LAB 7

File Input and Output

Name : Lee Pei Hui

Matric Number : 22004781/1

Occurrence : 7

Lecturer : Dr. Soraya

1. Write a program that store the table below in a binary file name coursename.dat. Then, ask the users to enter a course code. The program will display the course name from coursename.dat.

Course Code	Course Name
WXES1116	Programming I
WXES1115	Data Structure
WXES1110	Operating System
WXES1112	Computing Mathematics I

```
import java.io.*;
import java.net.URL;
import java.net.URLConnection;
import java.util.Scanner;

public class Lab7{
    static String[] code ={"WXES1116", "WXES1115", "WXES1110", "WXES1112"};
    static String[] name = {"Programming I", "Data Structure", "Operating System", "Computing
Mathematics I"};

    static boolean status = true;
    public static void main(String[] args) throws Exception{
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the course code: ");
        String input = sc.next();
        displayCourseName(input);
    }

    public static void storeTable(String[] args)throws Exception{
        /* store the table in a binary file name coursename.dat */
        FileOutputStream fileOutputStream = new FileOutputStream("coursename.dat");
        DataOutputStream dataOutputStream = new DataOutputStream(fileOutputStream);
        for(int i = 0; i<code.length; i++){
            dataOutputStream.writeUTF(code[i]);
            dataOutputStream.writeUTF(name[i]);
        }
        dataOutputStream.close();
    }
```

```
public static void displayCourseName(String input)throws Exception{
    FileInputStream fileInputStream = new FileInputStream("coursename.dat");
    DataInputStream dataInputStream = new DataInputStream(fileInputStream);
    while(true){
        if(input.equals(dataInputStream.readUTF())){
            System.out.println("Course Code: " + input);
            System.out.println("Course Name: " + dataInputStream.readUTF());
            status = true;
            break;
        }else{
            dataInputStream.readUTF();
        }
    }}
}
```

2. The code below is used to read the contents of a Web page. Write the contents of the Web page into a text file name index.htm.

```
import java.util.Scanner;
import java.net.URL;
import java.io.InputStream;
import java.net.URLConnection;
…
 try {
    URL u = new URL("http://www.fsktm.um.edu.my");
    URLConnection cnn = u.openConnection();
    InputStream stream = cnn.getInputStream();
    Scanner in = new Scanner(stream);

 }
 catch (IOException e) {
    System.out.println("IO Error:" + e.getMessage());
 }
```

```java
import java.io.*;
import java.net.URL;
import java.net.URLConnection;
import java.util.Scanner;

public class Lab7{
    public static void main(String[] args) {
        try {
                /*Write contents of Web page into a text file name index.htm*/
                PrintWriter out = new PrintWriter(new File("index.htm"));

                /*To read the contents of a Web page*/
                URL u = new URL("https://fsktm.um.edu.my");
                URLConnection cnn = u.openConnection();
                InputStream stream = cnn.getInputStream();
                Scanner in = new Scanner(stream);

                /*Write contents of Web page into a text file name index.htm*/
                while(in.hasNextLine()){
                    out.println(in.nextLine()); //While there is content read from the website,
write it to the text file
                }
                out.close();

                /*Exception*/
                }
                catch (IOException e) {
                System.out.println("IO Error:" + e.getMessage());
        }
    }
}
```

3. Write the statements that replace each line of a text file with its reverse and save it to a new text file name **reverse.txt**.

```
import java.io.*;
import java.lang.reflect.Array;
import java.util.Scanner;

public class Lab7{
    public static void main(String[] args) {
        try {
            Scanner inputStream = new Scanner(new FileInputStream("fileName"));
            PrintWriter outputStream = new PrintWriter(new FileOutputStream("reverse.txt"));
            while(inputStream.hasNextLine()){
                reverse(inputStream.nextLine());
            }
            outputStream.close();
        } catch (FileNotFoundException e) {

        }catch (IOException e){
            System.out.println("IO Error" + e.getMessage());
        }
    }
    public static void reverse(String s){
        char[] ori = s.toCharArray();
        char[] reversed = new char[s.length()];
        for(int i = 0; i<ori.length; i++){
            reversed[i] = ori[ori.length-1-i];
        }

        for (int j = 0; j<reversed.length; j++){
            System.out.println(reversed[j]);
        }
    }
}
```

4. Write a program that display the number of characters, words and lines in a text file. Assume that each word is separated by one space character.

```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

public class L7Q4 {

    public static void main(String[] args) {

        int lineCount = 0, wordCount = 0, charCount = 0;
        String line = "";

        try {
            BufferedReader file = new BufferedReader(new FileReader("./io_files/initial.txt"));

            line = file.readLine();

            while (line != null) {

                lineCount += 1;
                wordCount += line.split(" ").length;
                charCount += line.length();

                  line = file.readLine();
              }

              file.close();
          }
          catch (FileNotFoundException e) { System.out.println("File Not Found"); }
          catch (IOException e) { System.out.printf("IO Exception: %s\n", e); }

          System.out.printf("There are %d lines, %d words and %d characters in this file.\n",
 lineCount, wordCount, charCount);
      }
 }
```

5. Write a program that read data from a binary file **person.dat**. (Please download the binary file from the Web site.) Then, display the name, age and gender (M – Male, F – Female) in ascending order sort by name. The structure of the binary file is as below:

<intVariable-TotalNumberofRecord>
<StringVariable-Name><IntVariable-Age><CharVariable-Gender>

<StringVariable-Name><IntVariable-Age><CharVariable-Gender>

```
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.ObjectInputStream;

import java.util.Arrays;
import java.util.LinkedHashMap;

public class L7Q5 {

    public static void main(String[] args) {

        int recordCount = 0;

        try {

            ObjectInputStream file = new ObjectInputStream(new FileInputStream("./io_files/person.dat"));
            LinkedHashMap<String, Integer> data_1 = new LinkedHashMap<>();
            LinkedHashMap<String, Character> data_2 = new LinkedHashMap<>();

            recordCount = file.readInt();
           String[] names = new String[recordCount];

           for (int i = 0; i < recordCount; i++) {

               names[i] = file.readUTF();

               data_1.put(names[i], file.readInt());
               data_2.put(names[i], file.readChar());
           }

           Arrays.sort(names);

           for (String name: names) {
               System.out.printf("Name: %-12s, Age: %-2d, Gender: %1s\n", name, data_1.get(name), data_2.get(name));
           }

           file.close();
        }
        catch (FileNotFoundException e) { System.out.println("File Not Found"); }
        catch (IOException e) { System.out.printf("IO Exception: %s\n", e); }
    }
}
```

6. Write programs that merge data from two text files namely **product.txt** and **order.txt**. (Please download the text file from the Web site.) The **product.txt** contains the ProductID, ProductName, ProductPrice separated by comma. The **order.txt** contains the OrderID, ProductID, OrderQuantiy separated by comma. You program will display the ProductID, ProductName, OrderQuantity, PricePerUnit and TotalPrice for each order. (You can use the String.split(), Integer.parseInt(String) and Double.parseDouble(String))

ProductID	ProductName	Quantity	PricePerUnit	Total
SK079	Dettol Natural	20	4.99	99.80
SK013	100 Plus	74	6.49	480.26
SK088	Jasmine Pearl	27	37.99	1026.73
SK042	Ayamas Nuget Crispy	60	9.99	599.40
SK066	Ali Cafe	79	8.99	710.21

```
import java.io.BufferedReader;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.util.LinkedHashMap;

public class L7Q6 {

    public static void main(String[] args) {

        String line = "";
        String[] lineContent = new String[3];

        try {

            BufferedReader productFile = new BufferedReader(new FileReader("./io_files/product.txt"));
            BufferedReader orderFile = new BufferedReader(new FileReader("./io_files/order.txt"));

            LinkedHashMap<String, String> dataName = new LinkedHashMap<>();
            LinkedHashMap<String, Integer> dataQuantity = new LinkedHashMap<>();
            LinkedHashMap<String, Double> dataUnitPrice = new LinkedHashMap<>();
            LinkedHashMap<String, String> orderProduct = new LinkedHashMap<>();

            line = productFile.readLine();
```

```
            while (line != null) {

                lineContent = line.split(",");

                dataName.put(lineContent[0], lineContent[1]);
                dataUnitPrice.put(lineContent[0], Double.parseDouble(lineContent[2]));

                line = productFile.readLine();
            }

            line = orderFile.readLine();

            while (line != null) {

                lineContent = line.split(",");

                orderProduct.put(lineContent[0], lineContent[1]);
                dataQuantity.put(lineContent[1], Integer.parseInt(lineContent[2]));

                line = orderFile.readLine();
            }

            System.out.println("ProductID        ProductName        Quantity        PricePerUnit        Total");
            for (String order: orderProduct.keySet()) {
                System.out.printf("%-10s%-28s%2d%22.2f%15.2f\n",
                        orderProduct.get(order),
                        dataName.get(orderProduct.get(order)),
                        dataQuantity.get(orderProduct.get(order)),
                        dataUnitPrice.get(orderProduct.get(order)),
                        dataQuantity.get(orderProduct.get(order)) * dataUnitPrice.get(orderProduct.get(order))
                );
            }

            productFile.close();
            orderFile.close();
        }
        catch (FileNotFoundException e) { System.out.println("File Not Found"); }
        catch (IOException e) { System.out.printf("IO Exception: %s", e); }
    }
}
```

Link:

https://github.com/PeiHui369/Fundamentals-Of-Programming/tree/main/Lab%207