МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

Одеський національний політехнічний університет

Інститут Комп’ютерних Систем

Кафедра Інформаційних Систем та Технологій

Протокол лабораторної роботи №3

з дисципліни об’єктно-орієнтоване програмування

на тему: « РАЗРАБОТКА КОНСОЛЬНЫХ ПРИЛОЖЕНИЙ »

Виконав студент групи

АД-201

Стицковський Н.Ю.

Прийняв

Рудніченко Н.Д.

Одеса, 2021

**Содержание**

Введение ……. 1

Теоретическая часть ….. 2

Практическая часть ……. 3

Вывод …….. 4

Литература …….. 5

ВВЕДЕНИЕ

ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ:

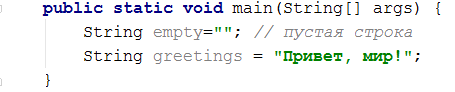
* Ознакомиться с классом String;
* Научиться использовать основные методы класса String.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

**ОБЪЕКТ**

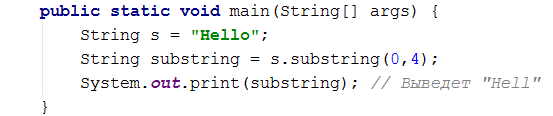
**Символьные строки**

Символьная строка в Java представляет собой последовательность символов в юникоде. В Java отсутствует встроенный тип для символьных строк. Вместо этого, в стандартной библиотеке Java содержится класс **String**. Каждая символьная строка, заключенная в кавычки, представляет собой экземпляр класса String:



**Подстроки**

С помощью метода **substring()** класса String можно выделить подстроку из отдельной строки. Например

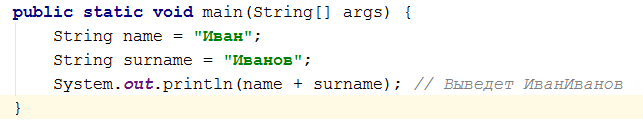


Второй параметр метода **substring()** обозначает позицию символа, который не следует включать в состав подстроки. В данном примере, новая строка будет содержать символы под индексами 0, 1, 2, 3. Длина новой подстроки будет равна , где а – первый параметр substring(), b – второй параметр. В данном случае, длина будет 4 - 0 = 4.



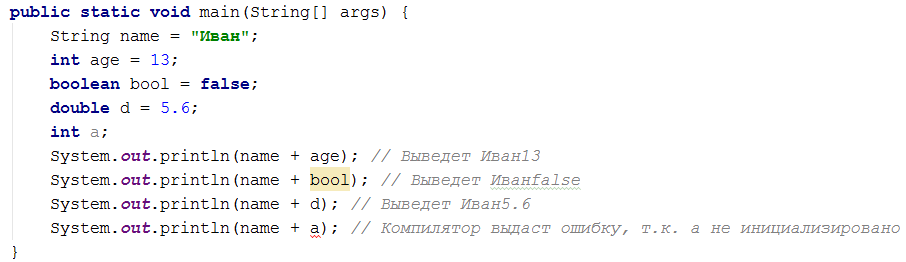
**Сцепление (конкатенация) строк**

В Java предоставляется возможность объединить две символьные строки, используя знак +.



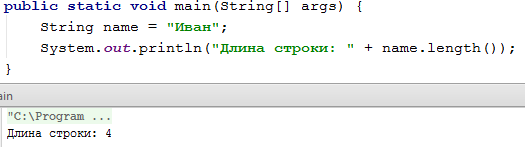
Обратите внимание на отсутствие пробела, строки сцепляются в том порядке, в котором были заданы.

Если вы сцепляете строку со значением, которое не является строковым, это значение преобразуется в строковое.



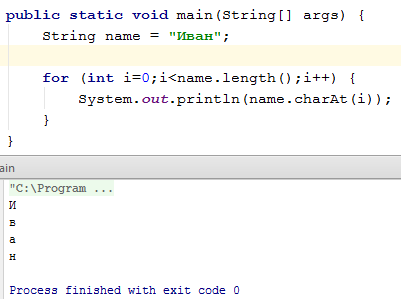
**Получить длину строки**

Чтобы получить длину строки, используется метод **length()**.



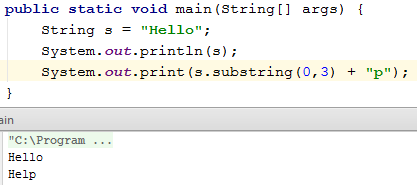
**Получить символ строки**

Чтобы получить определенный символ в строке, используется метод **charAt(index)**, где **index** – индекс символа в строке.



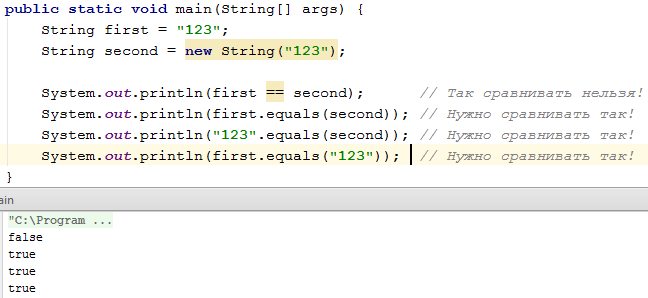
**Заменить символы в строке**

В Java нельзя изменять отдельные символы в строке, поэтому объекты типа String называются неизменяемыми (*immutable*). Так что, если требуется заменить символьную строку в переменной с «Hello» на «Help», этого нельзя добиться простой заменой символов. Для этого нужно создать новую строку, которая будет содержать неизменяемую часть старой + изменяемая часть.



**Проверка символьных строк на равенство**

Чтобы проверить две символьные строки на равенство, достаточно вызвать метод **equals()**. Так, выражение **s.equals(t)** возвращает **true**, если строки **s** и **t** равны и **false** в ином случае. В качестве s и t могут выступать переменные или литералы.



**Полная документация по классу String с описанием методов** –

<http://docs.oracle.com/javase/7/docs/api/java/lang/String.html>

# ПРАКТИЧЕСКАЯ ЧАСТЬ

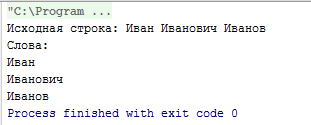
**ЗАДАНИЕ НА ЛАБОРАТОРНУЮ РАБОТУ:**

1. Реализуйте метод, который принимает на вход строку и возвращает true, если строка заканчивается на «ed» и false в противном случае;

2. Реализуйте метод, который принимает на вход строку и возвращает сумму цифр, которые были найдены в этой строке (если цифр нет – возвращается 0);

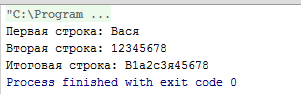
3. Реализуйте метод, который принимает на вход строку и возвращает длину наиболее длинного «блока» символов в этой строке («блок» - множество **одинаковых** **идущих подряд** символов, например, в строке «**aa**BCS» - самый длинный блок «аа» длиной 2 символа);

4. Реализуйте метод, который принимает на вход строку и выводит в консоль найденные слова в этой строке (слова отделяются пробелами, кроме случаев когда слово стоит в начале или в конце).



5. Реализуйте метод, который принимает на вход две строки (А и Б) и возвращает строку, которая состоит поочередно из i-ых символов каждой из строк. Например, сначала первый символ первой строки, потом первый символ второй строки, потом второй символ первой строки, потом второй символ второй строки и т.д. Если символы в одной из строк закончатся, оставшиеся символы другой строки дописываются в конец.

Пример:



Код программы:

public class Lab3Stytskovskyi {

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// TODO code application logic here

String fred = "Fred";

if (checkEd(fred) == true)

{

System.out.println("true");

}

String nums = "fr3dd13";

System.out.println(sumNumbers(nums));

String block = "aaanndddd";

System.out.println(longestBlock(block));

String words = "I love cake";

separateWords(words);

String wordA = "Java", wordB = "Script";

System.out.println(combineSymbols(wordA, wordB));

}

public static boolean checkEd(String s) //Метод 1

{ //Смысла в нём мало, но раз надо - держите

return s.endsWith("ed");

}

public static int sumNumbers(String s) //Метод 2

{ //Хотя-бы не просто использует одну встроенную функцию

int sum = 0;

for (int i = 0; i < s.length(); i++)

{

if (Character.isDigit(s.charAt(i)))

{

sum += Character.getNumericValue(s.charAt(i));

}

}

return sum;

}

public static int longestBlock (String s) //Метод 3

{ //Ничего сложного, но хотя-бы пришлось немного подумать

int longest = 0, temp = 1;

for (int i = 0; i < s.length(); i++)

{

if (i+1 < s.length() && s.charAt(i + 1) == s.charAt(i))

{

temp++;

}

else

{

if (longest < temp)

longest = temp;

temp = 1;

}

}

return longest;

}

public static void separateWords(String s) //Метод 4

{

String word = "";

for (int i = 0; i < s.length(); i++)

{

if (s.charAt(i) != ' ')

{

word += s.charAt(i);

}

else

{

System.out.println(word);

word = "";

}

if (i == s.length() - 1)

System.out.println(word);

}

}

public static String combineSymbols(String a, String b) //Метод 5

{

String combined = "";

int length;

if (a.length() > b.length())

length = a.length();

else

length = b.length();

for (int i = 0; i < length; i++)

{

if (i < a.length())

combined += a.charAt(i);

if (i < b.length())

combined += b.charAt(i);

}

return combined;

}

}

Результат работы программы:

