

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«Санкт-Петербургский государственный электротехнический университет
“ЛЭТИ” им.В.И.Ульянова (Ленина) »

Кафедра МОЭВМ

ОТЧЕТ
по лабораторно-практической работе № 1
«Знакомство со средой разработки Java - приложений»
по дисциплине «Объектно - ориентированное
программирование на языке Java»

Выполнил Костенко А.М.

Факультет КТИ

Группа № 3311

Подпись преподавателя _____

Санкт-Петербург
2024 г

Цель работы

Освоение среды разработки Eclipse, программирование, запуск и отладка консольного приложения.

Описание задания

1. Установите на свой компьютер JDK и Eclipse.
2. Создайте проект.
3. Создайте приложение, в котором объявите статический массив целых чисел.
4. В методе `main` проинициализируйте этот массив и напишите алгоритм его упорядочения по возрастанию (убыванию). После упорядочения выведите элементы массива на консоль.
5. Добавьте в приложение комментарии документации.
6. Запустите приложение и проверьте результат его работы.
7. Используйте режим отладки и проанализируйте с помощью команды `Run/Watch`, как в процессе упорядочения меняются значения элементов массива, а с помощью команды `Run/Inspect` – значение операции сравнения элементов массива.
8. Сгенерируйте документацию с помощью `Javadoc` и просмотрите ее в браузере

Описание проверки работоспособности приложения

Полную работоспособность приложения можно увидеть на примере 1 и 2.

Пример 1

← →

Java_Lab01

DB

Main.java

edu.java.lab1 > src > edu > java > lab1 > Main.java

1 package edu.java.lab1;

2

3 import java.util.Random;

4 import java.util.Arrays;

5

6 /**

7 * Main class for the application.

8 *

9 * @author Kostenko Aleksey 3311

10 * @version 1.0

11 */

12 Codeium: Refactor | Explain

13 public class Main {

14

15 /**

16 * The main method of the application. Generates an array of 10 random numbers,

17 * sorts them and prints them in ascending order.

18 *

19 * @param args command line arguments

20 */

21 Run | Debug | Codeium: Refactor | Explain | X

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

23 // Create an array of 10 integers

24 int[] array = new int[10];

25

26 // Create a Random object to generate random numbers

27

ПРОБЛЕМЫ

ВЫХОДНЫЕ ДАННЫЕ

ПОРТЫ

КОММЕНТАРИИ

КОНСОЛЬ ОТЛАДКИ

ТЕРМИНАЛ

+ ▾

Run: Main

🗑

⋮

⌵

×

🍏 ~/Documents/prog/labs_JAVA/Java_Lab01/Java_Lab01

✓ 15:32:13

/usr/bin/env /Library/Java/JavaVirtualMachines/jdk-22.jdk/Contents/Home/bin/java --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/odious/Library/Apple/Support/Code/User/workspaceStorage/1183fd7a67b63608a430182a1899bac0/redhat.java/jdt_ws/Java_Lab01_ffecafid/bin edu.java.lab1.Main

Original Array: [-343, 623, -701, 839, 666, 558, -942, 915, 369, -730]

Sorted Array (ascending): [-942, -730, -701, -343, 369, 558, 623, 666, 839, 915]

Sorted Array (descending): [915, 839, 666, 623, 558, 369, -343, -701, -730, -942]

🍏 ~/Documents/prog/labs_JAVA/Java_Lab01/Java_Lab01

✓ 15:32:14

🔧 Launchpad

🔍 0 0

🔊 0

🔗

Java: Ready

Строка 36, столбец 27

Пробелов: 4

UTF-8

CRLF

{ } Java

Codeium: {...}

Prettier

Пример 2

← →

Java_Lab01

DB

Main.java

edu.java.lab1 > src > edu > java > lab1 > Main.java

1 package edu.java.lab1;

2

3 import java.util.Random;

4 import java.util.Arrays;

5

6 /**

7 * Main class for the application.

8 *

9 * @author Kostenko Aleksey 3311

10 * @version 1.0

11 */

12 Codeium: Refactor | Explain

13 public class Main {

14

15 /**

16 * The main method of the application. Generates an array of 10 random numbers,

17 * sorts them and prints them in ascending order.

18 *

19 * @param args command line arguments

20 */

21 Run | Debug | Codeium: Refactor | Explain | X

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

23 // Create an array of 10 integers

24 int[] array = new int[10];

25

26 // Create a Random object to generate random numbers

27

ПРОБЛЕМЫ

ВЫХОДНЫЕ ДАННЫЕ

ПОРТЫ

КОММЕНТАРИИ

КОНСОЛЬ ОТЛАДКИ

ТЕРМИНАЛ

+ ▾

Run: Main

🗑

⋮

⌵

×

🍏 ~/Documents/prog/labs_JAVA/Java_Lab01/Java_Lab01

✓ 15:31:37

/usr/bin/env /Library/Java/JavaVirtualMachines/jdk-22.jdk/Contents/Home/bin/java --enable-preview -XX:+ShowCodeDetailsInExceptionMessages -cp /Users/odious/Library/Apple/Support/Code/User/workspaceStorage/1183fd7a67b63608a430182a1899bac0/redhat.java/jdt_ws/Java_Lab01_ffecafid/bin edu.java.lab1.Main

Original Array: [-252, -389, 168, 113, 137, -913, 369, 690, -626, 75]

Sorted Array (ascending): [-913, -626, -389, -252, 75, 113, 137, 168, 369, 690]

Sorted Array (descending): [690, 369, 168, 137, 113, 75, -252, -389, -626, -913]

🍏 ~/Documents/prog/labs_JAVA/Java_Lab01/Java_Lab01

✓ 15:31:38

🔧 Launchpad

🔍 0 0

🔊 0

🔗

Java: Ready

Строка 36, столбец 27

Пробелов: 4

UTF-8

CRLF

{ } Java

Codeium: {...}

Prettier

Текст программы

```
package edu.java.lab1;

import java.util.Random;
import java.util.Arrays;

/**
 * Main class for the application.
 *
 * @author Kostenko Aleksey 3311
 * @version 1.0
 */
public class Main {

    /**
     * The main method of the application. Generates an array of 10 random numbers,
     * sorts them and prints them in ascending order.
     *
     * @param args command line arguments
     */
    public static void main(String[] args) {
        // Create an array of 10 integers
        int[] array = new int[10];

        // Create a Random object to generate random numbers
        Random rand = new Random();

        // Fill the array with random numbers between -1000 and 1000
        for (int i = 0; i < array.length; i++) {
            array[i] = rand.nextInt(-1000, 1000);
        }

        // Print the original array
        System.out.println("Original Array: " + Arrays.toString(array));

        // Sort the array in ascending order
        method(array, true);

        // Print the sorted array in ascending order
        System.out.println("Sorted Array (ascending): " + Arrays.toString(array));

        // Sort the array in descending order
        method(array, false);

        // Print the sorted array in descending order
```

```

        System.out.println("Sorted Array (descending): " + Arrays.toString(array));
    }

    /**
     * Sorts the given array in ascending or descending order based on the flag.
     *
     * @param array the array to be sorted
     * @param flag true for ascending order, false for descending order
     */
    public static void method(int[] array, boolean flag) {
        // Get the length of the array
        int len = array.length;

        // Flag to check if any swaps were made in the inner loop
        boolean n;

        // Outer loop to iterate over the array
        for (int i = 0; i < len - 1; i++) {
            // Initialize the flag to false
            n = false;

            // Inner loop to compare adjacent elements
            for (int j = 0; j < len - i - 1; j++) {
                // Check if the elements need to be swapped based on the flag
                if ((array[j] > array[j + 1] && flag) || (array[j] < array[j + 1] && !flag)) {
                    // Swap the elements
                    int temp = array[j];
                    array[j] = array[j + 1];
                    array[j + 1] = temp;

                    // Set the flag to true
                    n = true;
                }
            }

            // If no swaps were made in the inner loop, the array is already sorted
            if (!n) break;
        }
    }
}

```

Ссылка на лабораторную работу
https://disk.yandex.ru/d/mX_9zLzuaDZLBA