

COMPUTER PROGRAMING II- HOMEWORK 3

This homework is about Input/Output (File System). In following, there are different options about records.

```
-----MENU-----  
-----1-ADD RECORD-----  
-----2-REMOVE RECORD-----  
-----3-SEARCH RECORD-----  
-----4-LIST ALL-----  
-----5-DELETE ALL-----  
-----6-CLOSE-----
```

Please make a choice :

- If the user selects “1-ADD RECORD”, he/she specifies the name, price and number of the product/record (The name, price and number of product are asked to the user.). Then, the record is saved using method in the file (transaction.txt). After addition, the file (transaction.txt) is updated using method (update).

```
// addRecord(name,price,number)  
    // update();  
//System.out.println("Record is added.");
```

- If the user selects “2-REMOVE RECORD”, he/she specifies the name of the product that want to remove (The name of product are asked to the user.). otherwise “the record is not available”. After deletion, the file (transaction.txt) is updated using method (update).

```
// removeRecord(name);  
    // update();  
    o If there is a record with the specified name,then the record is removed using method  
      in the file (transaction.txt),  
      //System.out.println("Record is removed.");  
    o Otherwise  
      // System.out.println("Record is not available.");
```

- If the user selects “3-SEARCH RECORD”, he/she specifies the name of the product that want to search (The name of product are asked to the user.).

```
// searchRecord(name);  
    o If there is a record with the specified name,then the name, price and number of the  
      record are shown in console using method.  
      //For example; Burger 21.5 1  
    o Otherwise  
      // System.out.println("Record is not available.");
```

- If the user selects “4-LIST RECORD”, The name, price and number of all records/products must be listed using method.

```
// listRecord();
//For example; Burger 21.5 1
           Cola  15.0 3
```

- If the user selects “5-DELETE ALL”, If the user want to delete all records, then “Are you sure? y/n” is asked to the user. If he/she says yes, all records is deleted using method. After deletion, the file (transaction.txt) is updated using method (update).

```
// deleteall();
           // update();
// System.out.println("All records are deleted.");
```

- If the user selects “6-CLOSE”, System is existed.

-----CODES-----

```
public class IOApp
{
    static String filename="transactions.txt";
    static int limitRecord=100;
    static Record[] records;
    static int lastIndex;
    static class Record
    {
        String name;
        Integer price, number;
    }
}

public static void main(String[] args)
{
    initialProcess(); // Do not change

    // Write your codes here

}

private static void listRecord()
{
    // Write your codes here
}
```

```
private static void addRecord(String name, Integer price, Integer number)
{
    Write your codes here
}
```

```
private static void update()
{
    Write your codes here
}
```

```
private static boolean removeRecord(String name)
{
    Write your codes here
}
```

```
private static void searchRecord(String name)
{
    Write your codes here
}
```

```
private static void deleteall()
{
    // Your code is here
}
```

// initialProcess() method must not be changed.

```
private static void initialProcess()
{
    records=new Record[limitRecord];
    for (int i=0;i<limitRecord;i++)
    {
        records[i]=new Record();
    }
    try {
        Reader reader=new InputStreamReader(new FileInputStream(filename),"Windows-1254");
        BufferedReader br=new BufferedReader(reader);
        String strLine;
        int i=0;
        while((strLine=br.readLine())!=null)
        {
            StringTokenizer tokens=new StringTokenizer(strLine,"\\t");
            String [] t=new String[3];
            int j=0;
```

```
    while (tokens.hasMoreTokens())
    {
        t[j]=tokens.nextToken();
        j++;
    }
    records[i].name=t[0];
    records[i].price=Integer.valueOf(t[1]);
    records[i].number=Integer.valueOf(t[2]);
    i++;
}
lastIndex=i;
reader.close();
} catch (Exception e) {
    System.err.println("Error: "+e.getMessage());
}

}
}
```