

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»

Выполнила Ефременко В.А.

Факультет КТИ

Группа № 3311

Подпись преподавателя _____

Санкт-Петербург

2024 г

Цель работы

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

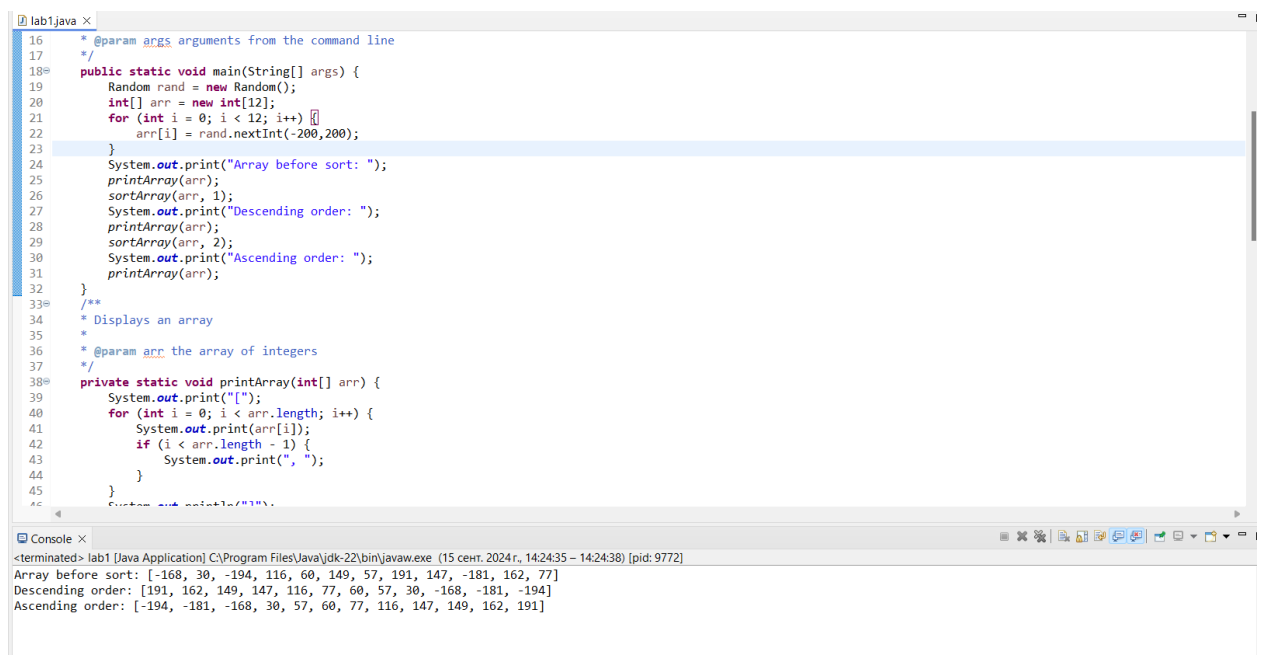
Описание задания

1. Установить на свой компьютер JDK и Eclipse.
2. Создать проект.
3. Создать приложение, в котором объявлен статический массив целых чисел.
4. В методе `main` проинициализируйте этого массива и напишите алгоритм его упорядочения по возрастанию (убыванию).

Описание проверки работоспособности приложения

Полную работоспособность приложения можно увидеть на примере 1 и 2.

Пример 1



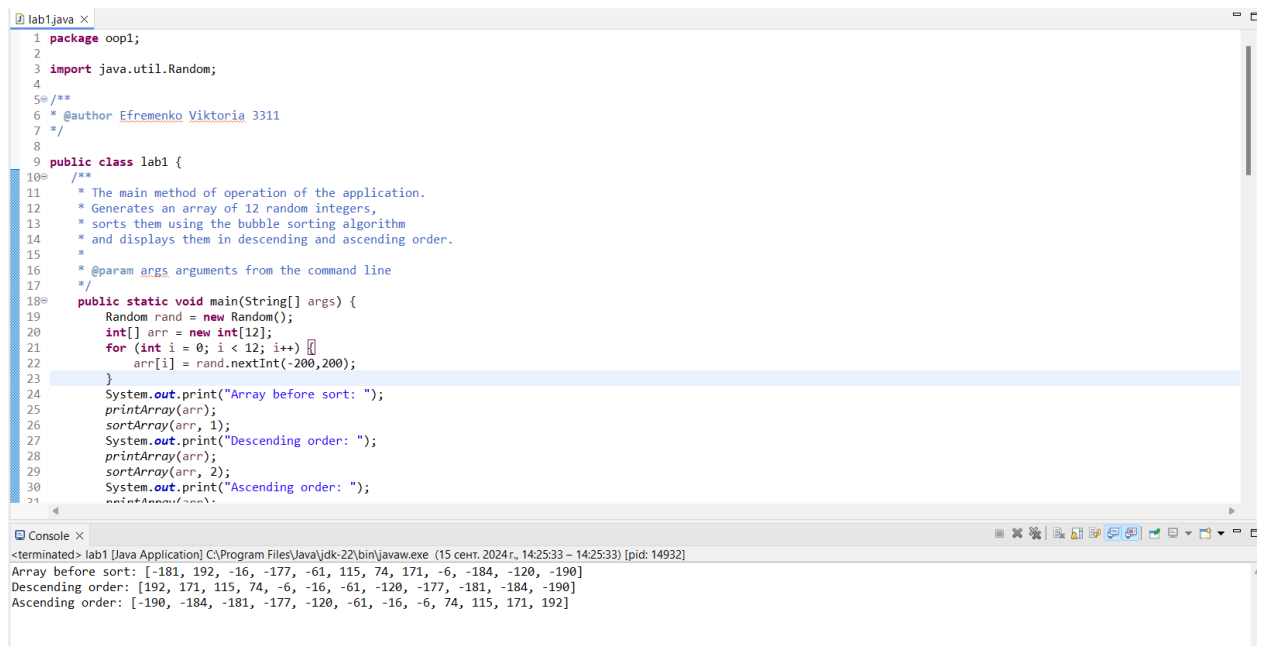
The screenshot shows the Eclipse IDE with a Java file named `lab1.java` open. The code defines a `main` method that generates an array of 12 random integers, prints it, sorts it in descending order, prints it again, sorts it in ascending order, and prints it a third time. A helper method `printArray` is also shown. The console output at the bottom displays the results of these operations.

```
16  * @param args arguments from the command line
17  */
18  public static void main(String[] args) {
19      Random rand = new Random();
20      int[] arr = new int[12];
21      for (int i = 0; i < 12; i++) {
22          arr[i] = rand.nextInt(-200,200);
23      }
24      System.out.print("Array before sort: ");
25      printArray(arr);
26      sortArray(arr, 1);
27      System.out.print("Descending order: ");
28      printArray(arr);
29      sortArray(arr, 2);
30      System.out.print("Ascending order: ");
31      printArray(arr);
32  }
33  /**
34   * Displays an array
35   *
36   * @param arr the array of integers
37   */
38  private static void printArray(int[] arr) {
39      System.out.print("[");
40      for (int i = 0; i < arr.length; i++) {
41          System.out.print(arr[i]);
42          if (i < arr.length - 1) {
43              System.out.print(", ");
44          }
45      }
46      System.out.print("]");
47  }
```

Console Output:

```
<terminated> lab1 [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (15 сент. 2024г., 14:24:35 – 14:24:38) [pid: 9772]
Array before sort: [-168, 30, -194, 116, 60, 149, 57, 191, 147, -181, 162, 77]
Descending order: [191, 162, 149, 147, 116, 77, 60, 57, 30, -168, -181, -194]
Ascending order: [-194, -181, -168, 30, 57, 60, 77, 116, 147, 149, 162, 191]
```

Пример 2



The screenshot shows an IDE window titled 'lab1.java'. The code defines a package 'oop1', imports 'java.util.Random', and includes a class 'lab1' with a 'main' method. The 'main' method generates an array of 12 random integers, sorts them using bubble sort, and prints the array in descending and ascending order. The console output shows the array before sorting, the descending order, and the ascending order.

```
1 package oop1;
2
3 import java.util.Random;
4
5 /**
6  * @author Efremenko Viktoria 3311
7  */
8
9 public class lab1 {
10     /**
11      * The main method of operation of the application.
12      * Generates an array of 12 random integers,
13      * sorts them using the bubble sorting algorithm
14      * and displays them in descending and ascending order.
15      */
16     @param args arguments from the command line
17     */
18     public static void main(String[] args) {
19         Random rand = new Random();
20         int[] arr = new int[12];
21         for (int i = 0; i < 12; i++) {
22             arr[i] = rand.nextInt(-200,200);
23         }
24         System.out.print("Array before sort: ");
25         printArray(arr);
26         sortArray(arr, 1);
27         System.out.print("Descending order: ");
28         printArray(arr);
29         sortArray(arr, 2);
30         System.out.print("Ascending order: ");
31         printArray(arr);
32     }
33 }
```

Console output:

```
<terminated> lab1 [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (15 сент. 2024 г., 14:25:33 - 14:25:33) [pid: 14932]
Array before sort: [-181, 192, -16, -177, -61, 115, 74, 171, -6, -184, -120, -190]
Descending order: [192, 171, 115, 74, -6, -16, -61, -120, -177, -181, -184, -190]
Ascending order: [-190, -184, -181, -177, -120, -61, -16, -6, 74, 115, 171, 192]
```

Ссылка на репозиторий

<file:///C:/java/oop1/doc/oop1/oop1/lab1.html> - javaDoc

https://disk.yandex.ru/d/aLVMT8_OMcQ6rw - ссылка на Яндекс диск, где находится сам файл java, видео и javaDov

Текст программы

```
package oop1;

import java.util.Random;

/**
 * @author Efremenko Viktoria 3311
 */

public class lab1 {
    /**
     * The main method of operation of the application.
     * Generates an array of 12 random integers,
     * sorts them using the bubble sorting algorithm
     * and displays them in descending and ascending order.
     */
    @param args arguments from the command line
    */
    public static void main(String[] args) {
        Random rand = new Random();
        int[] arr = new int[12];
        for (int i = 0; i < 12; i++) {
            arr[i] = rand.nextInt(-200,200);
        }
        System.out.print("Array before sort: ");
        printArray(arr);
        sortArray(arr, 1);
        System.out.print("Descending order: ");
        printArray(arr);
        sortArray(arr, 2);
        System.out.print("Ascending order: ");
        printArray(arr);
    }
}
```

```

    }
    System.out.print("Array before sort: ");
    printArray(arr);
    sortArray(arr, 1);
    System.out.print("Descending order: ");
    printArray(arr);
    sortArray(arr, 2);
    System.out.print("Ascending order: ");
    printArray(arr);
}
/**
 * Displays an array
 *
 * @param arr the array of integers
 */
private static void printArray(int[] arr) {
    System.out.print("[");
    for (int i = 0; i < arr.length; i++) {
        System.out.print(arr[i]);
        if (i < arr.length - 1) {
            System.out.print(", ");
        }
    }
    System.out.println("]");
}
/**
 * Sorts an array of integers in descending and ascending order using a bubble
sorting algorithm.
 * The method compares adjacent elements and swaps them if they are in the
wrong
order.
 *
 * @param arr the array of integers to be sorted
 * @param choice type of sort
 */
private static void sortArray(int[] arr, int choice) {
    if (choice == 1) {
        sortDescending(arr);
    } else if (choice == 2) {
        sortAscending(arr);
    }
}
/**
 * Sort array in descending order
 *
 * @param arr the array of integers
 */
private static void sortDescending(int[] arr) {
    boolean swapped;
    for (int i = 0; i < arr.length - 1; i++) {
        swapped = false;
        for (int j = 0; j < arr.length - i - 1; j++) {
            if (arr[j] < arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
                swapped = true;
            }
        }
        if (!swapped) break;
    }
}
}

```

```

/**
 * Sort array in ascending order
 *
 * @param arr the array of integers
 */
private static void sortAscending(int[] arr) {
    boolean swapped;
    for (int i = 0; i < arr.length - 1; i++) {
        swapped = false;
        for (int j = 0; j < arr.length - i - 1; j++) {
            if (arr[j] > arr[j + 1]) {
                int temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
                swapped = true;
            }
        }
        if (!swapped) break;
    }
}

```