

String

main:

- Создать класс `StringUtil` в `src/main/java/...`.
- Добавить методы в `StringUtil` для решения нижеприведенных *specifications of tasks*.
- One *Specification of task* - one *method*.
- Название методов должно быть осмысленным.

test:

- Создать класс `StringUtilTest` в `src/test/java/...`.
- Добавить тесты в `StringUtilTest` для тестирования созданных методов.
- One *method* - many *tests*.
- Название тестовых методов формируется по принципу: `realNameMethod()` + `Test` = `realNameMethodTest()`

NOTE

Процесс разработки и тестирования - это параллельные процессы, т.е. идет разработка метода, затем он тестируется.

NOTE

При разработке рекомендуется использовать **TDD (Test Driven Development)** - разработка на основе тестов. Основной принцип: в начале пишутся тесты, а потом код, который проходит эти тесты

Specification of task 1

Используя методы класса `String` вывести строку, которая является результатом склеивания двух строк.

Specification of task 2

Найти индекс символа `x` (`x` – символ вводимый пользователем) в строке. Если такого символа нет, то `-1`.

Specification of task 3

Определить, являются ли две строки одинаковыми (регистр важен).

Specification of task 4

В строке удалить пробелы в начале и конце. Сделать что бы вся строка состояла из прописных букв.

Specification of task 5

Из строки извлечь подстроку начиная с 10 символа. Новая строка должна быть не более 23 символов.

Specification of task 6

Заменить все грустные смайлы :(в строке на весёлые :).

Specification of task 7

Написать метод принимающий 2 параметра:

- `string`
- `word`

Возвращающий `true`, если строка начинается и заканчивается этим словом, иначе `false`

Specification of task 8

Определить, сколько гласных букв в предложении.

Specification of task 9

Подсчитать количество всех точек, запятых и восклицательных знаков в строке.

Specification of task 10

Проверить, является ли введённая строка палиндромом, т.е. читается одинаково в обоих направлениях

Specification of task 11

Разбить строку на равные части по `n` символов и сохранить отдельные части в массив. Вывести этот массив.

Specification of task 12

Подсчитать количество слов в тексте. Учесть, что слова могут разделяться несколькими пробелами.

Specification of task 13

Дано имя, фамилия и отчество. Вернуть инициалы в формате **Ф.И.О.** Учесть, что входные параметры могут быть в любом регистре, а результирующая строка должна быть в верхнем.

Specification of task 14

Вернуть строку, которая содержит все имеющиеся в тексте цифры.

Specification of task 15

Напечатать только те буквы 2-х слов, которые есть только в одном из них (в том числе повторяющиеся). Например, если заданные слова **процессор** и **информация**, то ответом должно быть: **п е с с и ф м а я**.

Specification of task 16*

Заданы два массива строк. Определить, является ли содержимое этих массивов идентичным. Порядок расположения элементов значения не имеет.

Specification of task 17*

Написать два цикла, выполняющих многократное сложение строк, один с помощью оператора сложения и **String**, второй с помощью **StringBuilder** и метода **append**. Сравнить скорость их выполнения.

Specification of task 18*

Заменить несколько последовательных одинаковых символов в строке одним. Например: **aaabbcddef** → **abcdef**

Specification of task 19*

- Преобразовать строку, содержащую римскую запись числа, в арабскую запись числа.
- Границы чисел: от **1** до **3999**.
- Предусмотреть случай, когда меньший символ идёт перед большим, например: **CM == 900**.

Соответствие:

M 1000

D 500

C 100

L 50

X 10

V 5

I 1