**Отчет**

* prNumber(x: Int) : Int – произведение цифр «х»
* max(num:Int): Int – поиск максимальной цифры в «num»
* min(num:Int): Int – поиск минимальной цифры в «num»
* vzSimpleNumbers(x: Int, y: Int): Boolean – проверка взаимопростоты «х», «y»
* kolVzSimpleNumbers(x: Int): Int – поиск взаимнопростых чисел
* sumNumbers3(x: Int): Int – вычисление суммы цифр, делящихся на 3(для пользователя)
* sumNumbers3(x: Int, sum: Int ): Int - вычисление суммы цифр, делящихся на 3
* kolVpSimpleNumbers(x: Int, y: Int, counter: Int): Int – поиск количества взаимнопростых чисел «x» и «y»
* foundDel(x: Int): Int – поиск делителя «х», у которого наибольшего количество взаимнопростых чисел с «х»
* sumTwoMillionSimpleNumbers(beforeNumber: Int): Long – сумма простых чисел до заданного «beforeNumber»
* changeArray(a: Array<Boolean>, step: Int, size: Int, i: Int): Array<Boolean>
* - изменение массива таким образом, чтобы элементы с кратными step иденксами были равны «false»
* func(a: Array<Boolean>, step: Int): Array<Boolean> - изменение массива до тех пор, пока это возможно
* existsSimpleNumbers(a: Array<Boolean>): Boolean – проверка, есть ли в массиве простые числа
* simpleNumber(x: Int): Boolean – проверка, является ли «х» простым числом

Методы класса String

* compareTo - Сравнивает этот объект с указанным объектом для заказа. Возвращает ноль, если этот объект равен указанному другому объекту, отрицательное число , если оно меньше другого, или положительное число, если оно больше другого.
* equals - Указывает, является ли какой-то другой объект "равным" этому.
* get - Указывает, является ли какой-то другой объект "равным" этому. Реализация должна соответствовать следующим требованиям:
* subSequence – возващает новую последовательность символовб начиная с startIndex указаного до endindex.
* all – возвращает true, если все символы соответствуют заданному предикату.
* any - возвращает true, если последовательность символов содержит хотя бы один символ.
* associate – возращает map, содержащую пару ключ-значение.
* chunked – разбивает последовательность символов на список строк, каждая из которых не превышает заданного размера.
* decapitalize – возвращает копию строки с первой буквой в нижнем регистре.
* drop – возвращает строку с удаленными первыми «n» символами.
* lastOrNull – возвращает последний символ или «Null».
* mapNotNull – возвращает список, содержащий только ненулевые результаты применения данной функции преобразования к каждому символу в исходной последовательности символов.
* matches – возвращает «true», если эта последовательность символов соответсвует заданному регулярному выражению.