

№1 Генератор случайных чисел: Напишите программу, которая генерирует и выводит 10 случайных чисел от 1 до 100.

```
import kotlin.random.Random

fun main() {

    val randomNumbers = List(10) { Random.nextInt(1, 101) }

    println(randomNumbers)

}
```

№2 Строковый анализатор: Напишите программу, которая принимает строку и выводит количество гласных и согласных букв.

```
fun main() {

    val input = "Пример строки"

    val (vowels, consonants) = analyzeString(input)

    println("Гласные: $vowels, Согласные: $consonants")

}
```

```
fun analyzeString(input: String): Pair<Int, Int> {

    val vowels = "жизнь боль"

    var vowelCount = 0

    var consonantCount = 0

    for (char in input) {

        when {

            char.isLetter() && char in vowels -> vowelCount++

            char.isLetter() -> consonantCount++

        }

    }

    return Pair(vowelCount, consonantCount)

}
```

№3 Конвертер валют: Реализуйте программу, которая конвертирует одну валюту в другую (например, доллар в евро).

```
fun main() {
```

```

val amountInDollars = 154.0
val exchangeRate = 0.88
val amountInEuros = convertCurrency(amountInDollars, exchangeRate)
println("$amountInDollars долларов = $amountInEuros евро")
}

```

```

fun convertCurrency(amount: Double, rate: Double): Double {
    return amount * rate
}

```

№4 Проверка на анаграмму: Напишите функцию, которая проверяет, являются ли две строки анаграммами.

```

fun isAnagram(str1: String, str2: String): Boolean {
    val processedStr1 = str1.replace(" ", "").lowercase()
    val processedStr2 = str2.replace(" ", "").lowercase()
    return processedStr1.toList().sorted() == processedStr2.toList().sorted()
}

```

```

fun main() {
    println(isAnagram("listen", "silent")) // true
    println(isAnagram("hello", "world")) // false
    println(isAnagram("Anagram", "nag a ram")) // true
}

```

№5 Нахождение простых чисел: Реализуйте программу, которая находит и выводит все простые числа до заданного числа N.

```

fun main() {
    val n = 100 // Заданное число
    val primes = findPrimes(n)
    println("Простые числа до $n: $primes")
}

```

```

fun findPrimes(n: Int): List<Int> {

```

```
        return (2..n).filter { isPrime(it) }  
    }
```

```
fun isPrime(num: Int): Boolean {  
    for (i in 2 until num) {  
        if (num % i == 0) return false  
    }  
    return true  
}
```

№6 Сортировка строк: Напишите функцию, которая принимает массив строк и сортирует его по алфавиту.

```
fun main() {  
    val strings = arrayOf("яблоко", "банан", "апельсин")  
    val sortedStrings = sortStrings(strings)  
    println(sortedStrings.joinToString(", "))  
}
```

```
fun sortStrings(arr: Array<String>): Array<String> {  
    return arr.sortedArray()  
}
```

№7 Изменение регистра: Создайте программу, которая принимает строку и меняет регистр всех букв на противоположный.

```
fun main() {  
    val input = "Hello World!"  
    val toggledCase = toggleCase(input)  
    println(toggledCase)  
}
```

```
fun toggleCase(input: String): String {  
    return input.map {  
        if (it.isUpperCase()) it.toLowerCase() else it.toUpperCase()  
    }  
}
```

```
}.joinToString("")  
}
```

№8 Угадай число

```
import kotlin.random.Random
```

```
fun main() {  
    val numberToGuess = Random.nextInt(1, 101)  
    var guessedNumber: Int? = null  
  
    println("Угадайте число от 1 до 100")  
  
    while (guessedNumber != numberToGuess) {  
        print("Введите ваше число: ")  
        guessedNumber = readLine()?.toIntOrNull()  
  
        when {  
            guessedNumber == null -> println("Пожалуйста, введите корректное число.")  
            guessedNumber < numberToGuess -> println("Больше!")  
            guessedNumber > numberToGuess -> println("Меньше!")  
            else -> println("Поздравляю! Вы угадали число!")  
        }  
    }  
}
```

№9 Генератор паролей: Реализуйте упражнение по генерации случайного пароля заданной длины, используя цифры, буквы и специальные символы.

```
import kotlin.random.Random
```

```
fun main() {  
    val passwordLength = 12 // Заданная длина пароля  
    val password = generatePassword(passwordLength)  
    println("Сгенерированный пароль: $password")  
}
```

```
}
```

```
fun generatePassword(length: Int): String {  
    val chars =  
    "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789!@#%^&*()"  
    return (1..length).map { chars.random() }.joinToString("")  
}
```

№10 По заданной строке реализуйте функцию, которая возвращает самое длинное слово в этой строке. Если есть два или более слов одинаковой длины, верните первое самое длинное слово из строки. Игнорируйте знаки препинания. Входная строка не может быть пустой или незаполненным отображением.

```
fun longestWord(input: String): String {  
    require(input.isNotBlank()) { "Input string cannot be empty or blank" }  
    val words = input.replace(Regex("[^\\p{L}\\p{N} ]"), "").split("\\s+".toRegex())  
    return words.maxByOrNull { it.length } ?: ""  
}
```

```
fun main() {  
    println(longestWord("Привет мир!"))  
    println(longestWord("Казакова лучшая!"))  
}
```