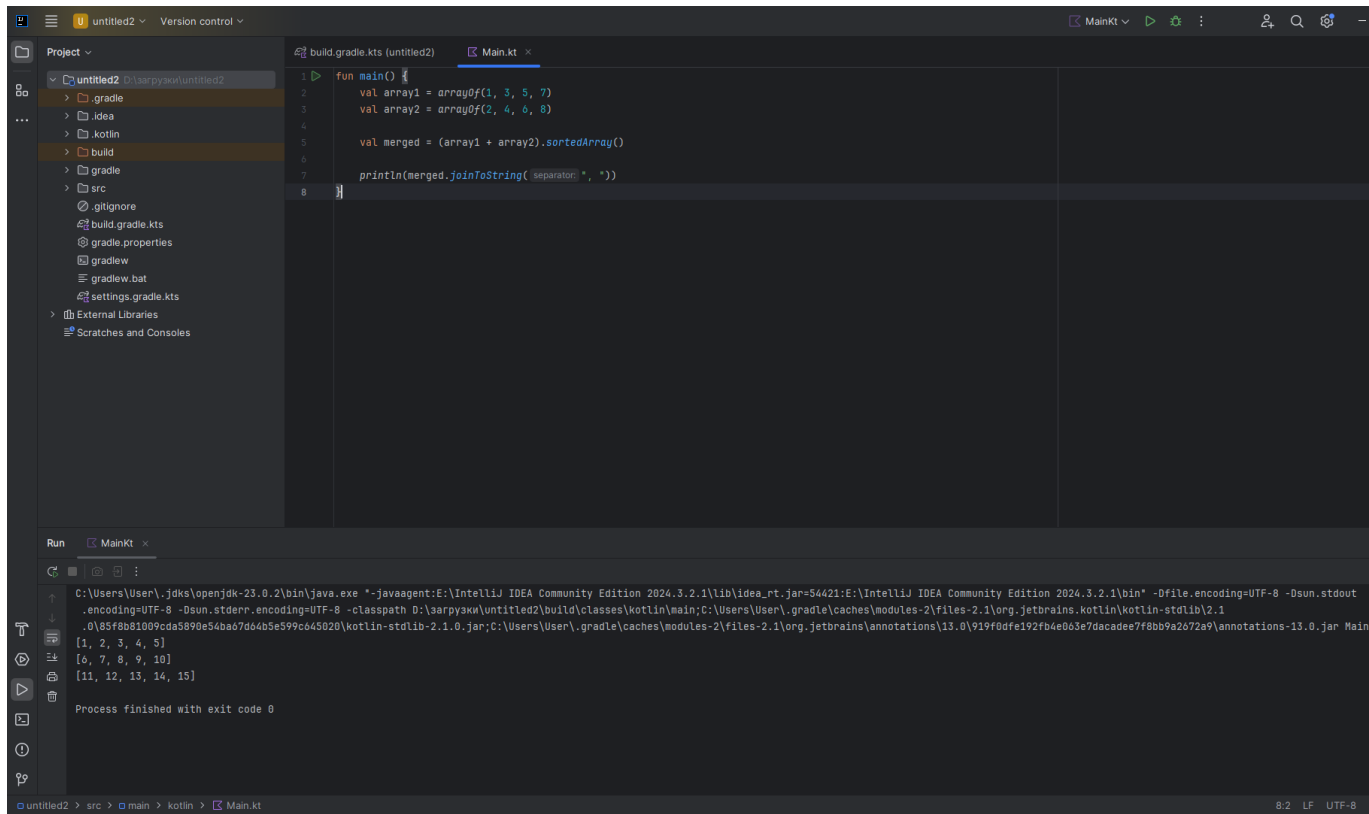
18.Группировка чисел: Разделите массив на группы по 5 элементов и выведите их.

```
fun main() {  
    val numbers = arrayOf(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15)  
    val grouped = numbers.toList().chunked(5)  
  
    for (group in grouped) {  
        println(group)  
    }  
}
```



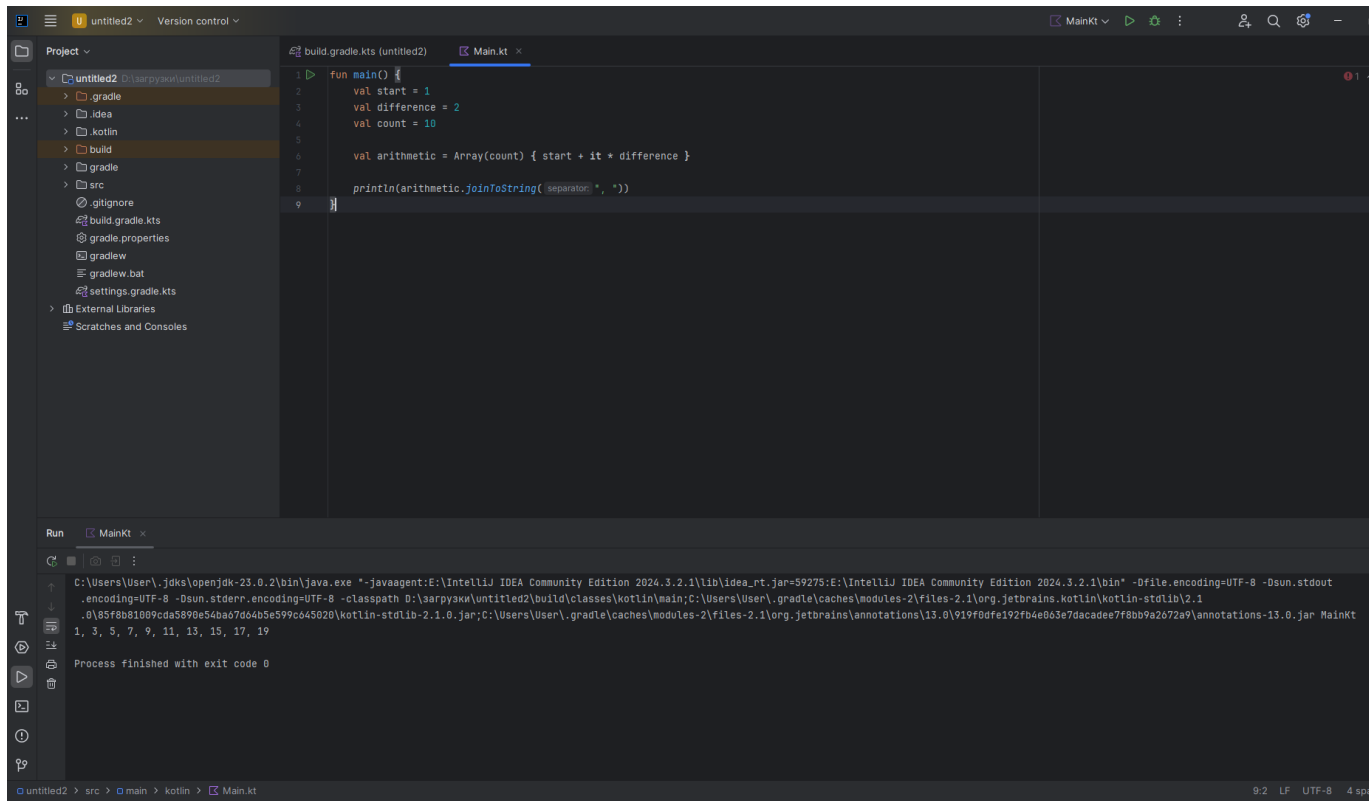
19.Слияние двух массивов: Напишите программу, которая сливает два отсортированных массива в один отсортированный массив

```
fun main() {  
    val array1 = arrayOf(1, 3, 5, 7)  
    val array2 = arrayOf(2, 4, 6, 8)  
  
    val merged = (array1 + array2).sortedArray()  
  
    println(merged.joinToString(", "))  
}
```



20. Числовая последовательность: Создайте массив целых чисел, представляющий арифметическую прогрессию, и выведите его.

```
fun main() {  
    val start = 1  
    val difference = 2  
    val count = 10  
  
    val arithmetic = Array(count) { start + it * difference }  
  
    println(arithmetic.joinToString(", "))  
}
```



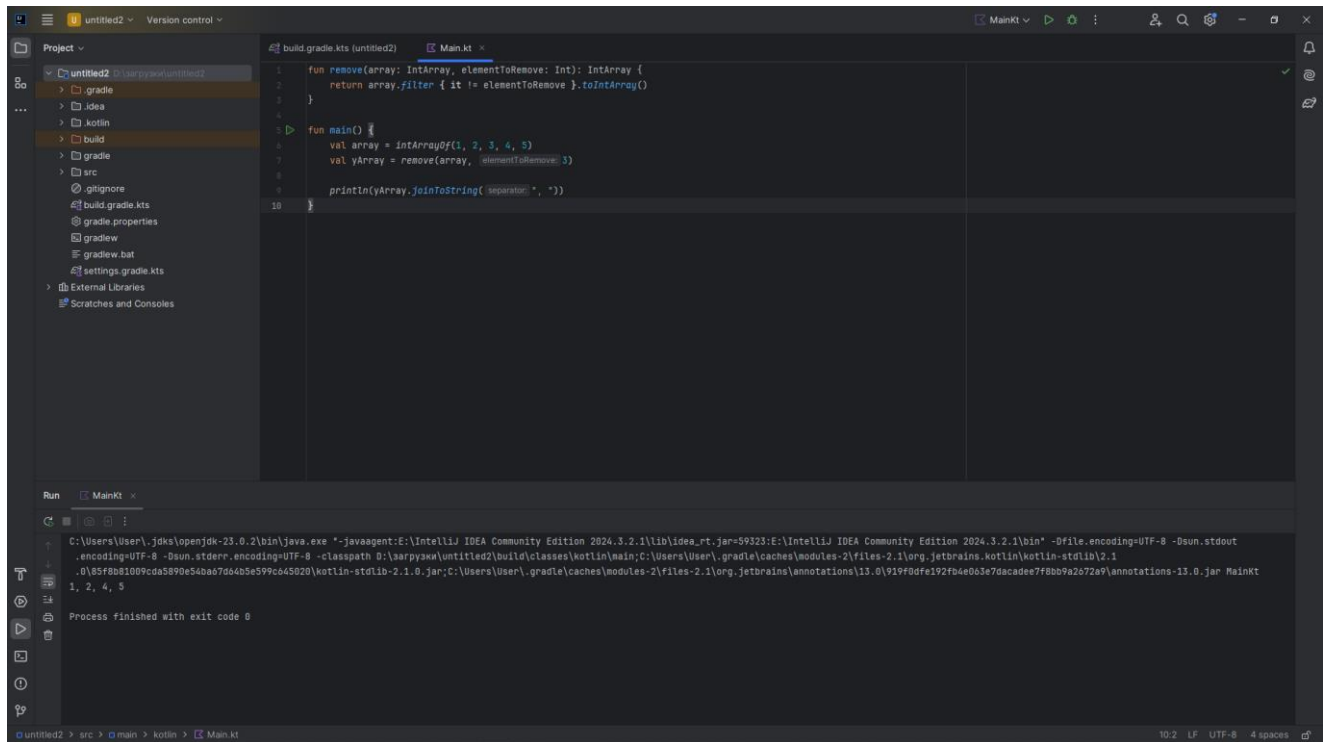
21.Удаление элемента: Реализуйте функцию, которая удаляет заданный элемент из массива.

```
fun remove(array: IntArray, elementToRemove: Int): IntArray {
    return array.filter { it != elementToRemove }.toIntArray()
}

fun main() {
    val array = intArrayOf(1, 2, 3, 4, 5)
    val yArray = remove(array, 3)

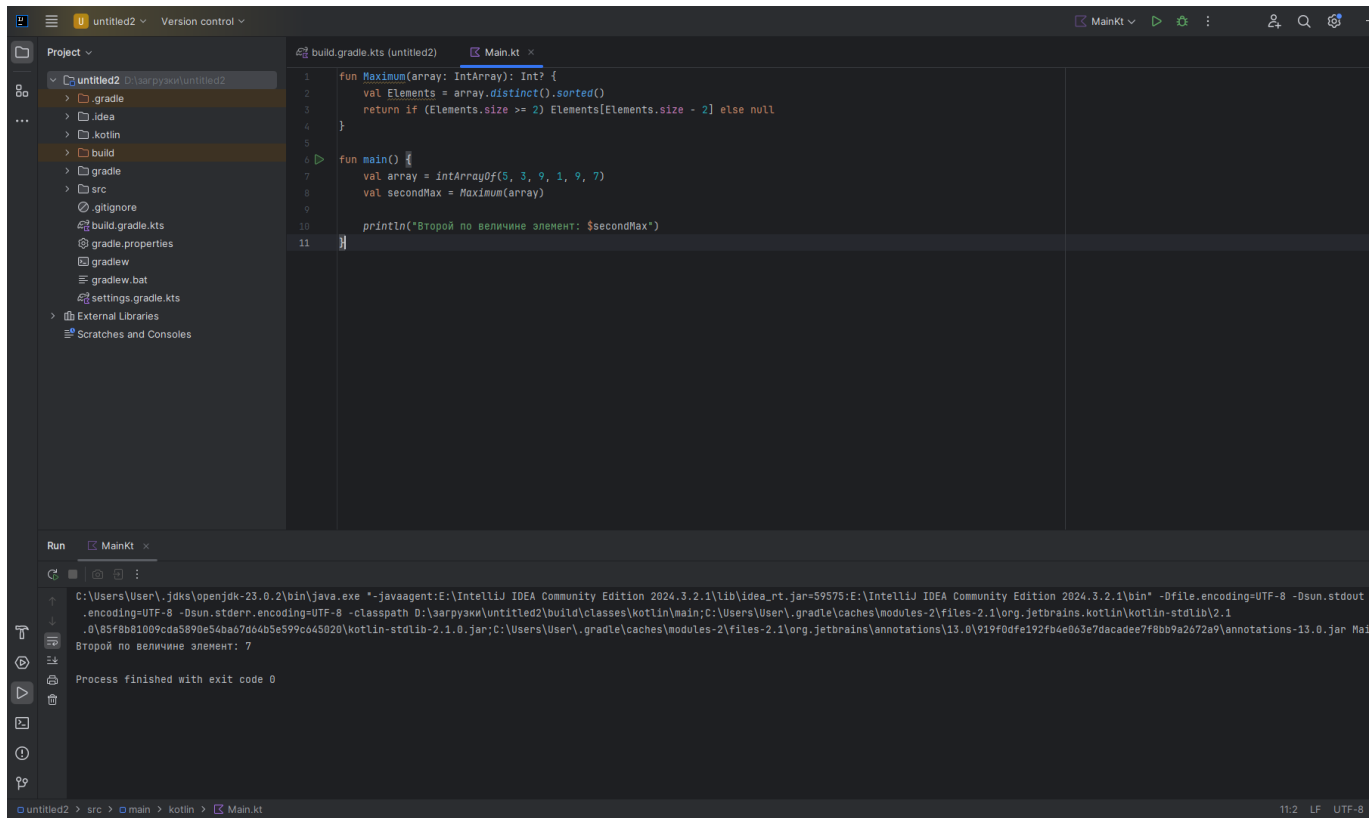
    println(yArray.joinToString(", "))
}
```





22.Поиск второго максимального: Напишите программу, которая находит второй по величине элемент в массиве.

```
fun Maximum(array: IntArray): Int? {  
    val Elements = array.distinct().sorted()  
    return if (Elements.size >= 2) Elements[Elements.size - 2] else null  
}  
  
fun main() {  
    val array = intArrayOf(5, 3, 9, 1, 9, 7)  
    val secondMax = Maximum(array)  
  
    println("Второй по величине элемент: $secondMax")  
}
```



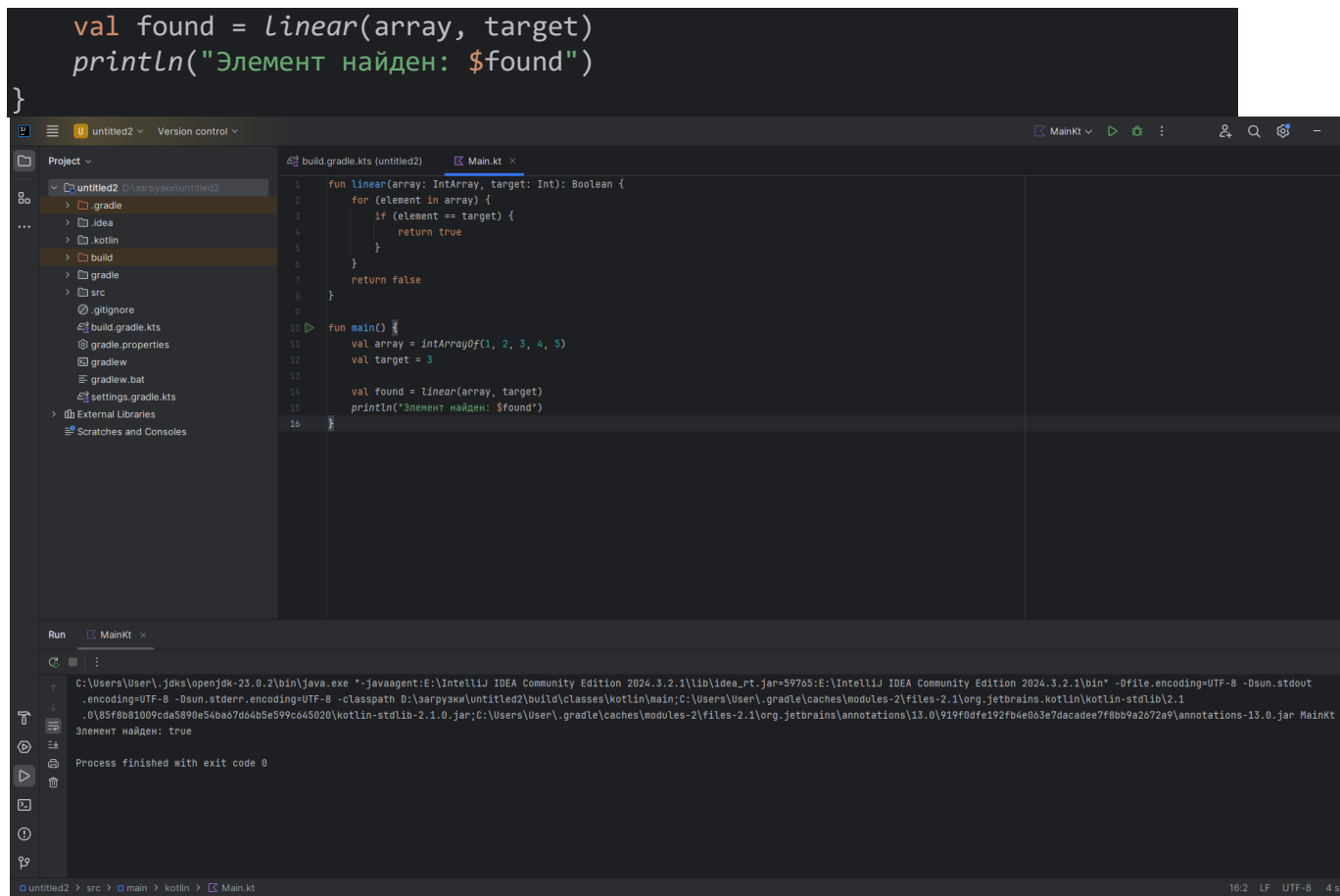
23.Объединение массивов: Напишите функцию, которая объединяет несколько массивов чисел и выводит результирующий массив

24.Транспонирование матрицы: Создайте матрицу (двумерный массив) и напишите программу, которая транспонирует её

25.Линейный поиск: Реализуйте линейный поиск элемента в массиве с возвратомBool-значения (найден или нет).

```
fun linear(array: IntArray, target: Int): Boolean {
    for (element in array) {
        if (element == target) {
            return true
        }
    }
    return false
}

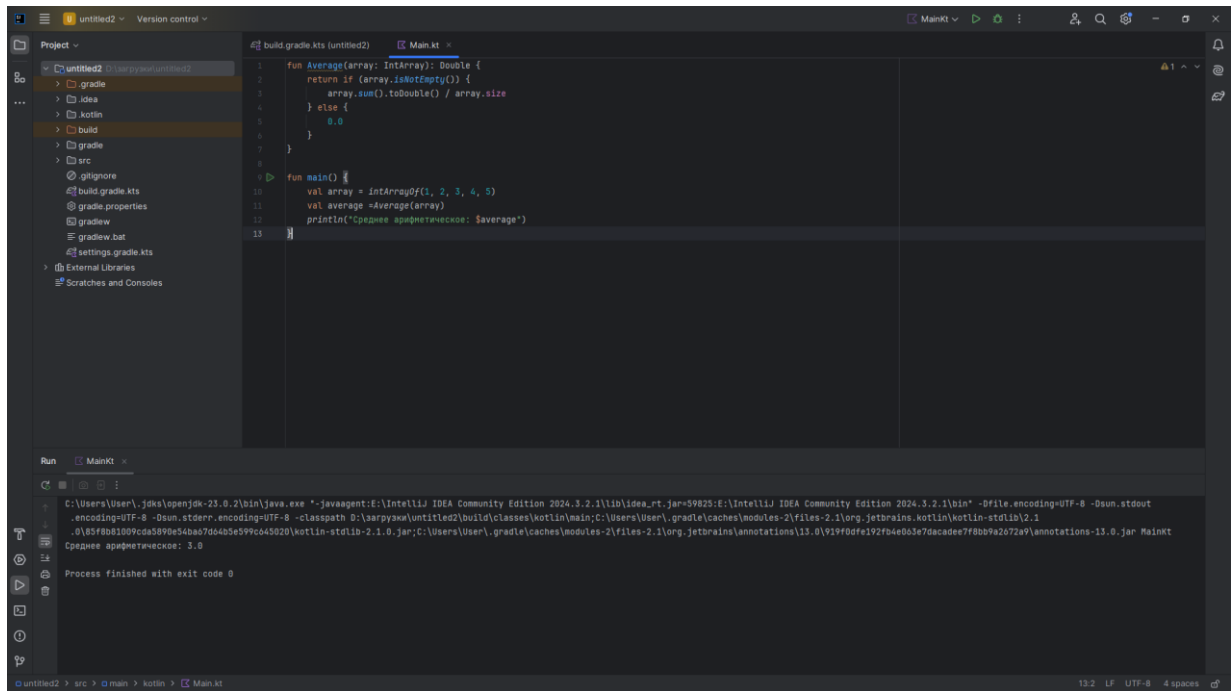
fun main() {
    val array = intArrayOf(1, 2, 3, 4, 5)
    val target = 3
}
```



26.Среднее арифметическое: Напишите программу, которая находит среднее арифметическое всех чисел в массиве.

```
fun Average(array: IntArray): Double {
    return if (array.isNotEmpty()) {
        array.sum().toDouble() / array.size
    } else {
        0.0
    }
}

fun main() {
    val array = intArrayOf(1, 2, 3, 4, 5)
    val average = Average(array)
    println("Среднее арифметическое: $average")
}
```



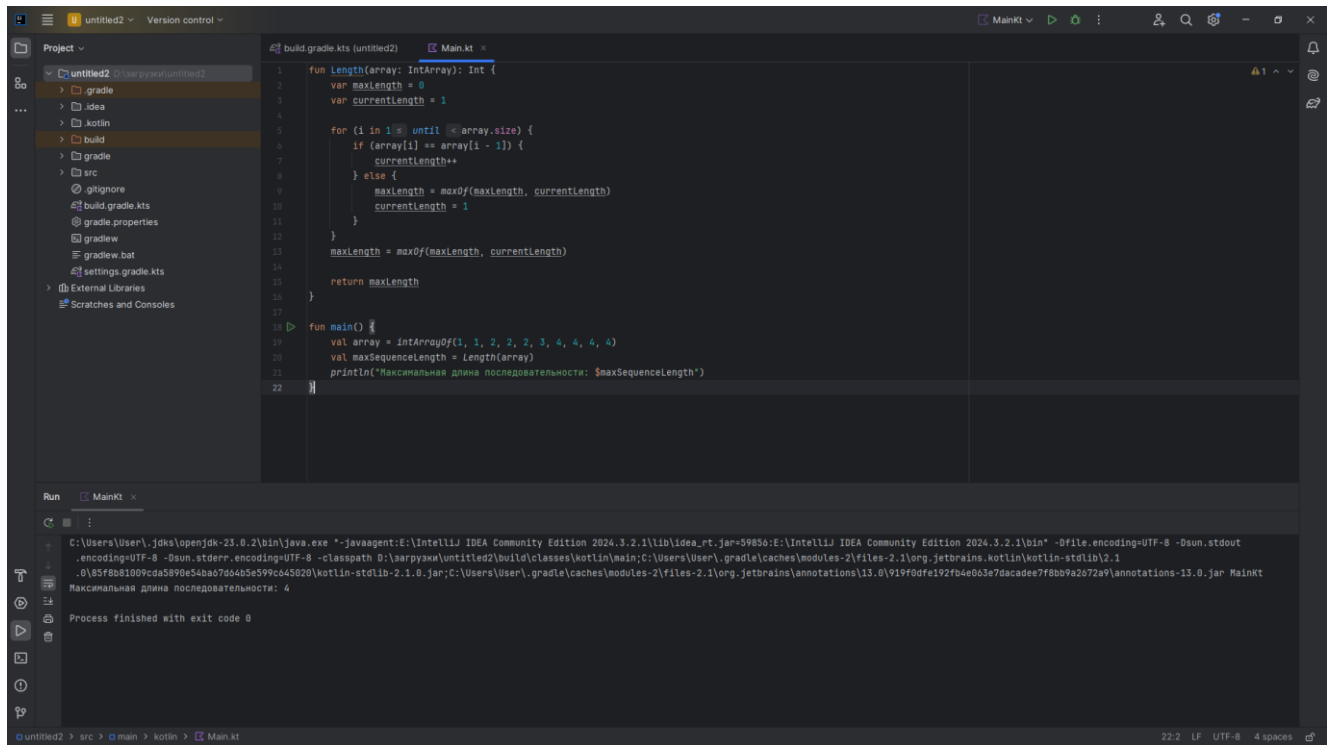
27.Максимальная последовательность: Найдите максимальную последовательность одинаковых элементов в массиве

```
fun Length(array: IntArray): Int {
    var maxLength = 0
    var currentLength = 1

    for (i in 1 until array.size) {
        if (array[i] == array[i - 1]) {
            currentLength++
        } else {
            maxLength = maxOf(maxLength, currentLength)
            currentLength = 1
        }
    }
    maxLength = maxOf(maxLength, currentLength)

    return maxLength
}

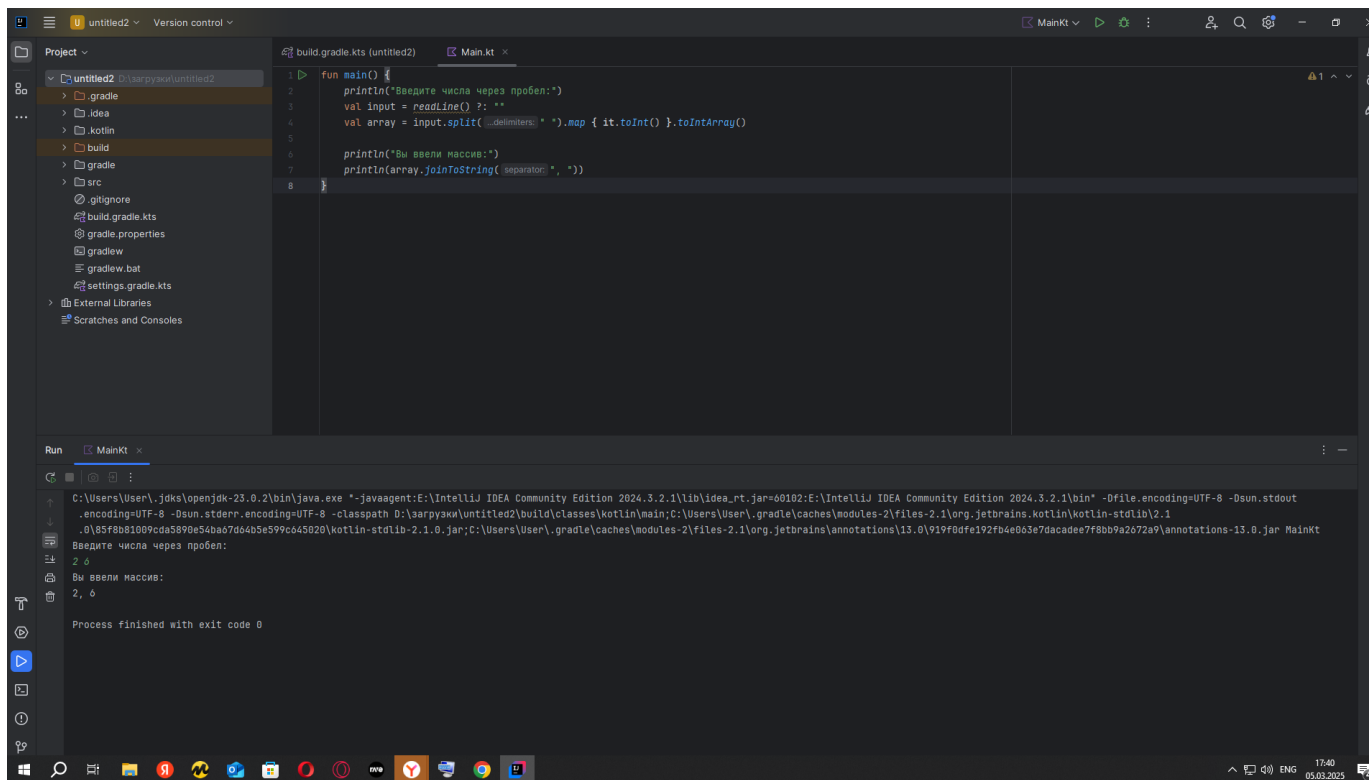
fun main() {
    val array = intArrayOf(1, 1, 2, 2, 2, 3, 4, 4, 4, 4)
    val maxSequenceLength = Length(array)
    println("Максимальная длина последовательности: $maxSequenceLength")
}
```



28. Ввод и вывод массива: Напишите программу, которая запрашивает у пользователя ввод массива чисел и затем выводит его

```
fun main() {
    println("Введите числа через пробел:")
    val input = readLine() ?: ""
    val array = input.split(" ").map { it.toInt() }.toIntArray()

    println("Вы ввели массив:")
    println(array.joinToString(", "))
}
```

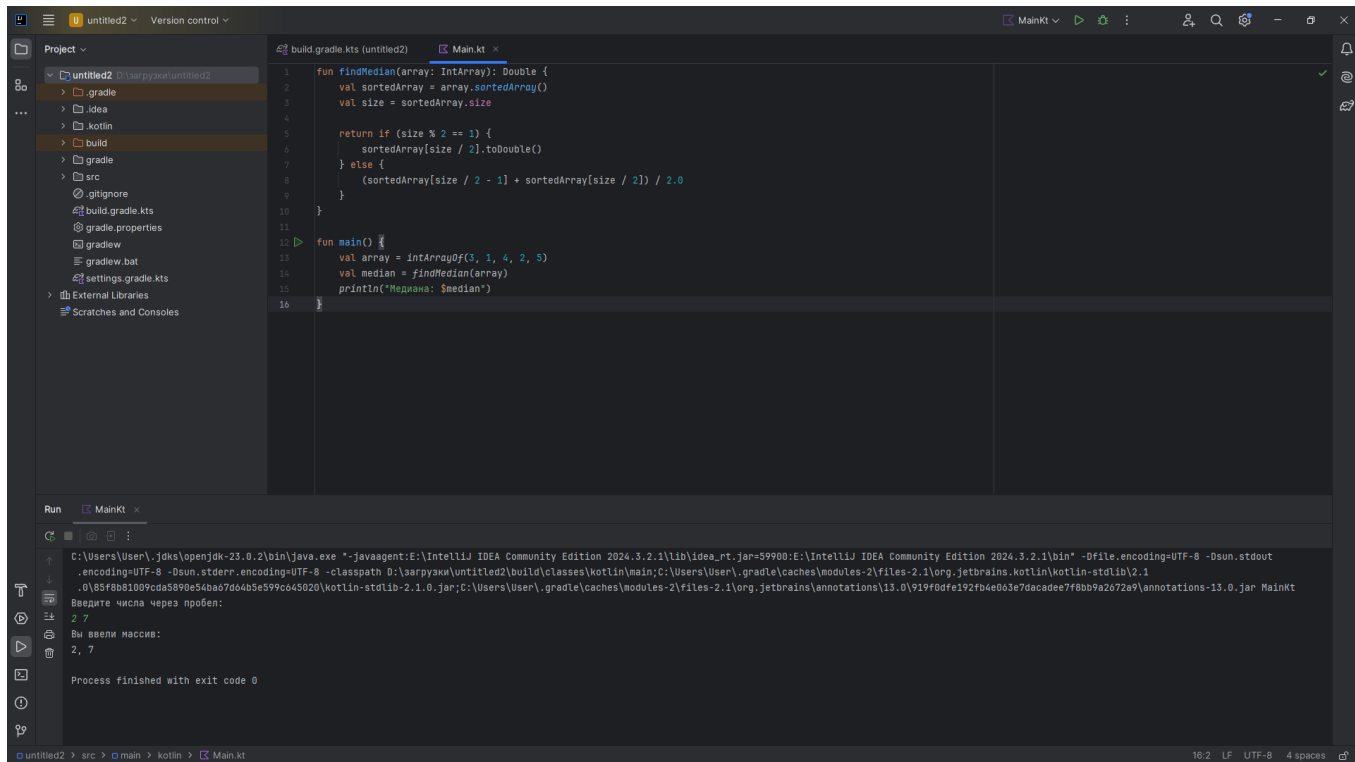


29.Нахождение медианы: Напишите программу, которая находит медиану в массиве.

```
fun findMedian(array: IntArray): Double {
    val sortedArray = array.sortedArray()
    val size = sortedArray.size

    return if (size % 2 == 1) {
        sortedArray[size / 2].toDouble()
    } else {
        (sortedArray[size / 2 - 1] + sortedArray[size / 2]) / 2.0
    }
}

fun main() {
    val array = intArrayOf(3, 1, 4, 2, 5)
    val median = findMedian(array)
    println("Медиана: $median")
}
```



30.Распределение по группам: Создайте массив из 100 целых чисел и разделите их на 10 групп по 10 элементов, затем выведите результаты

```
fun main() {
    val numbers = IntArray(100) { it + 1 }

    for (i in numbers.indices step 10) {
        val group = numbers.sliceArray(i until i + 10)
        println(group.joinToString(", "))
    }
}
```

