Practical No.4

Name: Jadhav Siddhesh Ramesh
Class: T.Y.BSc(Comp. Science)
Roll No: 32
Div: A Batch: B
Subject :Object Oriented Programming Language Using Java-I
Assignment Name: Exception and File Handeling
Performance Date: Submission Date: / /

Set-A)

Q.1)Define a class patient (patient_name, patient_age patient_oxy_level,patient_HRCT_report).
Create an object of patient. Handle appropriate exception while patient oxygen level less than 95% and HRCT scan report greater than 10, then throw user defined Exception "Patient is Covid Positive(+) and Need to Hospitalized" otherwise display its information.

Program:

```
import java.util.*;
class patient
{
String name;
int age;
int oxylevel;
int HRCTreport;
patient(String name,int age,int oxylevel,int HRCTreport)
{
this.name=name;
this.age=age;
this.oxylevel=oxylevel;
this.HRCTreport=HRCTreport;
}
}
public class main2patient extends Exception
{
public static void main(String[] args)
{
Scanner sc=new Scanner(System.in);
System.out.println("How many patient you want to insert:");
int number =sc.nextInt();
patient[]ob=new patient[number];
for(int j=0;j<number;j++)
{
System.out.println("Enter Name");
String name=sc.next();
System.out.println("Enter age");
int age=sc.nextInt();
System.out.println("Enter oxylevel");
int oxylevel=sc.nextInt();
System.out.println("Enter HRCTreport");
int HRCTreport=sc.nextInt();
ob[j]=new patient(name,age,oxylevel,HRCTreport);
}
for(int j=0;j<number;j++)
{
if (ob[j].oxylevel<95 &&ob[j].HRCTreport >10)
try
{throw new NullPointerException("\n");
}
catch(Exception e)
{
System.out.println("patient is covide positive(+) and need toHospitalized");
}
else
{
System.out.println("Name:"+ob[j].name);
System.out.println("Age:"+ob[j].age);
System.out.println("oxylevel:"+ob[j].oxylevel);
System.out.println("HRCTreport:"+ob[j].HRCTreport);
System.out.println("\n");
}
}
}
}
```

OUTPUT:

```
How many patient you want to insert:
2
Enter Name
str
Enter age
18
Enter oxylevel
94
Enter HRCTreport
12
Enter Name
pqr
Enter age
20
Enter oxylevel
97
Enter HRCTreport
9
patient is covide positive(+) and need toHospitalized
Name:pqr
Age:20
oxylevel:97
HRCTreport:9
```

Q.2)Write a program to read a text file “sample.txt” and display the contents of a file in reverse
order and also original contents change the case (display in upper case).

Program:

```
import java.io.*;
import java.util.*;
class read
{
```

```
public static void main(String args[]) throws IOException
{
FileReader f=new FileReader("sample.txt");
Scanner sc=new Scanner(f);
String CH,CH2;
while(sc.hasNext()){
StringBuilder CH1=new StringBuilder();
CH=sc.next( );
CH2=CH.toUpperCase();
CH1.append(CH2);
CH1.reverse( );
System.out.println(CH1);
}
f.close( );
}
}
```

```
sampal.txt file
java
computer
```

OUTPUT:

```
AVAJ
RETUPMOC
```

Q.3)Accept the names of two files and copy the contents of the first to the second. First file having Book name and Author name in file. Second file having the contents of First file and also add the comment 'end of file' at the end.

Program:

```
import java.io.*;
import java.util.*;
class copy
{
public static void main(String arg[]) throws Exception
{
Scanner sc=new Scanner(System.in);
System.out.print("Source file name:\n");
String file1=sc.next();
System.out.print("Destination file name:\n");
String file2=sc.next();
FileReader fin=new FileReader(file1);
FileWriter fout=new FileWriter(file2,true);
int c;
while((c=fin.read())!=-1)
{
fout.write(c);
}
System.out.println("Copy finish...");
fin.close();
fout.close();
}
}
```

```
TEXT Files:
File1.txt:
java
```

```
sunmicrosystem
File2.txt:|
[empty]
OUTPUT:
Source file name:
File1.txt
Destination file name:File2.txt
Copy finish...
File2.txt:
[reload]
java
sunmicrosystem
```

Set-B)

Q.1)Define class EmailId with members ,username and password. Define default and parameterized constructors. Accept values from the command line Throw user defined exceptions – “InvalidUsernameException” or “InvalidPasswordException” if the username and password are invalid

Program:

```java
import java.util.*;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Scanner;
class EmailId
{
String domainID,username,password;
EmailId()
{
domainID="K7";
username="K";
password="1234";
}
EmailId(String user,String pass,String domID)
{
this.domainID=domID;
this.username=user;
this.password=pass;
}
public static void main(String[] args)
{
String user,pass, domID;
int p=-1,up=-1,u=-1,d=-1;
EmailId obj=new EmailId();
Scanner sc= new Scanner(System.in);
System.out.println("Enter username: ");
user=sc.nextLine();
System.out.println("\nEnter password: ");
pass=sc.nextLine();
System.out.println("\nEnter Domain ID: ");
domID=sc.nextLine();
System.out.println("Enter a date in dd/mm/yyyy format :");
String date = sc.next();
EmailId obj1=new EmailId(user,pass,domID);
```

```
if((obj.username).equals(obj1.username))
{
u=1;
if(obj.password.equals(obj1.password))
{
up=1;
}
else{
p=0;
}
}
else
{
u=0;
}
if(dateValidation(date)==true)
d=1;
else
d=0;
if(d==1)
{
if(u==0)
{
try
{
throw new InvalidUsernameException(user);
}
catch (Exception e) {
System.out.println(e) ;
}
}
if (p==0)
{
{
try
{
throw new InvalidPasswordException(pass);
}
catch (Exception e)
{
System.out.println(e) ;
}
}
}
if(up==1)
{
System.out.println("Valid email id");
}
}
else
{
System.out.println("Not a Valid email id");
}
}
public static boolean dateValidation(String date)
{
boolean status = false;
if (checkDate(date))
{
DateFormat dateFormat = new SimpleDateFormat("dd/MM/yyyy");
dateFormat.setLenient(false);
try{
dateFormat.parse(date);
status = true;
}
catch (Exception e)
{
status = false;
}
}
return status;
}
static boolean checkDate(String date)
{
String pattern = "(0?[1-9]|[12][0-9]|3[01])\\/(0?[1-9]|1[0-2])\\/([0-9]{4})";
boolean flag = false;
if (date.matches(pattern))
{
flag = true;}
return flag;
}
}
class InvalidUsernameException extends Exception
{
String num1;
InvalidUsernameException(String num2)
{
num1=num2;
}
public String toString()
{
return ("exception in thread ‘main’ InvalidUsernameException: Username: "+ num1 + "doesn’t
match");
}
}
class InvalidPasswordException extends Exception
{
String num1;
InvalidPasswordException(String num2)
{
num1=num2;
}
public String toString()
{
```

```
return ("exception in thread ‘main’ InvalidPasswordException: Username: "+ num1 + "doesn’t match") ;
}
}
```

OUTPUT:

```
Enter username:
K
Enter password:
1234
Enter Domain ID:
K7
Enter a date in dd/mm/yyyy format :02/09/2025
Valid email id
Invalid Statment:
Enter username:
P
Enter password:
234
Enter Domain ID:
Pra
Enter a date in dd/mm/yyyy format :
03/09/2025
exception in thread ‘main’ InvalidUsernameException: Username: Pdoesn’t match
```

Q.2)Define a class MyDate (day, month, year) with methods to accept and display a MyDate object. Accept date as dd, mm, yyyy. Throw user defined exception “InvalidDateException” if the date is invalid.
Examples of invalid dates : 03 15 2019, 31 6 2000, 29 2 2021

Program:

```
import java .io.*;
class InvalidDateException extends Exception
{
}
class MyDate
{
int day,mon,yr;
void accept(int d,int m,int y)
{
day=d;
mon=m;
yr=y;
}
void display()
{
System.out.println("Date is valid : "+day+"/"+mon+"/"+yr);
}
}
class Datemain1
{
public static void main(String arg[]) throws Exception
{
System.out.println("Enter Date : dd mm yyyy ");
BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
```

```
int day=Integer.parseInt(br.readLine());
int mon=Integer.parseInt(br.readLine());
int yr=Integer.parseInt(br.readLine());
int flag=0;
try
{
if(mon<=0 || mon>12)
throw new InvalidDateException();
else
{
if(mon==1 || mon==3 || mon==5 || mon==7 || mon==8 || mon==10 || mon==12)
{
if(day>=1 && day <=31)flag=1;
else
throw new InvalidDateException();
}
else if (mon==2)
{
if(yr%4==0)
{
if(day>=1 && day<=29)
flag=1;
else throw new InvalidDateException();
}
else
{
if(day>=1 && day<=28)
flag=1;
else throw new InvalidDateException();
}
}
else
{
if(mon==4 || mon == 6 || mon== 9 || mon==11)
{
if(day>=1 && day <=30)
flag=1;
else throw new InvalidDateException();
}
}
}
if(flag== 1)
{
MyDate dt = new MyDate();
dt.accept(day,mon,yr);
dt.display();
}
}
catch (InvalidDateException mm)
{
System.out.println("Invalid Date");
}
```

```
}
}
```

OUTPUT:

```
Enter Date : dd mm yyyy
Enter Date : dd mm yyyy
29
29
2
2
2019
2024
Invalid Date
Date is valid : 29/2/2024
Enter Date : dd mm yyyy
31
1
2014
Date is valid : 31/1/2014
Enter Date : dd mm yyyy
31
6
2025
Invalid Date
```