Q38. In extension of Q15 of this file, please raise a user defined checked Exception, if less than 0 or more than 100 marks are entered for grade.

package javapracfile;

import java.util.Scanner;

class LessThan0Exception extends Exception{

@Override public String toString(){

return "GradesLessThan0Exception";

}

}

class GreaterThan100Exception extends Exception{

@Override public String toString(){

return "GradesGreaterThan0Exception";

}

}

public class Q38 {

private int marks;

public void getMarks()throws LessThan0Exception, GreaterThan100Exception{

System.out.print("Enter marks: ");

marks = new Scanner(System.in).nextInt();

if(marks < 0){

throw new LessThan0Exception();

}

else if(marks > 100){

throw new GreaterThan100Exception();

}

}

public void displayMarks(){

System.out.println("Marks: "+marks);

}

public static void main(String[] args) {

Q38 obj = new Q38();

try{

obj.getMarks();

obj.displayMarks();

}catch(LessThan0Exception | GreaterThan100Exception e){

System.out.println(e);

}

}

}

Q40. In q38 instead of handling it through try catch use throws to transfer the exception till JVM level is reached.

package javapracfile;

import java.util.Scanner;

public class Q40 {

private int marks;

public void getMarks()throws LessThan0Exception, GreaterThan100Exception{

System.out.print("Enter marks: ");

marks = new Scanner(System.in).nextInt();

if(marks < 0){

throw new LessThan0Exception();

}

else if(marks > 100){

throw new GreaterThan100Exception();

}

}

public void displayMarks(){

System.out.println("Marks: "+marks);

}

public static void main(String[] args)throws LessThan0Exception, GreaterThan100Exception {

Q40 obj = new Q40();

obj.getMarks();

obj.displayMarks();

}

}

Q41. Write a program to print details of main thread. Also change its name and priority.

package javapracfile;

public class Q41 {

public static void main(String a[]){

System.out.println(Thread.currentThread());

Thread t = Thread.currentThread();

t.setName("MyMain");

t.setPriority(Thread.MAX\_PRIORITY);

System.out.println(Thread.currentThread());

}

}

Q43. Write a java multithreaded (2 or more)java program , one thread will print odd number and another will print even numbers and main thread is there it will print date and time.

package javapracfile;

public class Q43 extends Thread{

Thread t1, t2;

Q43(){

t1 = new Thread("Child\_Thread\_1");

t2 = new Thread("Child\_Thread\_2");

this.Start();

}

final public void Start(){

t1.start();

t2.start();

}

@Override public void run(){

try{

Thread t1 = Thread.currentThread();

oddNums();

t1.sleep(1500);

Thread t2 = Thread.currentThread();

evenNums();

t2.sleep(1500);

}catch(InterruptedException e){

System.out.println(e);

}}

static void oddNums(){

for(int i=0; i<=10; i++){

if(i % 2 != 0)

System.out.println(i);}}

static void evenNums(){

for(int i=0; i<=10; i++){

if(i % 2 == 0)

System.out.println(i);

}

}

public static void main(String a[]){

Thread T = new Thread(new Q43());

T.start();}}

Q47. Write a program to accept string from user, reverse the string and save it in file. Read the file and display same string on screen it in exact format. (use character Stream) [Use of Scanner class is not allowed].

package javapracfile;

import java.io.\*;

public class Q46 {

private final BufferedInputStream br;

private final File f;

private final FileInputStream fin;

private final FileOutputStream fout;

public Q46() throws FileNotFoundException, IOException{

br = new BufferedInputStream(System.in);

f = new File("C:\\Users\\aakas\\Desktop\\file.txt");

this.CreateFile();

fin = new FileInputStream(f);

fout = new FileOutputStream(f);

}

final boolean CreateFile()throws IOException{

if(f.exists() == true){

return true;

}else{

f.createNewFile();

return false;

}

}

void fileOutput() throws IOException{

char str;

System.out.println("Enter Characters || Enter 'q' to quit!");

do{

str = (char)br.read();

fout.write(str);

}while(str != 'q');

f.setReadable(true);

}

void fileInput() throws IOException{

char ch[] = new char[(int)f.length()];

//if(f.canRead() == true){

for(int i=0; i<f.length()-1;i++){

ch[i] = (char)fin.read();

}

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

System.out.print(ch[i]);

}

//}

}

public static void main(String a[]) throws FileNotFoundException, IOException{

Q46 obj;

try{

obj = new Q46();

obj.fileOutput();

obj.fileInput();

}catch(Exception e){

System.out.println(e);

}

}

}

Q48. Write a java program to show IP address of your host machine and show all the IPaddress where www.google .com is hosted.

package javapracfile;

import java.net.InetAddress;

public class Q48 {

public static void main(String a[]){

try{

InetAddress ip[] = InetAddress.getAllByName("www.google.com");

System.out.println("Host Name: "+ip[0].getHostName());

for (InetAddress ip1 : ip) {

System.out.println("Host Address: "+ip1.getHostAddress());

}

InetAddress ip2 = InetAddress.getLocalHost();

System.out.println("Local Host: "+ip2.getHostName());

System.out.println("Local Host-Address: "+ip2.getHostAddress());

}catch(Exception e){

System.out.println(e);

}

}

}

Q50. Write a java program[Application] to create login window [UName[Label]:TextField, Password[Label]:Textfield, Save Button, Cancel Button]

/\*

<applet code="LoginWindow.class" height=200 width=300>

</applet>

\*/

package javapracfile;

import java.awt.\*;

import java.applet.\*;

public class LoginWindow extends Applet{

Button btn1, btn2;

Label l1, l2;

TextField t1, t2;

@Override public void init(){

l1 = new Label("UName");

add(l1);

t1 = new TextField(30);

add(t1);

l2 = new Label("Password");

add(l2);

t2 = new TextField(30);

add(t2);

btn1 = new Button("Save");

add(btn1);

btn2 = new Button("Cancel");

add(btn2);

}

}

Q52. Apply event handling on Q 52, On click of save button generate error message if any of the text box is blank.

package javapracfile;

import java.awt.event.\*;

import javax.swing.\*;

public class Q52 extends LoginWindow implements ActionListener{

@Override public void start(){

btn1.addActionListener(this);

}

@Override public void actionPerformed(ActionEvent ae){

if((t1.getText()).equals("")){

JOptionPane.showMessageDialog(btn1, "Please enter name!");

}

else if((t2.getText()).equals("")){

JOptionPane.showMessageDialog(btn1, "Please enter password!");

}

}

}

Q53. Write a java program to save details Database (SQL or Access).[Use Q50 for entering details]

package javapracfile;

import java.sql.\*;

import javax.swing.\*;

import java.awt.event.ActionEvent;

public class Q53 extends Q52{

Connection con;

PreparedStatement pstm;

String query = "insert into login values(?,?)";

@Override public void start(){

super.start();

}

void createConnection(){

try{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

con = DriverManager.getConnection("jdbc:odbc:mydb");

}catch(Exception e){

System.out.println(e);

}

}

void insertIntoDatabase(){

try{

this.createConnection();

pstm = con.prepareStatement(query);

pstm.setString(1, t1.getText());

pstm.setString(2, t2.getText());

pstm.executeUpdate();

pstm.close();

con.close();

JOptionPane.showMessageDialog(this,"datainserted","Confirmation", JOptionPane.INFORMATION\_MESSAGE);

}catch(Exception e){

System.out.println(e);

JOptionPane.showMessageDialog(this,"Failed to Connect to Database","Error Connection", JOptionPane.ERROR\_MESSAGE);

System.exit(0);

}

}

@Override public void actionPerformed(ActionEvent ae){

super.actionPerformed(ae);

Object obj = ae.getSource();

if(obj == btn1){

this.insertIntoDatabase();

}

}

}