

3. Strings and basics of text processing

Для закрепления навыков по данной теме решите следующие задачи.

Работа со строкой как с массивом символов

1. Дан массив названий переменных в camelCase. Преобразовать названия в snake_case.
2. Замените в строке все вхождения 'word' на 'letter'.
3. В строке найти количество цифр.
4. В строке найти количество чисел.
5. Удалить в строке все лишние пробелы, то есть серии подряд идущих пробелов заменить на одиночные пробелы. Крайние пробелы в строке удалить.

Работа со строкой как с объектом типа String или StringBuilder

1. Дан текст (строка). Найдите наибольшее количество подряд идущих пробелов в нем.
2. В строке вставить после каждого символа 'a' символ 'b'.
3. Проверить, является ли заданное слово палиндромом.
4. С помощью функции копирования и операции конкатенации составить из частей слова “информатика” слово “торт”.
5. Подсчитать, сколько раз среди символов заданной строки встречается буква “a”.
6. Из заданной строки получить новую, повторив каждый символ дважды.
7. Вводится строка. Требуется удалить из нее повторяющиеся символы и все пробелы. Например, если было введено "abc cde def", то должно быть выведено "abcdef".
8. Вводится строка слов, разделенных пробелами. Найти самое длинное слово и вывести его на экран. Случай, когда самых длинных слов может быть несколько, не обрабатывать.
9. Посчитать количество строчных (маленьких) и прописных (больших) букв в введенной строке. Учитывать только английские буквы.
10. Строка X состоит из нескольких предложений, каждое из которых кончается точкой, восклицательным или вопросительным знаком. Определить количество предложений в строке X.

Работа с регулярными выражениями (Pattern, Matcher)

1. Создать приложение, разбирающее текст (текст хранится в строке) и позволяющее выполнять с текстом **три** различных операции: отсортировать абзацы по количеству предложений; в каждом предложении отсортировать слова по длине; отсортировать лексемы в предложении по убыванию количества вхождений заданного символа, а в случае равенства - по алфавиту.
2. Дана строка, содержащая следующий текст (xml-документ):

```
<notes>
  <note id = "1">
    <to>Вася</to>
    <from>Света</from>
    <heading>Напоминание</heading>
    <body>Позвони мне завтра!</body>
  </note>
  <note id = "2">
    <to>Петя</to>
    <from>Маша</from>
    <heading>Важное напоминание</heading>
    <body/>
  </note>
</notes>
```

Напишите анализатор, позволяющий **последовательно** возвращать содержимое узлов xml-документа и его тип (открывающий тег, закрывающий тег, содержимое тега, тег без тела). Пользоваться готовыми парсерами XML для решения данной задачи нельзя.