SCD-Lab

Lab#6

Name: Anas-Altaf

Roll.no: 22F-3639

Java Codes:

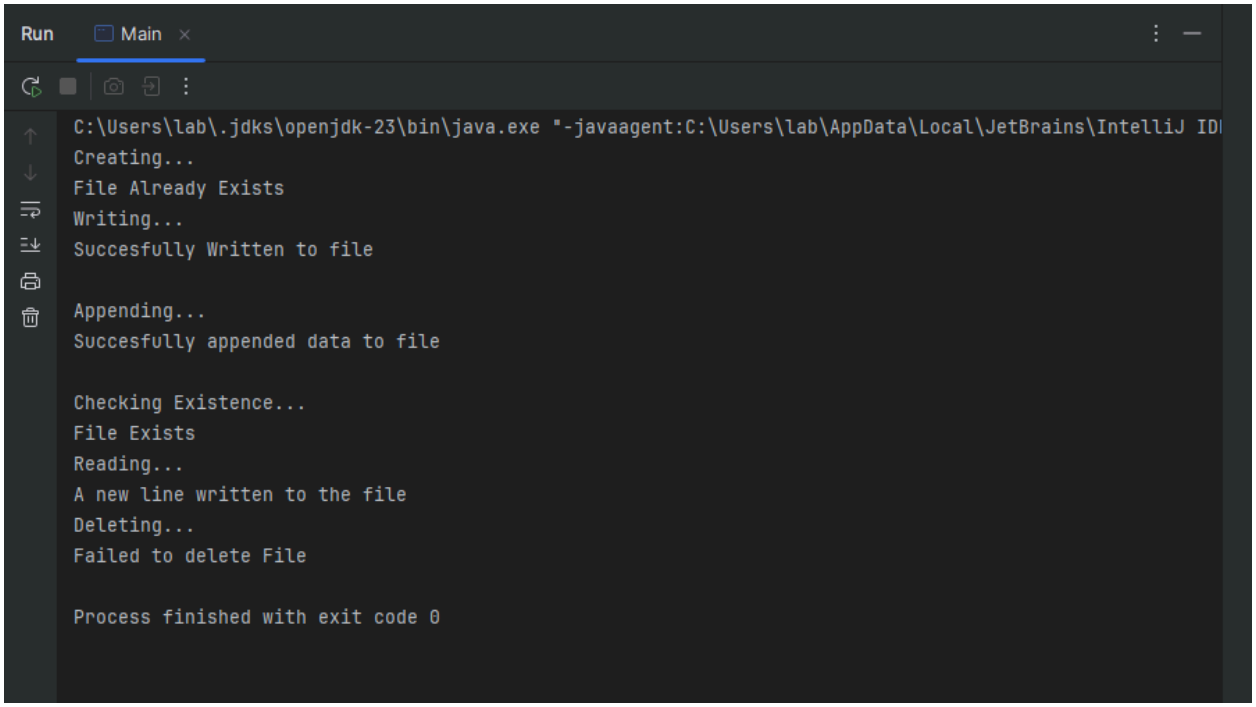
T-1:

```
package T1;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;

public class Main {
   static File f1 = new File("file1.txt");
   static FileWriter fileWriter;

   static {
       try {
           fileWriter = new FileWriter(f1);
       } catch (IOException e) {
           throw new RuntimeException(e);
       }
   }

   public static void createFile() throws IOException {
       File f1 = new File("file1.txt");
       if (f1.createNewFile()) {
           System.out.println("File Created : " + f1.getName());
       } else {
           System.out.println("File Already Exists");
       }
   }

   public static void writeToFile() throws IOException, NullPointerException {
```

```
        /// Writing to Fils
        fileWriter = new FileWriter(f1);

        fileWriter.write("A new line written to the file");
        fileWriter.close();
        System.out.println("Succesfully Written to file\n");
    }

    public static void appendToFile() throws IOException {
        /// Appendind Data to FIle
        fileWriter = new FileWriter(f1, true);

        fileWriter.write("\nA new line appended to the file");
        fileWriter.close();
        System.out.println("Succesfully appended data to file\n");
    }

    public static void checkIfExists() {
        /// Checking file existense
        if (f1.exists()) {
            System.out.println("File Exists");
        } else {
            System.out.println("File Does not exists");
        }
    }

    public static void deleteFile() {
        /// Deleting a File
        if (f1.delete()) {
            System.out.println("File deleted Successfully");
        } else {
            System.out.println("Failed to delete File");
        }

    }

    public static void readFromFile() throws FileNotFoundException {
        Scanner reader = new Scanner(f1);
        String fileData = reader.nextLine();
        System.out.println(fileData);
    }

    public static void main(String[] args) {
        try {
            System.out.println("Creating...");
            createFile();
            System.out.println("Writing...");
            writeToFile();
            System.out.println("Appending...");
```

```
            appendToFile();
            System.out.println("Checking Existence...");
            checkIfExists();
            System.out.println("Reading...");
            readFromFile();
            System.out.println("Deleting...");
            deleteFile();

        } catch (IOException e) {
            System.out.println(e.toString());

        } catch (Exception e) {
            System.out.println(e.toString());
        }
    }
}
```

Output:



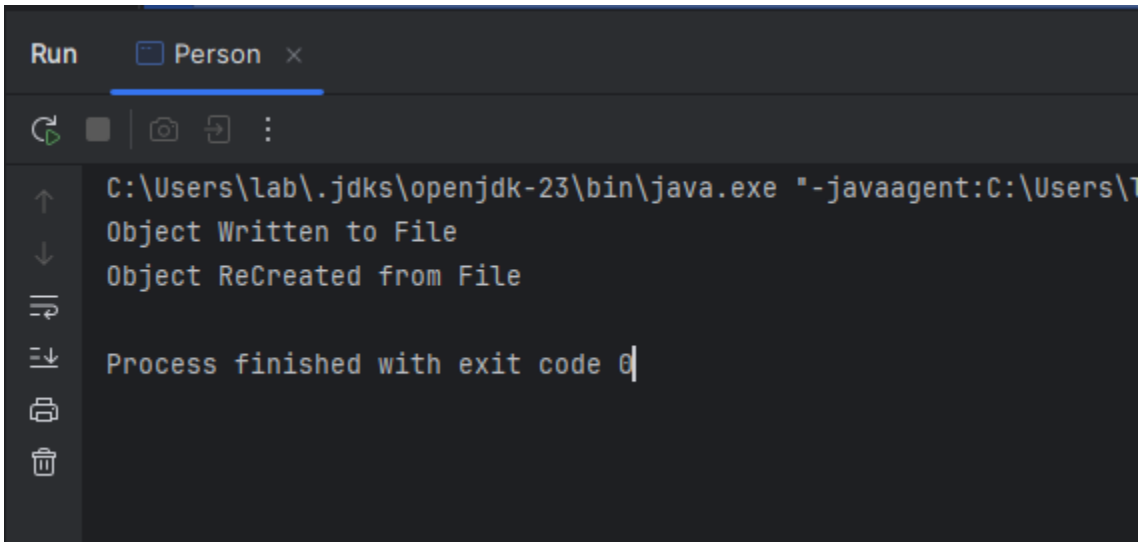
T-2:

```
package T2;

import java.io.*;

public class Person implements Serializable {
```

```
    private String name;
    private int age;

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public static void main(String[] s) {
        Person person1 = new Person("Anas", 20);
        Person person2 = new Person("Umair", 20);
        try {
            //Writing Objects
            FileOutputStream f = new FileOutputStream(new
File("src/T2/PersonObjects.txt"));
            ObjectOutputStream o = new ObjectOutputStream(f);
            o.writeObject(person1);
            o.writeObject(person2);
            System.out.println("Object Written to File");
            o.close();
            f.close();
            //Reading Objects
            FileInputStream fi = new FileInputStream(new
File("src/T2/PersonObjects.txt"));
            ObjectInputStream oi = new ObjectInputStream(fi);

            Person wrObject1 = (Person) oi.readObject();
            Person wrObject2 = (Person) oi.readObject();
            System.out.println("Object ReCreated from File");

        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        }


    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getAge() {
        return age;
```

```
    }

    public void setAge(int age) {
        this.age = age;
    }
}
```

Output:

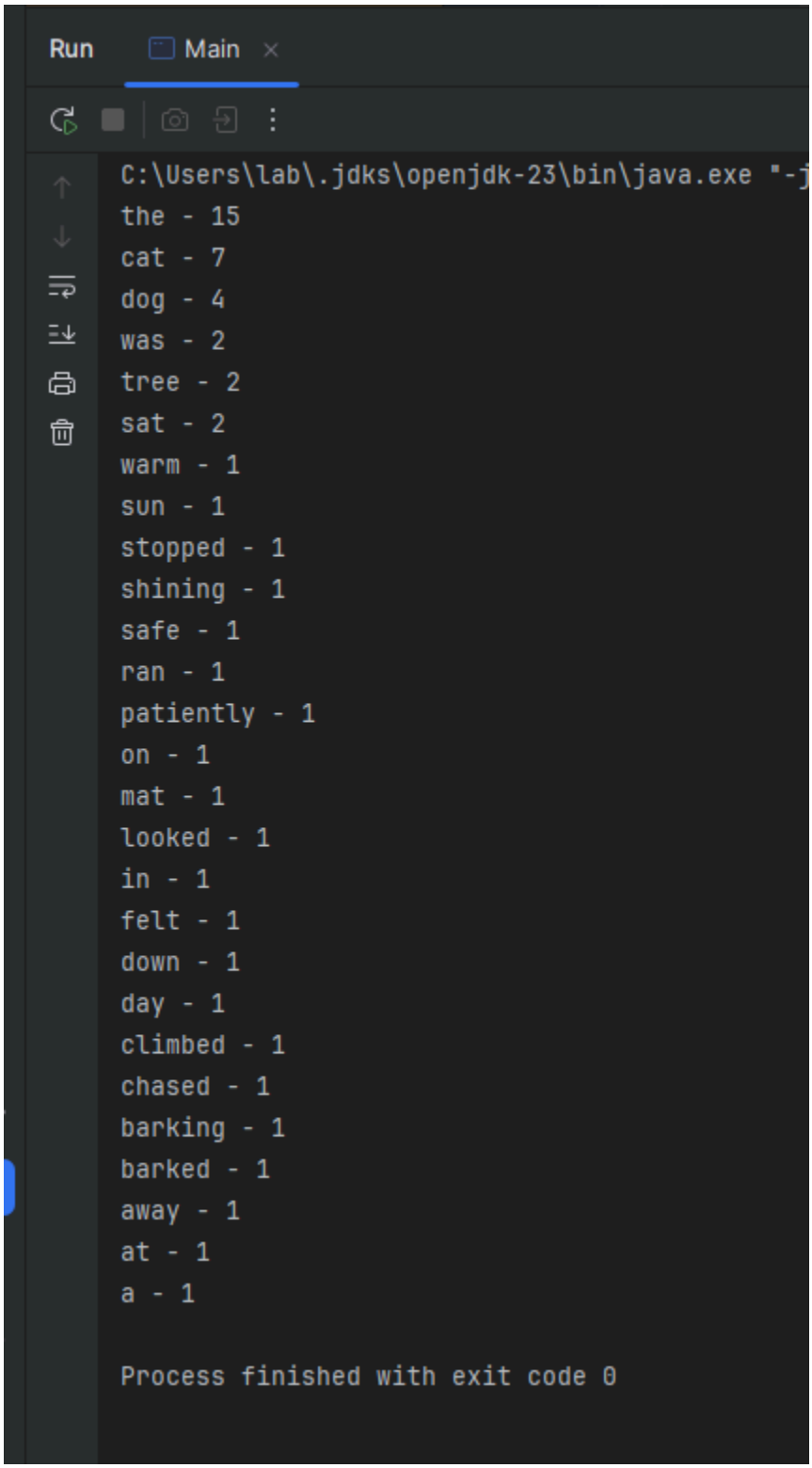


T-3:

```
package T3;

import java.io.File;
import java.io.FileNotFoundException;
import java.util.*;
import java.util.Map.Entry;
import java.util.stream.Collectors;

public class Main {
    private static Map<String, Integer> sortByValue(Map<String, Integer>
unsortMap, final boolean order) {
        List<Entry<String, Integer>> list = new
LinkedList<>(unsortMap.entrySet());

        // Sorting the list based on values
        list.sort((o1, o2) -> order ? o1.getValue().compareTo(o2.getValue()) ==
0
                ? o1.getKey().compareTo(o2.getKey())
                : o1.getValue().compareTo(o2.getValue()) :
o2.getValue().compareTo(o1.getValue()) == 0
```

```
                ? o2.getKey().compareTo(o1.getKey())
                : o2.getValue().compareTo(o1.getValue()));
        return list.stream().collect(Collectors.toMap(Entry::getKey,
Entry::getValue, (a, b) -> b, LinkedHashMap::new));

    }


    static void countFreq(String str) {
        Map<String, Integer> mp = new HashMap<>();


        String arr[] = str.split(" ");


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


            if (mp.containsKey(arr[i])) {
                mp.put(arr[i], mp.get(arr[i]) + 1);
            } else {
                mp.put(arr[i], 1);
            }
        }

        mp = sortByValue(mp, false);
        for (Map.Entry<String, Integer> entry :
                mp.entrySet()) {
            System.out.println(entry.getKey() +
                    " - " + entry.getValue());
        }
    }


    public static void main(String[] args) throws FileNotFoundException {
        File f1 = new File("src/T3/dataFile.txt");
        Scanner reader = new Scanner(f1);
        String fileData = reader.nextLine();

        countFreq(fileData);
    }


}
```

Output:


Run
Main
C:\Users\lab\.jdks\openjdk-23\bin\java.exe "-j
the - 15
cat - 7
dog - 4
was - 2
tree - 2
sat - 2
warm - 1
sun - 1
stopped - 1
shining - 1
safe - 1
ran - 1
patiently - 1
on - 1
mat - 1
looked - 1
in - 1
felt - 1
down - 1
day - 1
climbed - 1
chased - 1
barking - 1
barked - 1
away - 1
at - 1
a - 1
Process finished with exit code 0