­­МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

­­­

Лабораторна робота №4

з курсу «Об’єктно-орієнтоване програмування»

для студентів базового напрямку 6.08.04 "Комп’ютерні науки"

(заочна форма навчання)

Варіант 14

Виконав студент гр. КНз-2

Чалий Михайло

­­

Львів 2014

# Робота з рядками в Java

## Мета роботи

Отримати навички програмування з використанням стрічок у мові Java.

## Теоретичні відомості

Важливе місце в обробці інформації займає робота з текстами. Текстові рядки у мові Java є об’єктами. Вони представляються екземплярами класу String або класу StringBuffer. Звичайно, можна занести текст у масив символів типу char або навіть у масив байтів типу byte, але тоді не буде можливості використати готові методи роботи з текстовими рядками.

Для чого в мову введені два класи для зберігання рядків? В об’єктах класу String зберігаються рядки-константи незмінної довжини і змісту. Це значно прискорює оброблення рядків і дозволяє економити пам’ять, розділяючи рядок між об’єктами, які його використовують. Довжину рядків, які зберігаються в об’єктах класу StringBuffer, можна змінювати, вставляючи і додаючи рядки і символи, видаляючи підрядки або з’єднуючи декілька рядків в один. У багатьох випадках, коли потрібно змінити довжину рядка типу String , компілятор Java неявно перетворює його в тип StringBuffer, змінює довжину, потім перетворює назад у тип String. Наприклад, наступна дія

String s = "Це" + " один " + "рядок";

компілятор виконує наступним чином:

String s = new StringBuffer().append("Це").append(" один ").append("рядок").toString();

Створюється об’єкт класу StringBuffer, у нього послідовно додаються рядки "Це", " один ", "рядок", і одержаний об’єкт класу StringBuffer буде приведений до типу String методом toString (). Нагадаємо, що символи у рядках зберігаються у кодуванні Unicode, в якому кожний символ займає два байти. Тип кожного символу - char

## Завдання

Дано рядок, oо складається із слів, розділених пропусками. Вивести рядок, що містить ці ж слова, розділених одним пропуском, але розміщені у зворотному порядку.

## Реалізація

Лістінг 1. SentenceReverce.java

**public** **class** SentenceReverse {

**public** **static** **void** main(String[] args) {

String input = "Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec a diam lectus";

System.*out*.printf("Input: %s%n", input);

String[] tokens = input.split(" ");

**int** tokenNum = tokens.length;

StringBuilder resultBuilder = **new** StringBuilder();

**for**(**int** i = tokenNum - 1; i > 0; i--){

resultBuilder.append(tokens[i]);

**if** (i != 0){

resultBuilder.append(" ");

}

}

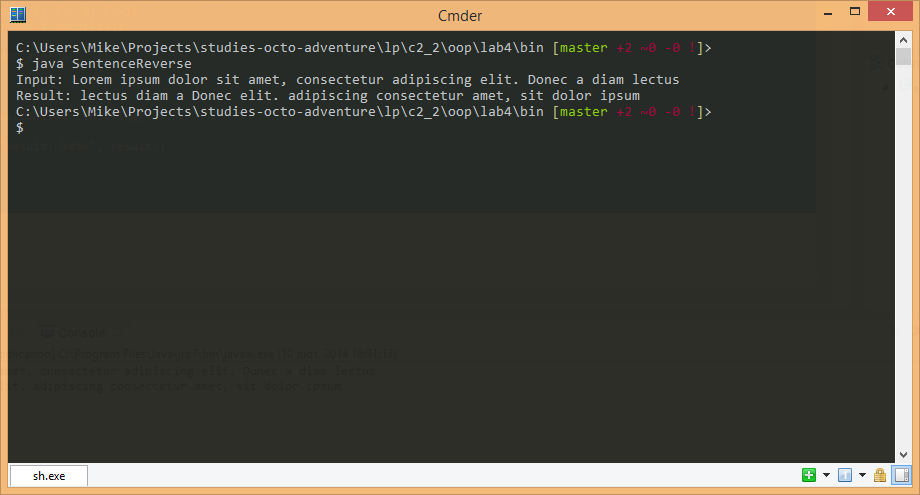
String result = resultBuilder.toString();

System.*out*.printf("Result: %s%n", result);

}

}

## Результат



## Висновки

Отримав навички програмування з використанням стрічок у мові Java.