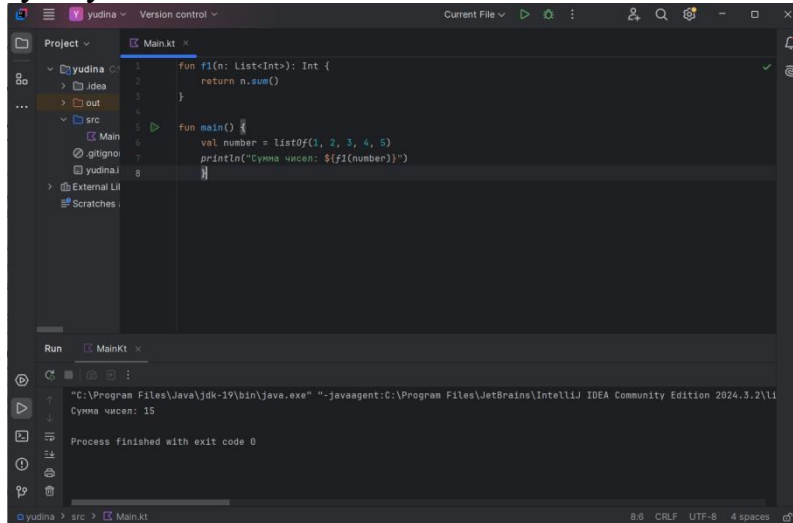


## Практическая работа 8

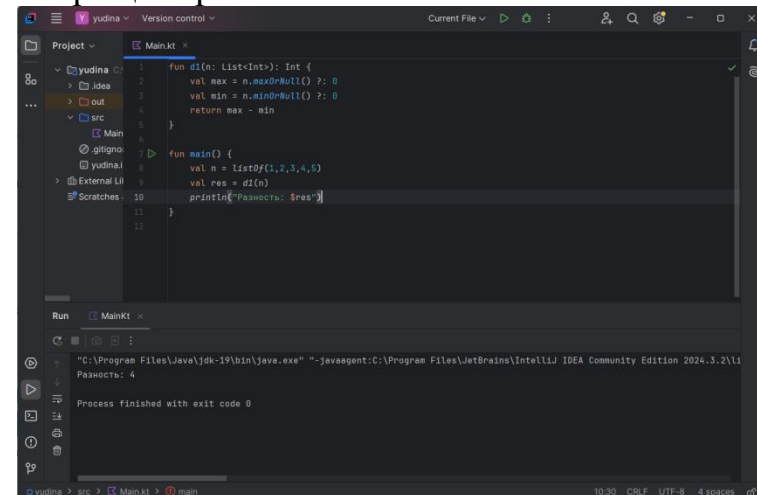
### Функциональное программирование

1. Напишите функцию, которая принимает список на вход, и возвращает сумму всех элементов этого списка.



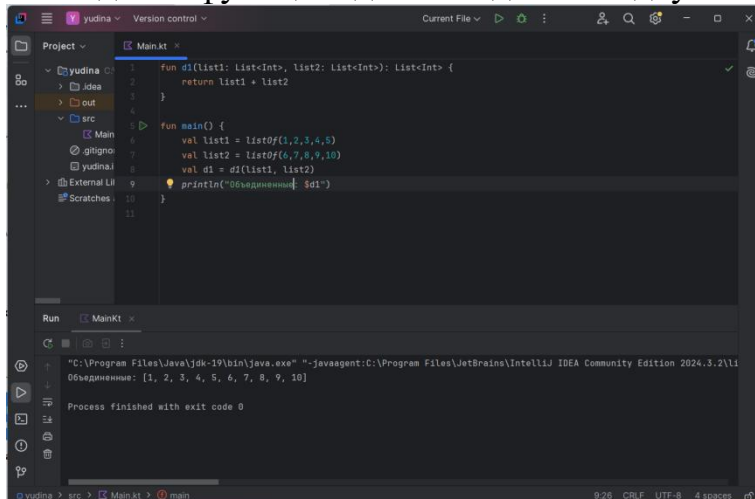
```
fun f1(n: List<Int>): Int {  
    return n.sum()  
}  
  
fun main() {  
    val number = listOf(1, 2, 3, 4, 5)  
    println("Сумма чисел: ${f1(number)}")  
}
```

2. Напишите функцию, которая получает на вход список целых чисел и возвращает разность самого большого и самого маленького из них.



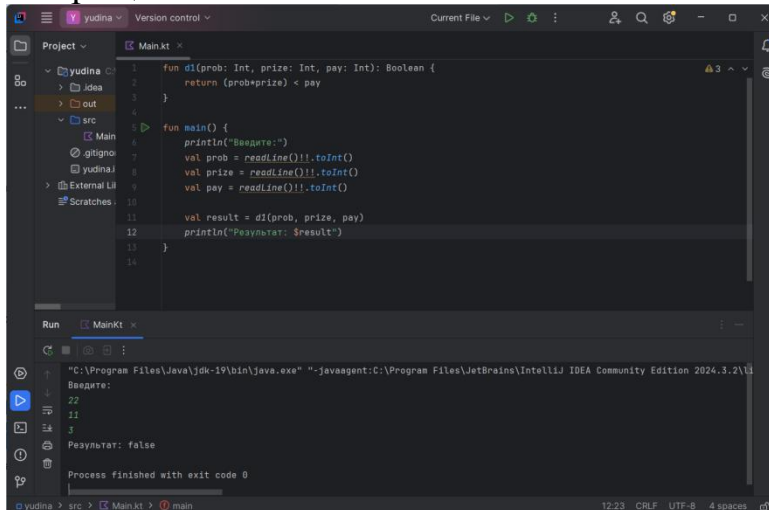
```
fun d1(n: List<Int>): Int {  
    val max = n.maxOrNull() ?: 0  
    val min = n.minOrNull() ?: 0  
    return max - min  
}  
  
fun main() {  
    val n = listOf(1, 2, 3, 4, 5)  
    val res = d1(n)  
    println("Разность: $res")  
}
```

### 3. Создайте функцию для объединения двух списков целых чисел.



```
fun d1(list1: List<Int>, list2: List<Int>): List<Int> {  
    return list1 + list2  
}  
  
fun main() {  
    val list1 = listOf(1,2,3,4,5)  
    val list2 = listOf(6,7,8,9,10)  
    val d1 = d1(list1, list2)  
    println("Объединенные: $d1")  
}
```

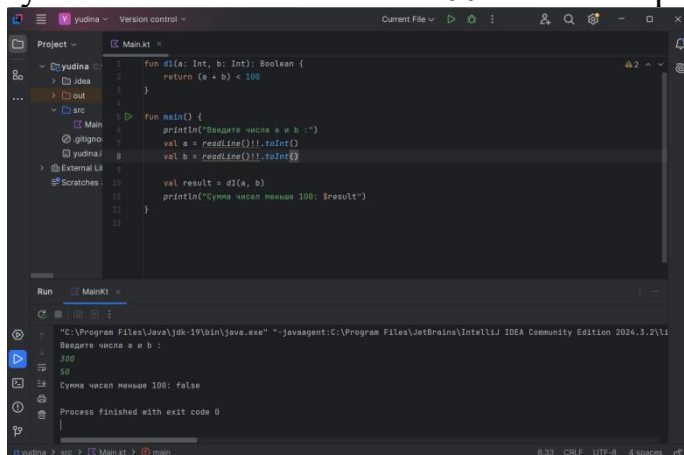
### 4. Создайте функцию, которая принимает три аргумента prob, prize, pay и возвращает True, если $\text{prob} * \text{prize} > \text{pay}$ , в противном случае возвращает False.



```
fun d1(prob: Int, prize: Int, pay: Int): Boolean {  
    return (prob*prize) < pay  
}  
  
fun main() {  
    println("Введите:")  
    val prob = readLine()!!.toInt()  
    val prize = readLine()!!.toInt()  
    val pay = readLine()!!.toInt()  
  
    val result = d1(prob, prize, pay)  
    println("Результат: $result")  
}
```

5. Создайте функцию, которая принимает три аргумента prob, prize, pay и возвращает True, если  $prob * prize > pay$ , в противном случае возвращает False.

6. Функция получает на вход два числа. Она должна вернуть True, если сумма этих чисел меньше 100 и False в противном случае.

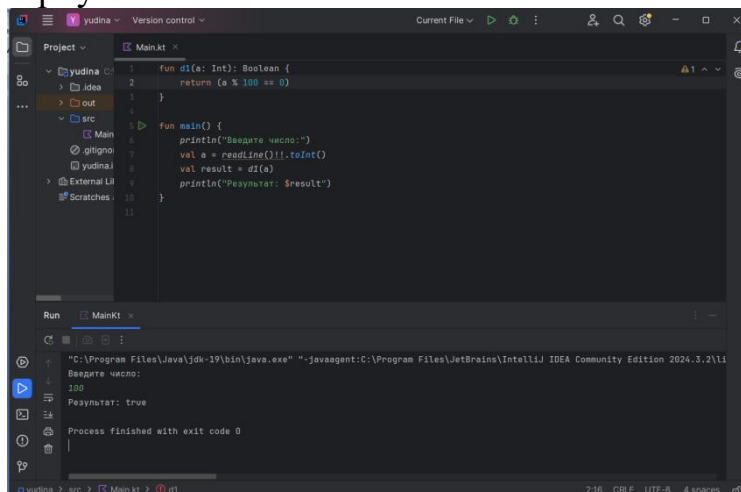


```
fun d1(a: Int, b: Int): Boolean {
    return (a + b) < 100
}

fun main() {
    println("Введите числа a и b :")
    val a = readLine()!!.toInt()
    val b = readLine()!!.toInt()

    val result = d1(a, b)
    println("Сумма чисел меньше 100: $result")
}
```

7. Напишите функцию, которая принимает целое число и возвращает True, если оно делится на 100. В противном случае функция должна вернуть False.

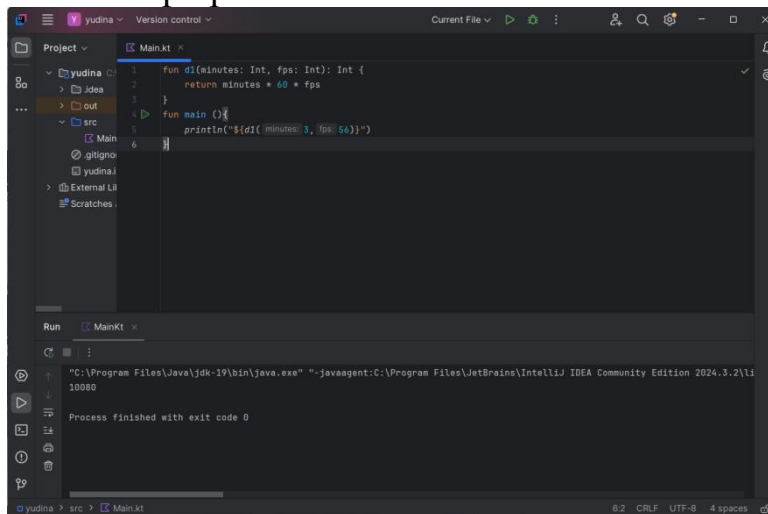


```
fun d1(a: Int): Boolean {
    return (a % 100 == 0)
}

fun main() {
    println("Введите число:")
    val a = readLine()!!.toInt()
    val result = d1(a)
}
```

```
println("Результат: $result")
}
```

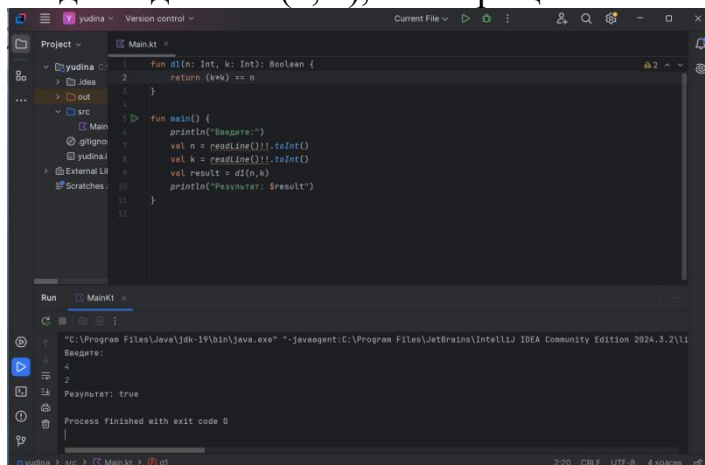
8. Напишите функцию, которая принимает количество минут и частоту кадров (FPS) и возвращает, сколько за это время кадров показывает компьютер при этом FPS.



```
fun d1(minutes: Int, fps: Int): Int {
    return minutes * 60 * fps
}

fun main() {
    println("${d1(3, 56)}")
}
```

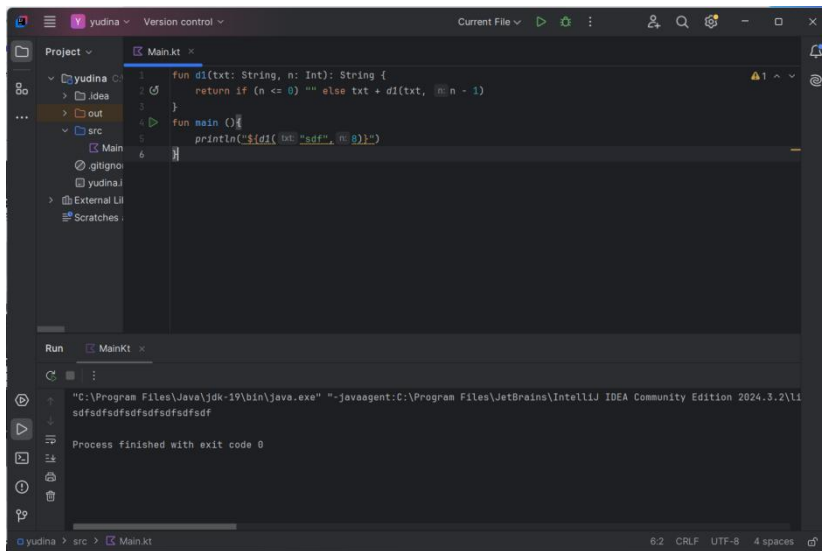
9. Напишите функцию, которая возвращает True, если  $k^k == n$  для входных данных (n, k), и возвращает False в противном случае.



```
fun d1(n: Int, k: Int): Boolean {
    return (k*k) == n
}

fun main() {
    println("Введите:")
    val n = readLine()!!.toInt()
    val k = readLine()!!.toInt()
    val result = d1(n, k)
    println("Результат: $result")
}
```

10. Создайте рекурсивную функцию, которая принимает два параметра и повторяет строку n количество раз. Первый параметр txt - это строка, которую нужно повторить, а второй параметр - количество повторений строки.



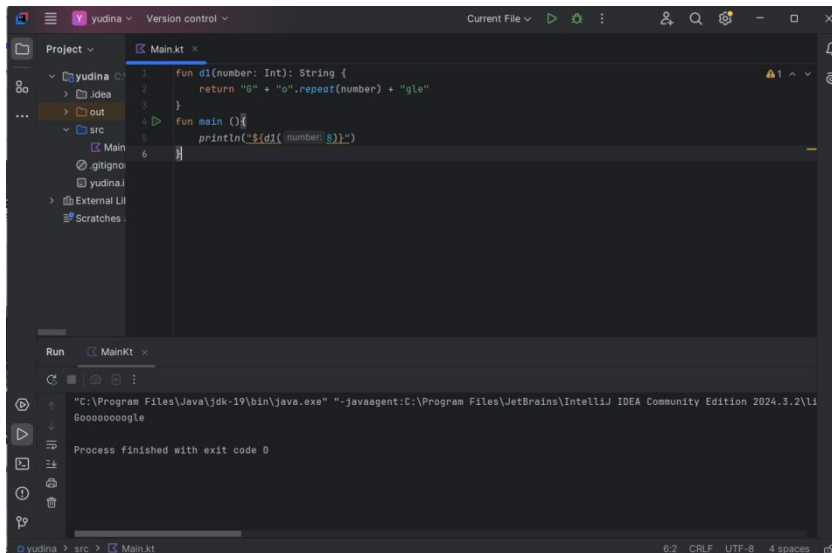
```

fun d1(txt: String, n: Int): String {
    return if (n <= 0) "" else txt + d1(txt, n - 1)
}
fun main () {
    println("${d1("sd", 8)}")
}

```

11. Создайте функцию, которая принимает уравнение (например, "1+1") и возвращает ответ.

12. Напишите функцию, которая принимает число number, и возвращает слово Google с количеством букв o, равным number.



```

fun d1(number: Int): String {
    return "G" + "o".repeat(number) + "gle"
}
fun main () {
    println("${d1(8)}")
}

```

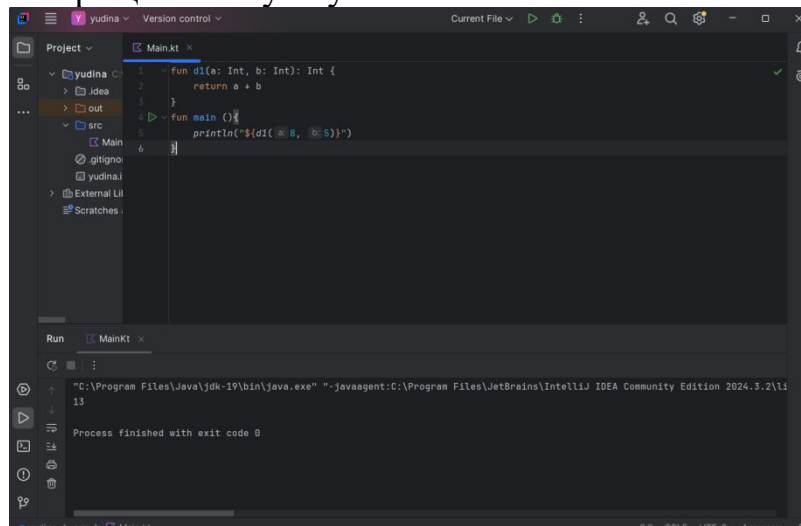
13.Приветствие: Напишите функцию, которая выводит "Привет, мир!" на экран.

```
"C:\Program Files\Java\jdk-19\bin\java.exe
Привет, мир!!!

Process finished with exit code 0

fun main() {
    println("Привет, мир!!!")
}
```

14.Сумма двух чисел: Создайте функцию, которая принимает два числа и возвращает их сумму.



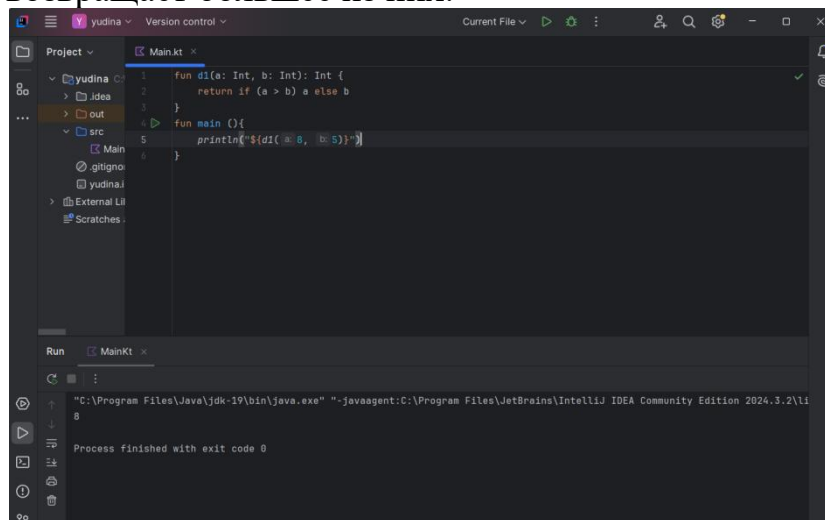
The screenshot shows the IntelliJ IDEA interface. The project 'yudina' is open, and the file 'Main.kt' is selected. The code in 'Main.kt' is as follows:

```
1 fun d1(a: Int, b: Int): Int {
2     return a + b
3 }
4 fun main () {
5     println("${d1(8, 5)}")
6 }
```

The Run window at the bottom shows the command: `"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.2\li` and the output: `13`. The process finished with exit code 0.

```
fun d1(a: Int, b: Int): Int {
    return a + b
}
fun main () {
    println("${d1(8, 5)}")
}
```

15.Сравнение чисел: Напишите функцию, которая принимает два числа и возвращает большее из них.



The screenshot shows the IntelliJ IDEA interface. The project 'yudina' is open, and the file 'Main.kt' is selected. The code in 'Main.kt' is as follows:

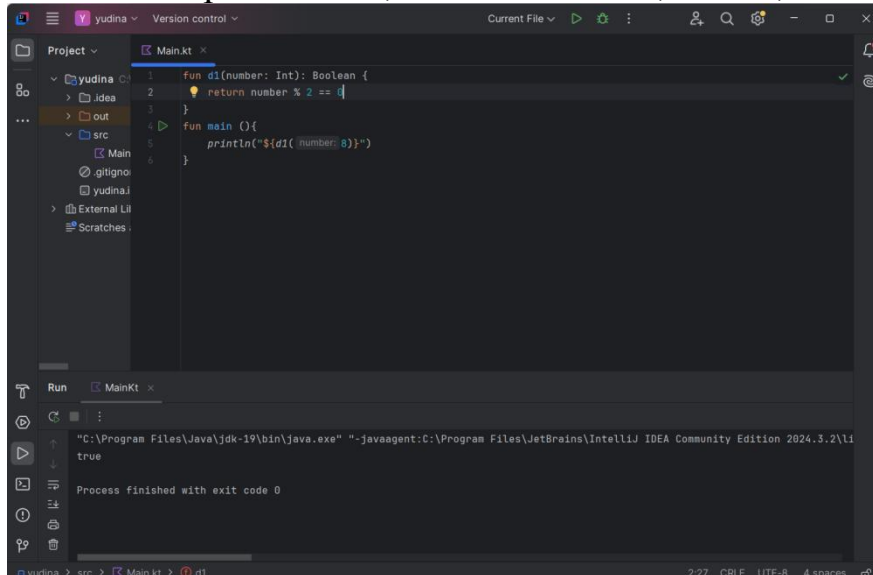
```
1 fun d1(a: Int, b: Int): Int {
2     return if (a > b) a else b
3 }
4 fun main () {
5     println("${d1(8, 5)}")
6 }
```

The Run window at the bottom shows the command: `"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.2\li` and the output: `8`. The process finished with exit code 0.

```
fun d1(a: Int, b: Int): Int {
    return if (a > b) a else b
}
fun main () {
    println("${d1(8, 5)}")
}
```

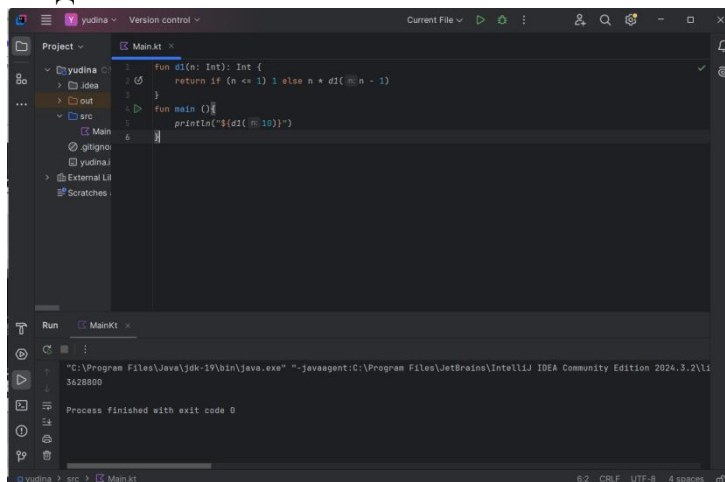
```
fun d1(a: Int, b: Int): Int {
    return if (a > b) a else b
}
fun main () {
    println("${d1(8, 5)}")
}
```

16.Определение четности: Создайте функцию, которая принимает одно число и возвращает true, если оно четное, и false, если нечетное.



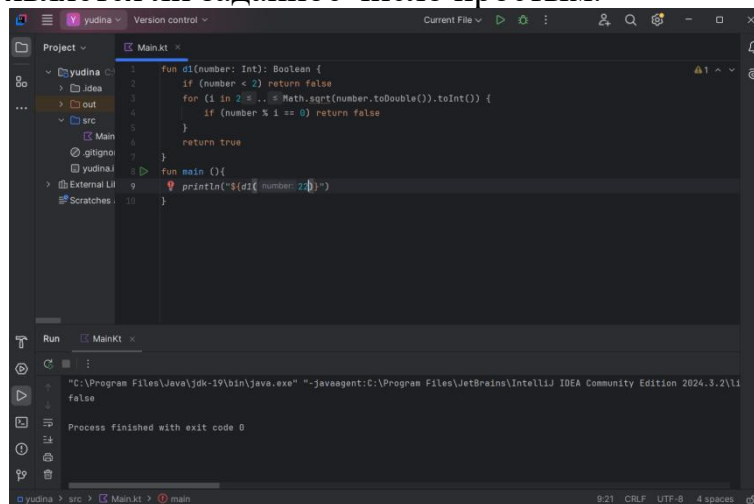
```
fun d1(number: Int): Boolean {
    return number % 2 == 0
}
fun main () {
    println("${d1(8)}")
}
```

17.Факториал числа: Реализуйте функцию, которая вычисляет факториал введенного числа.



```
fun d1(n: Int): Int {
    return if (n <= 1) 1 else n * d1(n - 1)
}
fun main () {
    println("${d1(10)}")
}
```

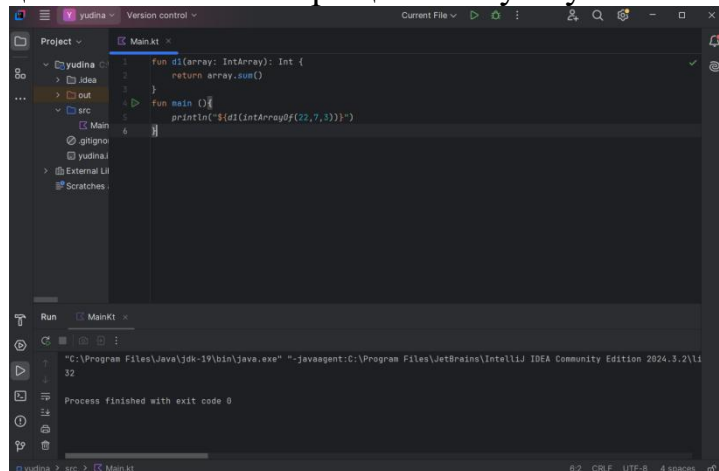
18. Проверка на простоту: Напишите функцию, которая проверяет, является ли заданное число простым.



```
fun d1(number: Int): Boolean {
    if (number < 2) return false
    for (i in 2..Math.sqrt(number.toDouble()).toInt()) {
        if (number % i == 0) return false
    }
    return true
}

fun main () {
    println("${d1(22)}")
}
```

19. Сумма чисел в массиве: Создайте функцию, которая принимает массив целых чисел и возвращает их сумму.

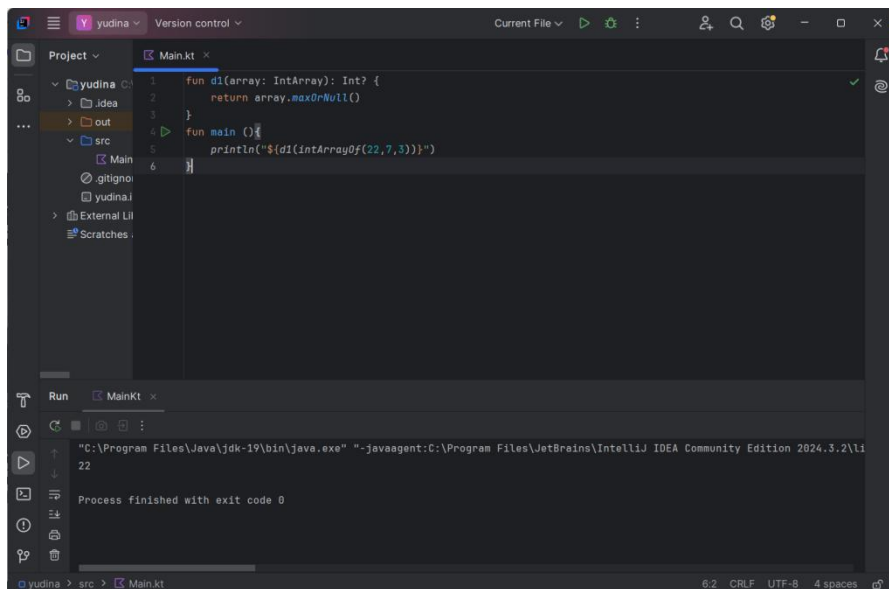


```
fun d1(array: IntArray): Int {
    return array.sum()
}

fun main () {
    println("${d1(intArrayOf(22,7,3))}")
}
```

20. Наибольшее число в массиве: Напишите функцию, которая находит и возвращает максимальное значение в массиве.

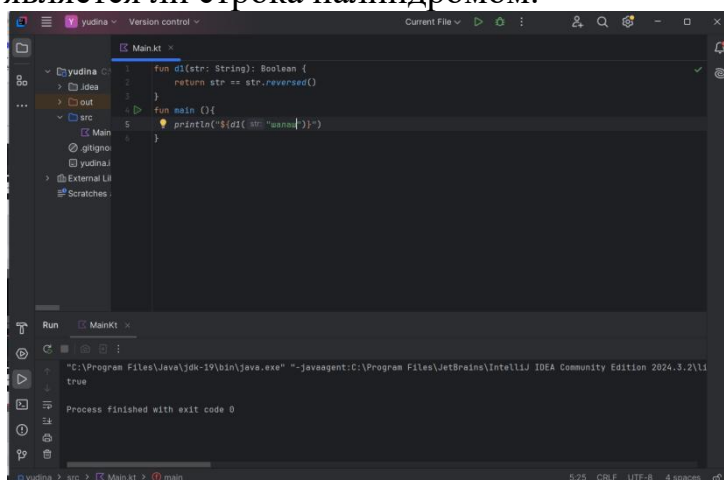




```
fun d1(array: IntArray): Int? {  
    return array.maxOrNull()  
}  
fun main () {  
    println("${d1(intArrayOf(22,7,3))}")  
}
```

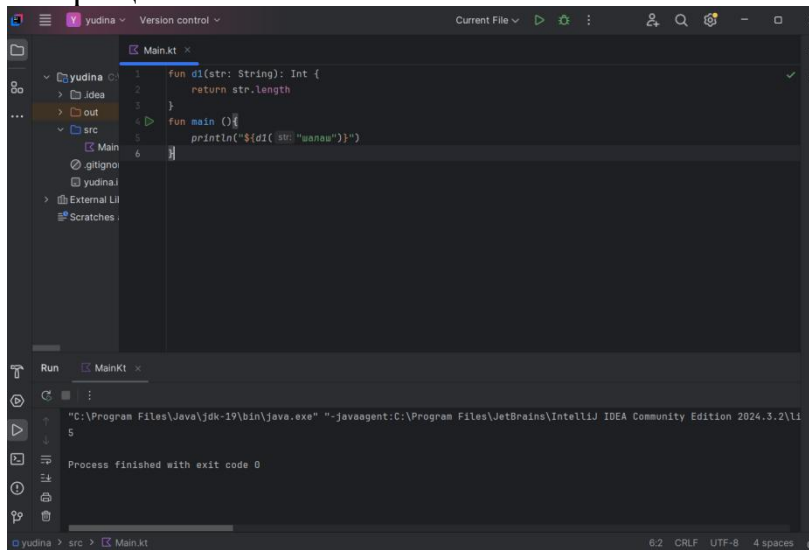
21.Сортировка массива: Создайте функцию, которая сортирует массив чисел в порядке возрастания.

22.Проверка палиндрома: Напишите функцию, которая проверяет, является ли строка палиндромом.



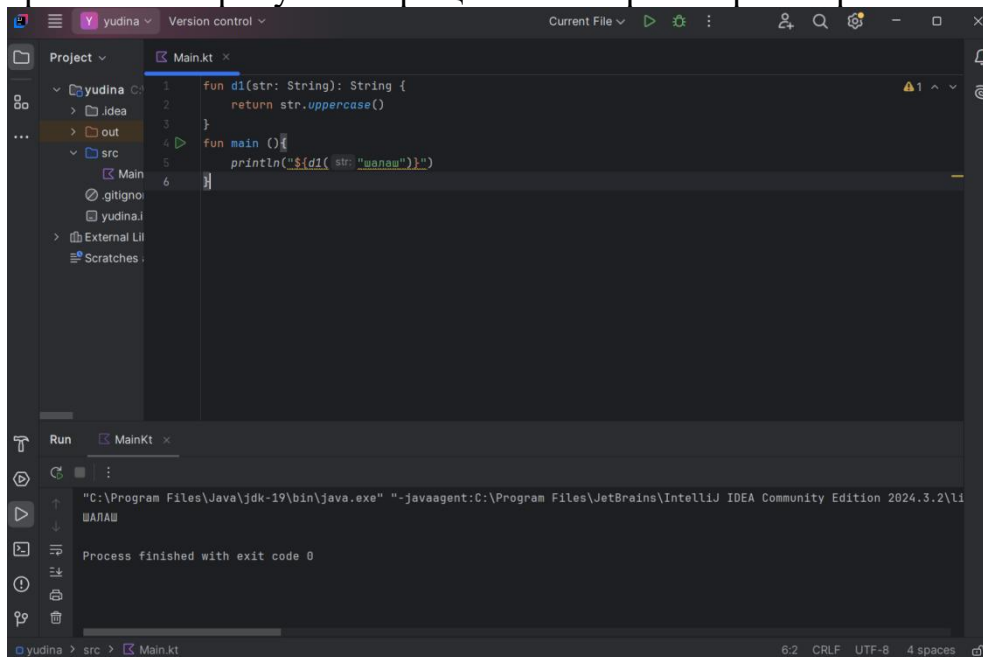
```
fun d1(str: String): Boolean {  
    return str == str.reversed()  
}  
fun main () {  
    println("${d1("шалаш")}")  
}
```

23.Количество символов: Создайте функцию, которая принимает строку и возвращает количество символов в ней.



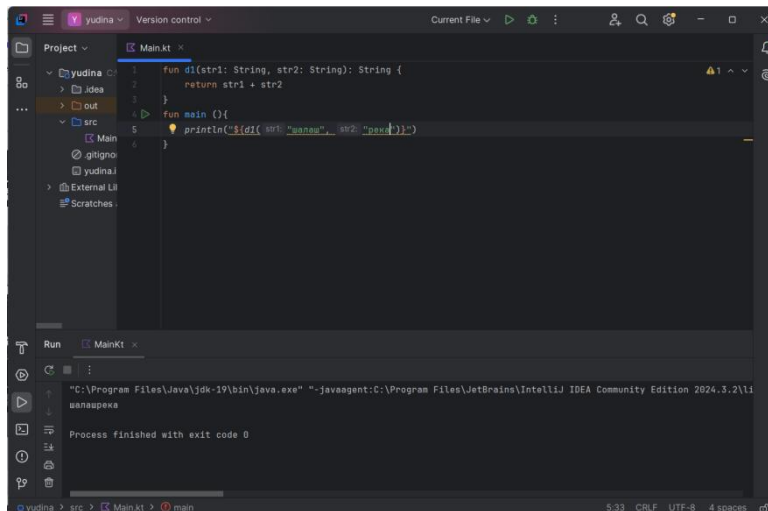
```
fun d1(str: String): Int {  
    return str.length  
}  
fun main () {  
    println("${d1("шалаш")}")  
}
```

24.Конвертация в верхний регистр: Напишите функцию, которая принимает строку и возвращает её в верхнем регистре.



```
fun d1(str: String): String {  
    return str.uppercase()  
}  
fun main () {  
    println("${d1("шалаш")}")  
}
```

25.Объединение строк: Создайте функцию, которая принимает две строки и возвращает их объединение.

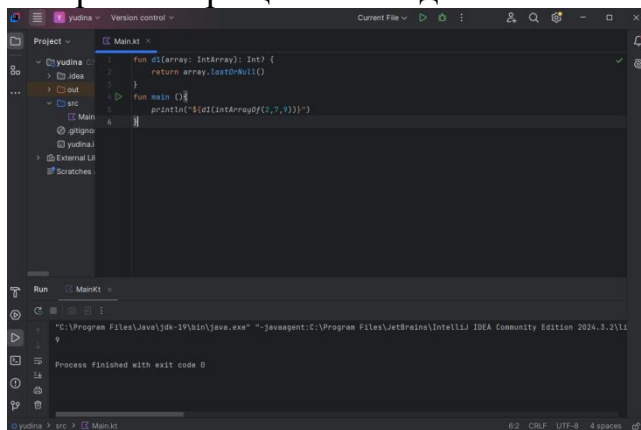


```

fun d1(str1: String, str2: String): String {
    return str1 + str2
}
fun main () {
    println("${d1("шалаш", "река")}")
}

```

26. Возвращение последнего элемента массива: Напишите функцию, которая возвращает последний элемент переданного массива.

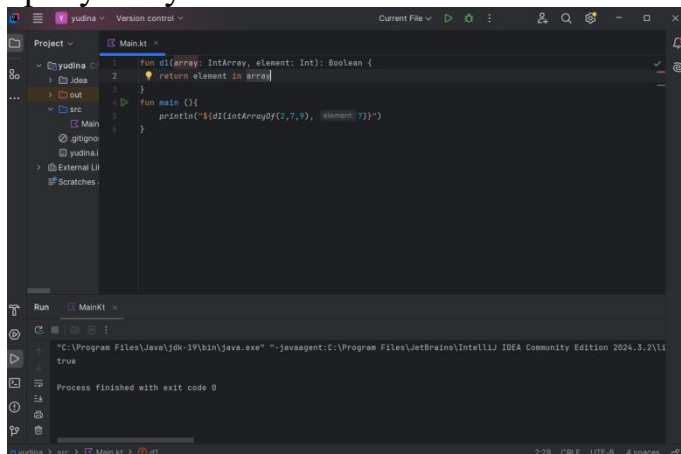


```

fun d1(array: IntArray): Int? {
    return array.lastOrNull()
}
fun main () {
    println("${d1(intArrayOf(2, 7, 9))}")
}

```

27. Проверка наличия элемента: Создайте функцию, которая проверяет, присутствует ли элемент в массиве.



```

Run: "C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3.2\lib\idea-agent.jar"
true
Process finished with exit code 0

```

```
fun d1(array: IntArray, element: Int): Boolean {
    return element in array
}

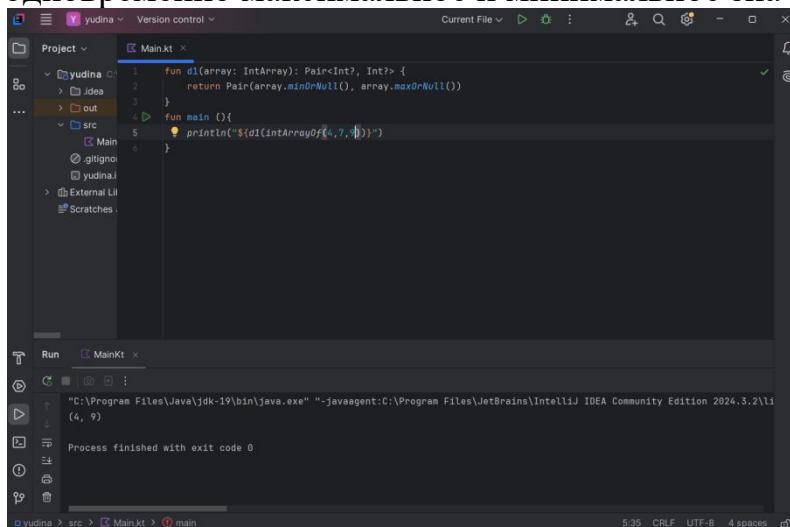
fun main () {
    println("${d1(intArrayOf(2,7,9), 7)}")
}
```

28.Создание массива от 1 до N: Напишите функцию, которая создает массив целых чисел от 1 до N.

```
fun d1(n: Int): IntArray {
    return (1..n).toList().toIntArray()
}

fun main () {
    println("${d1(3)}")
}
```

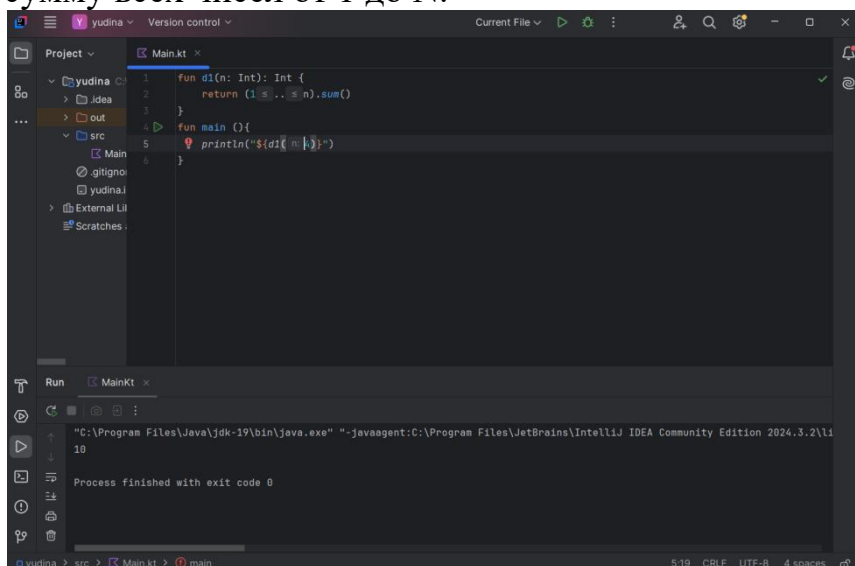
29.Максимум и минимум: Создайте функцию, которая находит одновременно максимальное и минимальное значения в массиве.



```
fun d1(array: IntArray): Pair<Int?, Int?> {
    return Pair(array.minOrNull(), array.maxOrNull())
}

fun main () {
    println("${d1(intArrayOf(4,7,9))}")
}
```

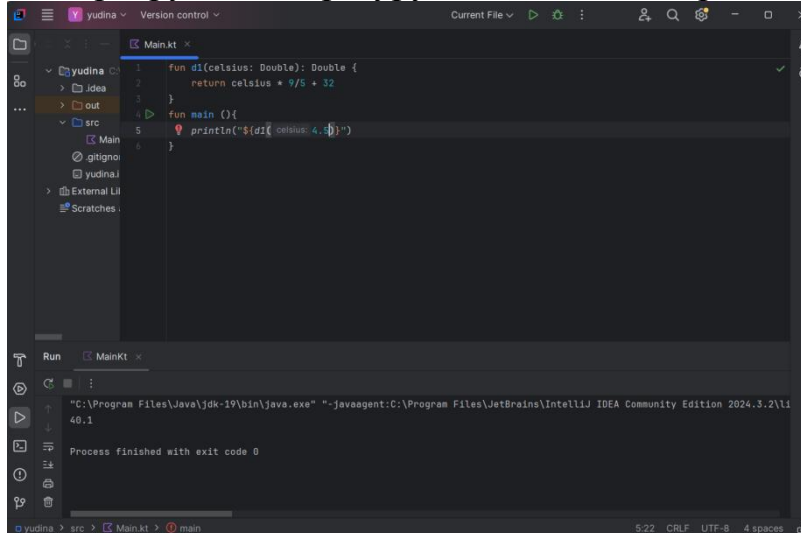
30.Сумма чисел от 1 до N: Напишите функцию, которая рассчитывает сумму всех чисел от 1 до N.



```
fun d1(n: Int): Int {
    return (1..n).sum()
}

fun main () {
    println("${d1(4)}")
}
```

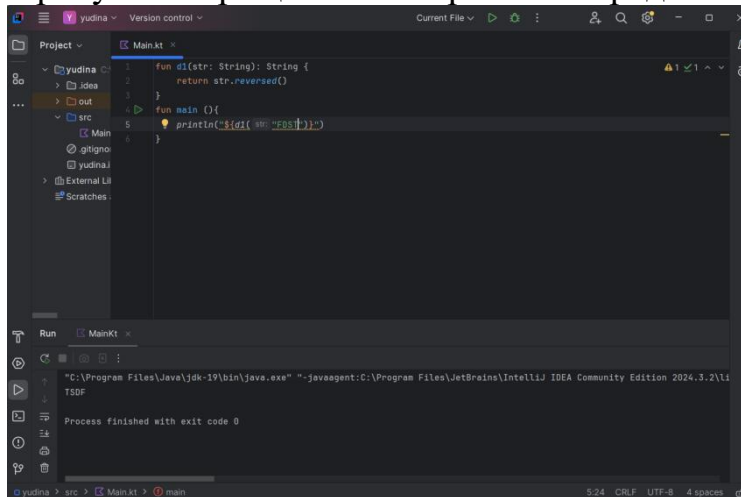
31. Преобразование Celsius в Fahrenheit: Создайте функцию, которая конвертирует температуру из Цельсия в Фаренгейт.



```
fun d1(celsius: Double): Double {
    return celsius * 9/5 + 32
}

fun main () {
    println("${d1(4.5)}")
}
```

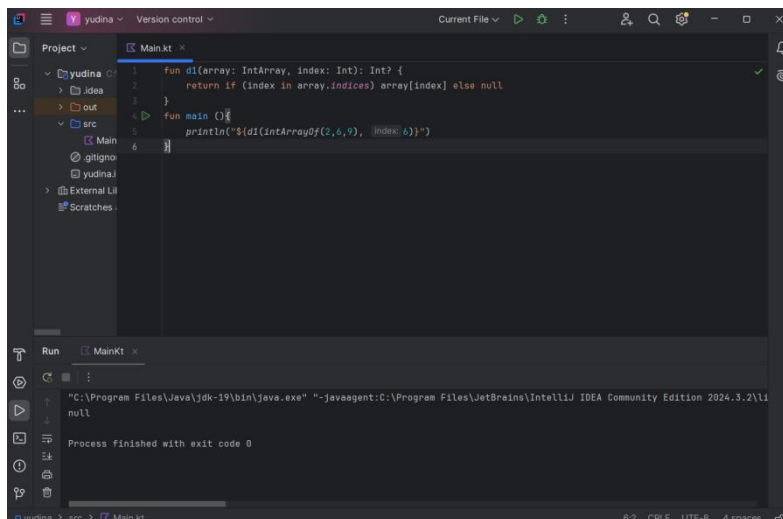
32. Обратный порядок строки: Напишите функцию, которая принимает строку и возвращает её в обратном порядке.



```
fun d1(str: String): String {
    return str.reversed()
}

fun main () {
    println("${d1("FDST")}")
}
```

33. Поиск элемента по индексу: Создайте функцию, которая находит элемент по индексу в массиве.

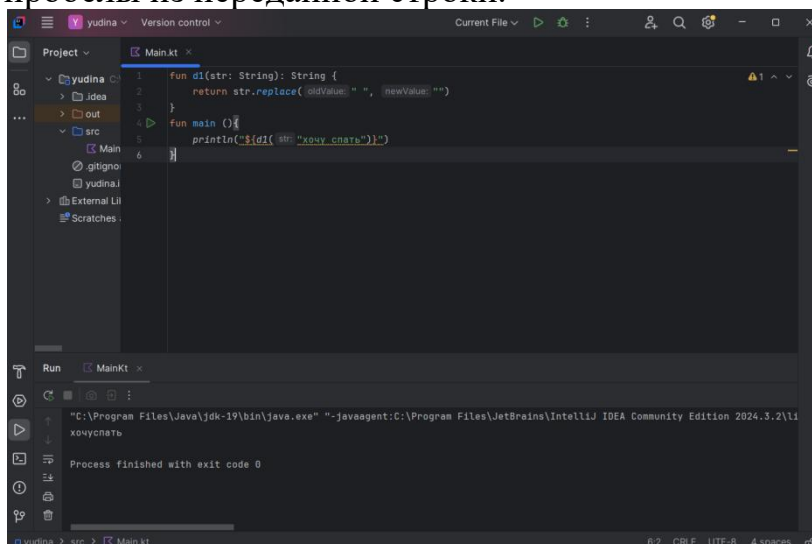


```

fun d1(array: IntArray, index: Int): Int? {
    return if (index in array.indices) array[index] else null
}
fun main () {
    println("${d1(intArrayOf(2,6,9), 6)}")
}

```

34. Удаление пробелов из строки: Напишите функцию, которая удаляет все пробелы из переданной строки.

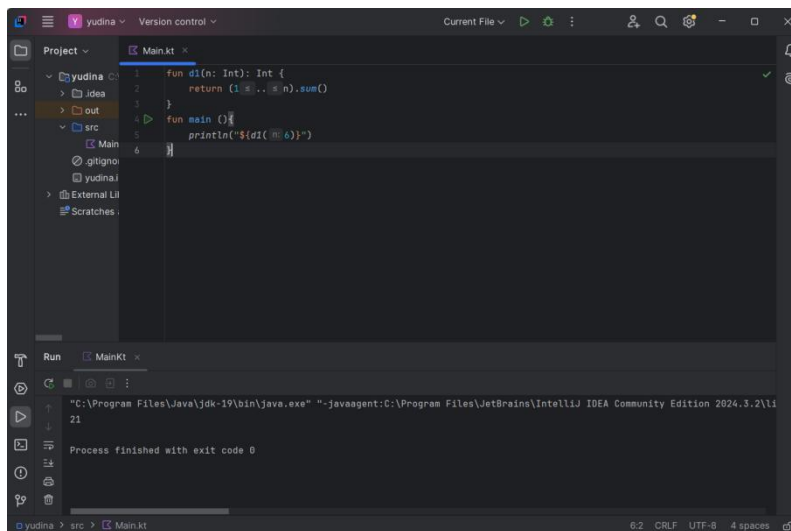


```

fun d1(str: String): String {
    return str.replace(" ", "")
}
fun main () {
    println("${d1("хочу спать")}")
}

```

35. Сумма первых N натуральных чисел: Создайте функцию, которая возвращает сумму первых N натуральных чисел.

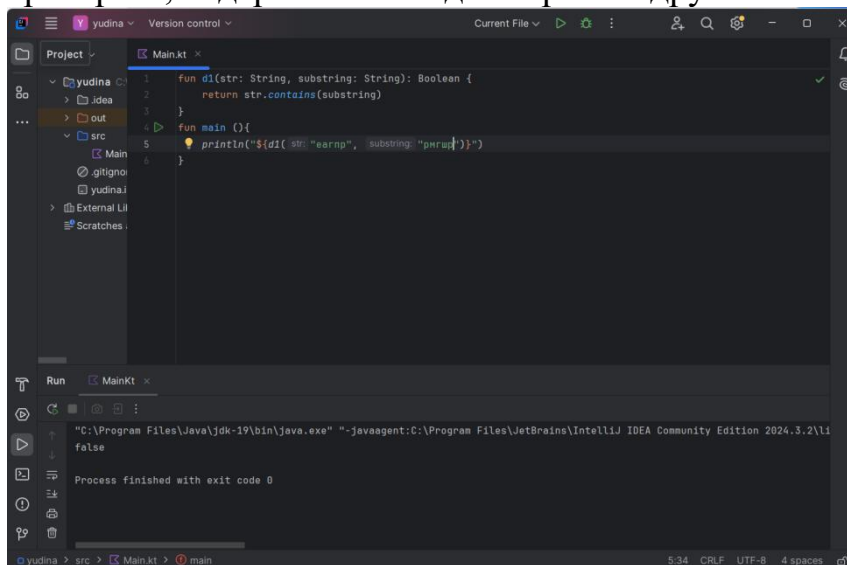


```

fun d1(n: Int): Int {
    return (1..n).sum()
}
fun main () {
    println("${d1(6)}")
}

```

36. Проверка строки на наличие подстроки: Напишите функцию, которая проверяет, содержится ли одна строка в другой.

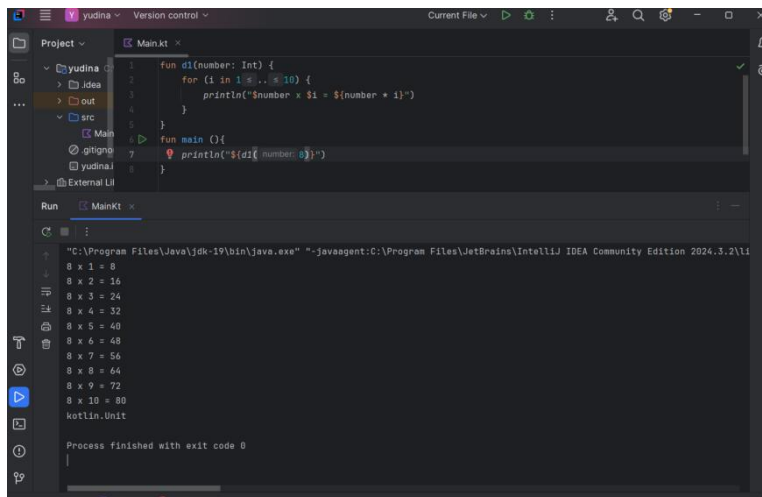


```

fun d1(str: String, substring: String): Boolean {
    return str.contains(substring)
}
fun main () {
    println("${d1("earnp", "pmrsh")}")
}

```

37. Печать таблицы умножения: Создайте функцию, которая выводит таблицу умножения от 1 до 10 для заданного числа.



```
fun dl(number: Int) {  
    for (i in 1..10) {  
        println("$number x $i = ${number * i}")  
    }  
}  
fun main () {  
    println("${dl(8)}")  
}
```

38.Нахождение длины строки: Напишите функцию, которая возвращает длину переданной ей строки.

39.Переворот массива: Создайте функцию, которая переворачивает массив целых чисел.

40.Копирование массива: Напишите функцию, которая копирует массив и возвращает новый массив.

41.Количество гласных в строке: Создайте функцию, которая считает количество гласных в строке.

42.Индекс первого вхождения: Напишите функцию, которая возвращает индекс первого вхождения элемента в массиве, или -1, если элемент не найден.