**Lab Exercise -2**

**JAVA Programming**

**Ashutosh Chandrakant Deshmukh**

**Roll No : 15**

**Division : Technocrats**

Q.1 Write a servlet program to accept online voter details for registration of voters. Assume suitable table structure.

**Ans –**

• **RegisterForm.html**

##### <!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*Register*>

Name:<input type=*"text"* name=*"name"*> <br> age:<input type=*"text"* name=*"age"*> <br> epic no:<input type=*"text"* name=*"epic"*> <br>

<input type=*"submit"* value=*"Register"*>

</form>

</body>

</html>

• **Register.java (Servlet)**

**import** java.io.IOException; **import** java.io.PrintWriter;

**import** javax.servlet.ServletException; **import** javax.servlet.annotation.WebServlet; **import** javax.servlet.http.HttpServlet; **import** javax.servlet.http.HttpServletRequest; **import** javax.servlet.http.HttpServletResponse;

**import** java.sql.Connection; **import** java.sql.DriverManager; **import** java.sql.SQLException; **import** java.sql.Statement;

/\*\*

* Servlet implementation class Register

\*/

@WebServlet("/Register")

**public** **class** Register **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

* **@see** HttpServlet#HttpServlet()

\*/

**public** Register() { **super**();

###### // **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** service(HttpServletRequest request, HttpServletResponse

response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub response.setContentType("text/html"); PrintWriter out = response.getWriter();

String name=request.getParameter("name");

String epic=request.getParameter("epic"); String age=request.getParameter("age");

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection con =

DriverManager.*getConnection*("jdbc:mysql://localhost:3306/voter", "root", "");

Statement stmt = con.createStatement();

String s = "insert into voter values('"+name+"','"+epic+"',"+age+")";

**int** i = stmt.executeUpdate(s);

out.println(i+ "Record Insrted");

con.close();

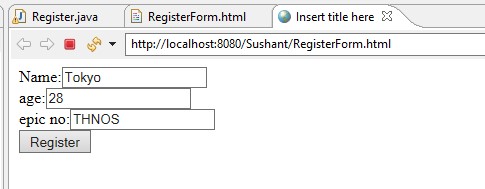
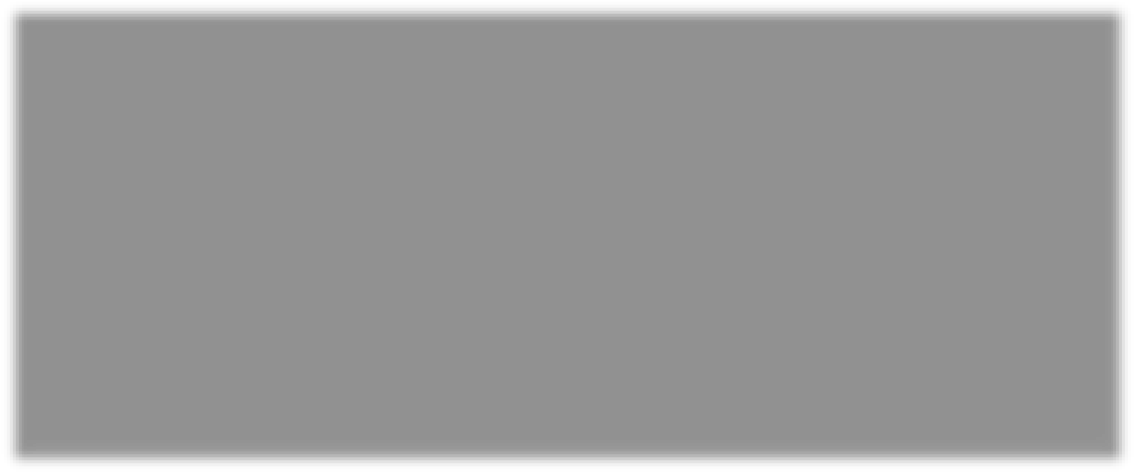
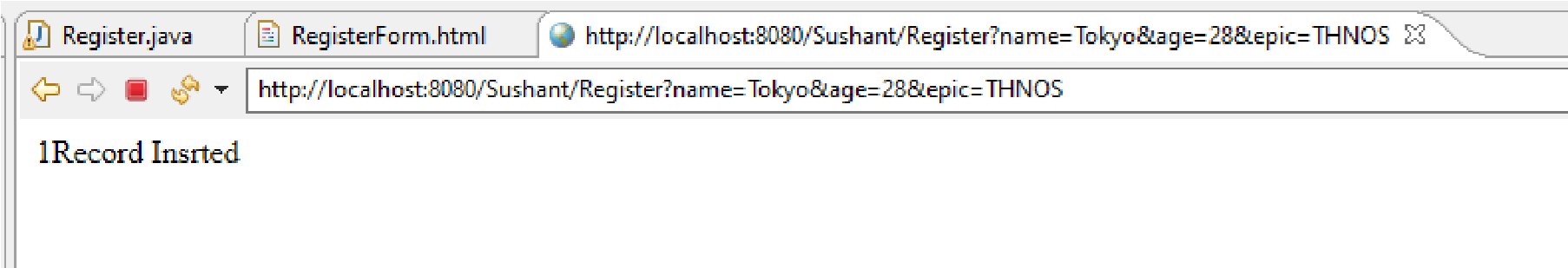
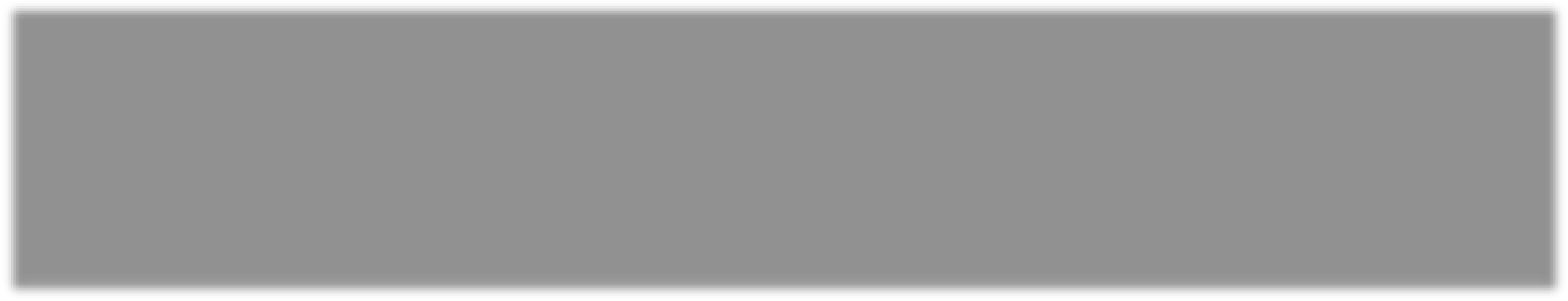
}

**catch**(Exception e) {out.println(e);}

}

}

**O/p-**



**Q.2 - Validate the data by using a servlet. Employee Information .**

**Validation parameters: a) All fields are compulsory. b) Emp-no should benumeric c) E-name should not contain special characters d) Salary should be numeric with two precision.**

## Ans –

## **•** EmployeeForm.html

##### <!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*EmployeeValidate*>

Employee Name:<input type=*"text"* name=*"name"* required> <br>

Employee Number:<input type=*"text"* name=*"enum"* required> <br>

Salary :<input type=*"text"* name=*"salary"* required> <br>

<input type=*"submit"* value=*"Validate"*>

</form>

</body>

</html>

## **•** EmployeeValidate.java

**import** java.io.IOException; **import** java.io.PrintWriter;

**import** javax.servlet.ServletException; **import** javax.servlet.annotation.WebServlet; **import** javax.servlet.http.HttpServlet; **import** javax.servlet.http.HttpServletRequest; **import** javax.servlet.http.HttpServletResponse;

**import** java.sql.Connection; **import** java.sql.DriverManager; **import** java.sql.SQLException; **import** java.sql.Statement;

/\*\*

* Servlet implementation class Register

\*/

@WebServlet("/Register")

**public** **class** Register **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

* **@see** HttpServlet#HttpServlet()

\*/

**public** Register() { **super**();

###### // **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** service(HttpServletRequest request, HttpServletResponse

response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub response.setContentType("text/html"); PrintWriter out = response.getWriter();

String name=request.getParameter("name");

String num=request.getParameter("enum"); String salary=request.getParameter("salary");

**int** Enum = 0;

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection con =

DriverManager.*getConnection*("jdbc:mysql://localhost:3306/voter", "root", "");

Statement stmt = con.createStatement();

**if** ((num == **null**) || (num.equals(""))) { out.println("PROVIDE Emp Num NUMBER...");

} **else** { **try** {

Enum = Integer.*parseInt*("num"); } **catch** (NumberFormatException nfe) {

out.println("PROVIDE int DATA IN Emp NUMBER...");

}

}

**if**((name == **null**) || (name.matches("^[a-zA-Z]\*$")) == **true**)

{

out.println("Name cannot be null or cannot include special char");

}

**if**((salary == **null**) || (salary.matches("^[0-9]\*\\.[0-9]{2}$ or ^[09]\*\\.[0-9][0-9]$")) == **false**)

{

|  |  |  |
| --- | --- | --- |
|  |  | out.println("Salary cannot be Null / Enter correct salary"); |
|  | } |  |
|  |  |  |

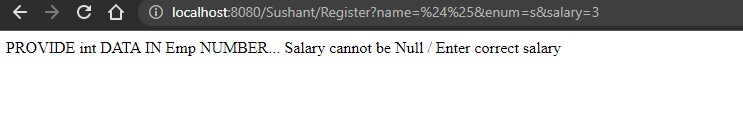
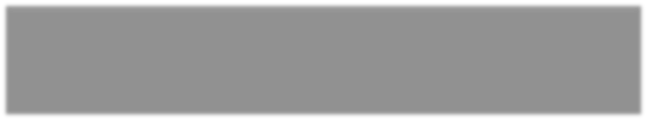
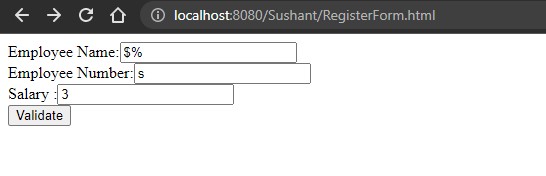
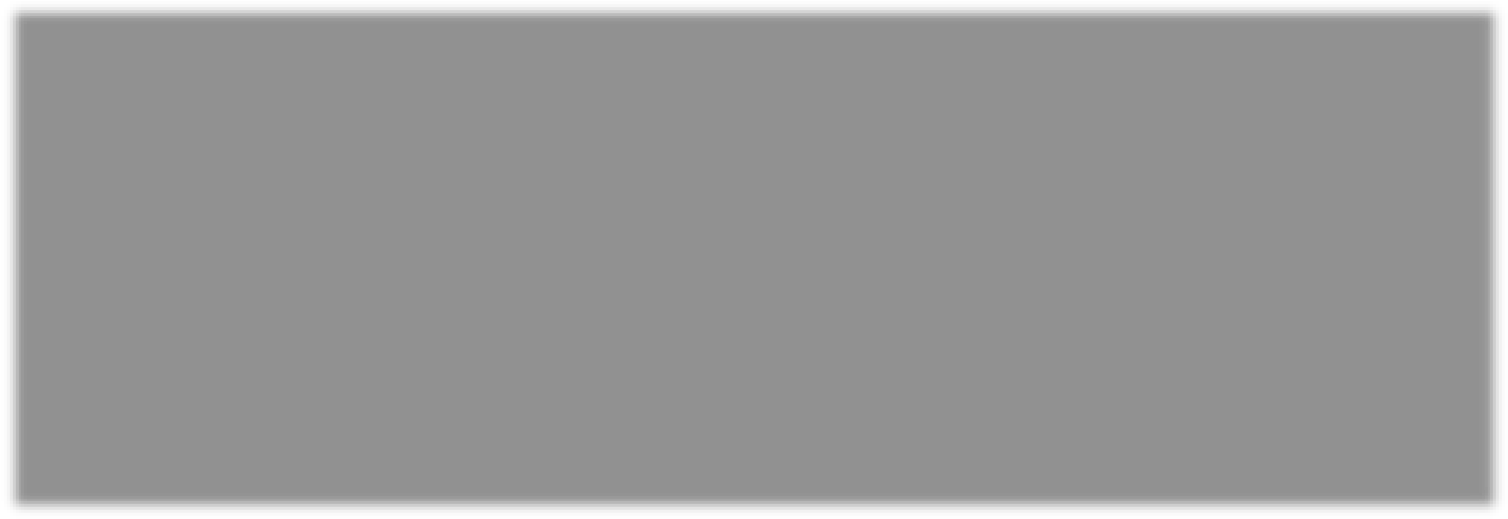
}

**catch**(Exception e) {out.println(e);}

}

}

## **O/p –**



**Q.3 - Write a servlet to check username & password passed from html page. If it is “Scott” & “tiger”, display welcome message else show the same html page again. [With res.sendRedirect (“http://localhost:8080/login.html”)]**

## Ans –

## **•** login.html

##### <!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*Register*>

Enter User Name:<input type=*"text"* name=*"username"* required> <br> password:<input type=*"text"* name=*"password"* required> <br>

<input type=*"submit"* value=*"Check"*>

</form>

</body>

</html>

## **•** Register.java

**import** java.io.IOException; **import** java.io.PrintWriter; **import** javax.servlet.ServletException; **import** javax.servlet.annotation.WebServlet; **import** javax.servlet.http.HttpServlet; **import** javax.servlet.http.HttpServletRequest; **import** javax.servlet.http.HttpServletResponse;

**import** java.sql.Connection; **import** java.sql.DriverManager; **import** java.sql.SQLException; **import** java.sql.Statement;

/\*\*

* Servlet implementation class Register

\*/

@WebServlet("/Register")

**public** **class** Register **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

* **@see** HttpServlet#HttpServlet()

\*/

**public** Register() { **super**();

###### // **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** service(HttpServletRequest request, HttpServletResponse

response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub response.setContentType("text/html"); PrintWriter out = response.getWriter();

String name=request.getParameter("username");

String pass=request.getParameter("password");

**if**(name.equals("scott") && pass.equals("tiger"))

{

out.println("<h1>Welcome</");

}

**else**

{

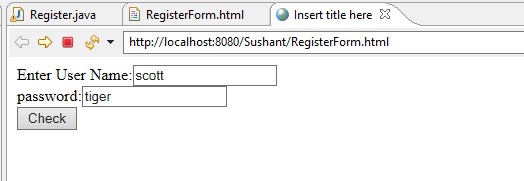
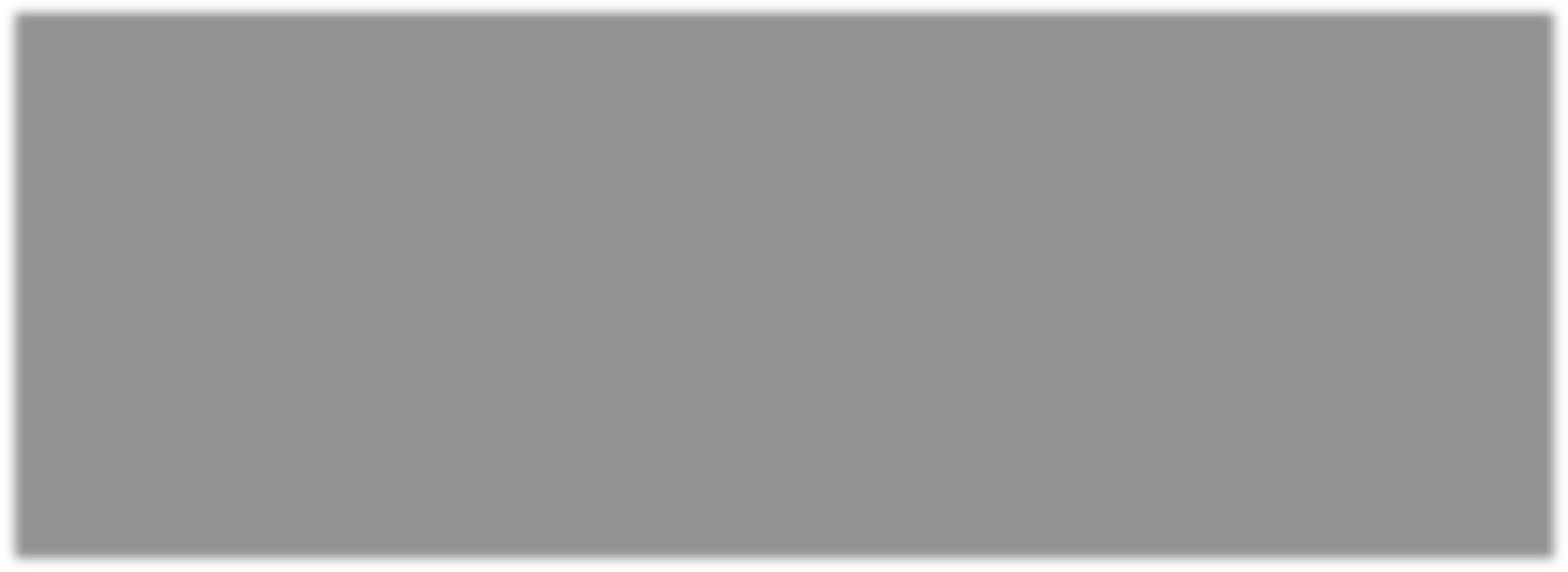
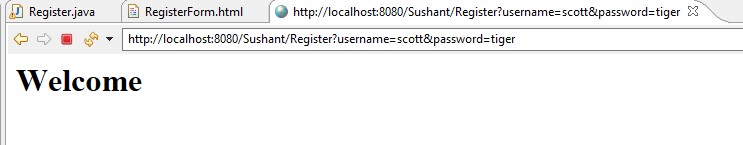
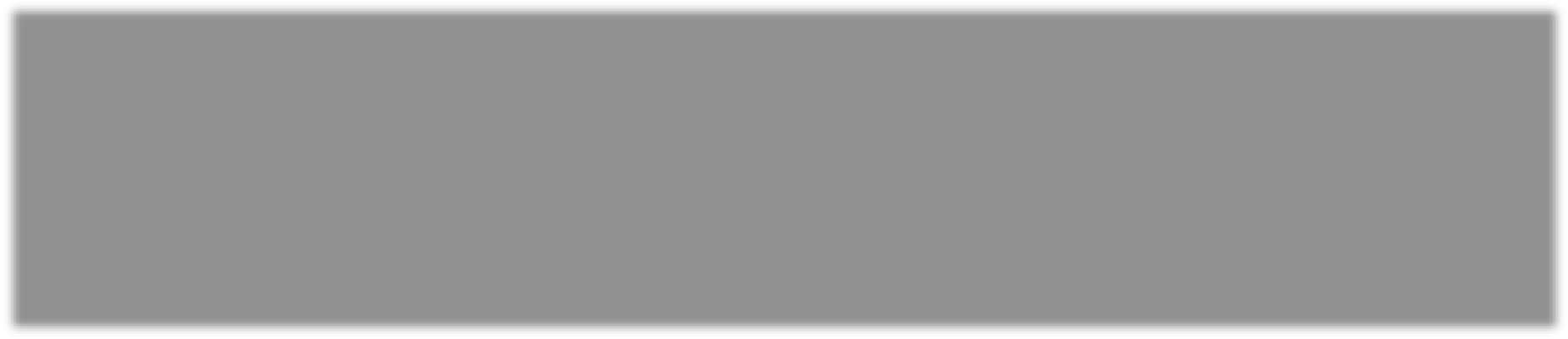
response.sendRedirect("http://localhost:8080/proj/RegisterForm.html");

}

}

}

## O/p –



**Q.4 - Create a menu driven program for Bank account(acc\_no, Name, amt) (Hint: use vector)**

**1. Add 2. Search 3. Delete 4. Display**

# Ans –

**import** java.io.\*; **import** java.util.Vector; **import** java.util.\*; **class** Vector1

{

**public** **static** **void** main(String args[])

{

**int** choice=0;

DataInputStream in=**new** DataInputStream(System.***in***);

Vector v = **new** Vector(); String s,name, acc\_no, amt; **try** { **do**

{

System.***out***.println("Select your choice:");

System.***out***.println("1 - Add");

System.***out***.println("2 - Delete");

System.***out***.println("3 - Display"); System.***out***.println("4 - Exit"); choice=Integer.*parseInt*(in.~~readLine~~()); **switch**(choice)

{ **case** 1:

System.***out***.println("Enter Account number:");

### ~~acc\_no=in.~~readLine~~();~~

System.***out***.println("Enter Name:"); name = in.~~readLine~~();

System.***out***.println("Enter Amount:"); amt = in.~~readLine~~();

v.addElement(acc\_no);

v.addElement(name);

v.addElement(amt);

System.***out***.println(name+" "+amt+" Added"); **break**;

**case** 2 : **if**(v.isEmpty())

System.***out***.println("list is empty"); **else**

{

System.***out***.println("Enter account :"); acc\_no=in.~~readLine~~(); **if**(v.contains(acc\_no))

{

v.removeElement(acc\_no);

System.***out***.println("Account name removed");

} **else**

System.***out***.println("Account does not exist");

} **break**;

**case** 3 : **if**(v.isEmpty())

System.***out***.println("list is empty"); **else**

System.***out***.println("Vector : "+v.toString()); **break**; **case** 4 : System.*exit*(0); **break**;

}

System.***out***.println("Do you want to continue? Press y for Yes or Press N for No");

### ~~s=in.~~readLine~~();~~

}**while**(s.equals("y"));

}

**catch**(Exception e)

{

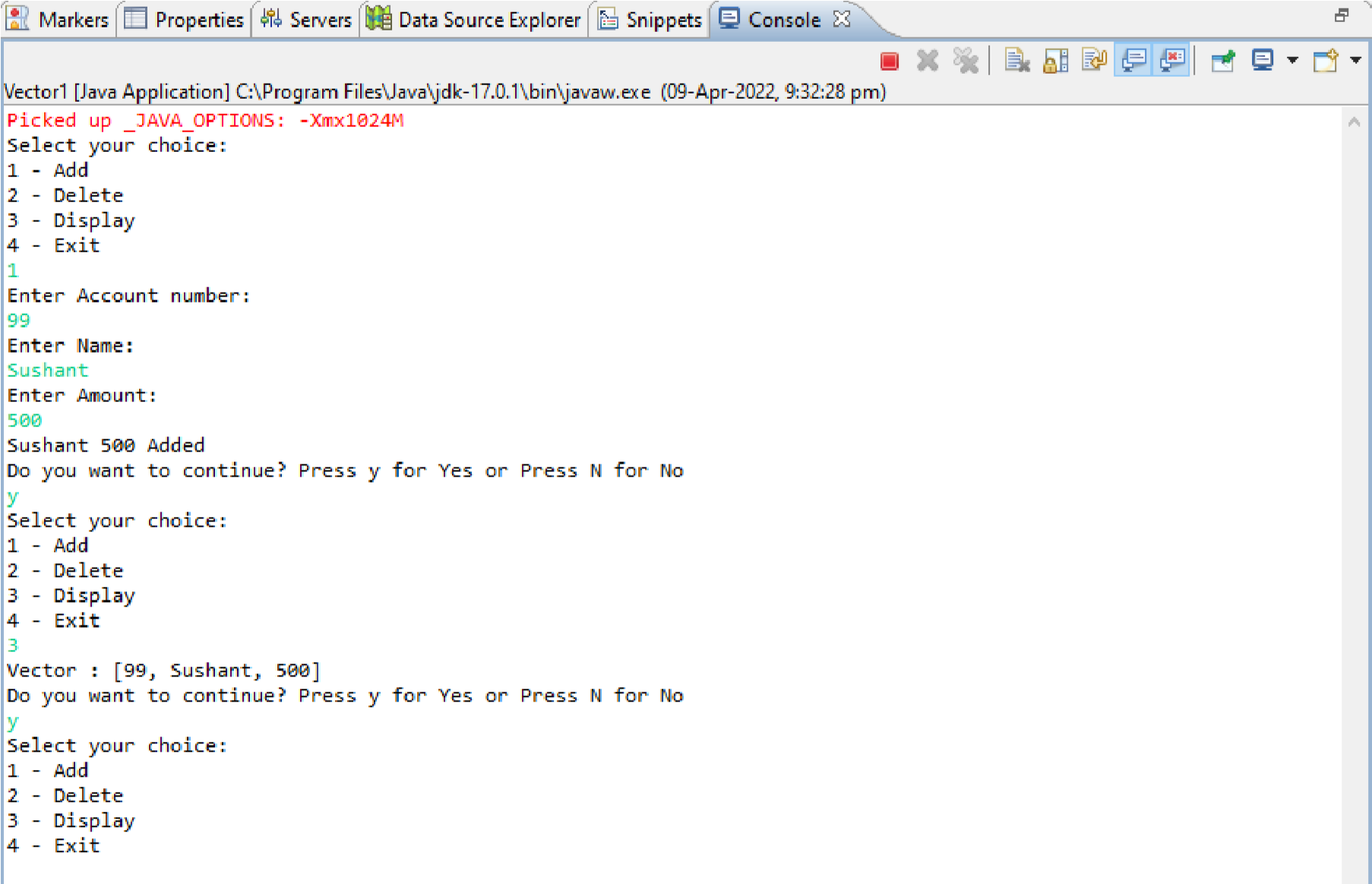
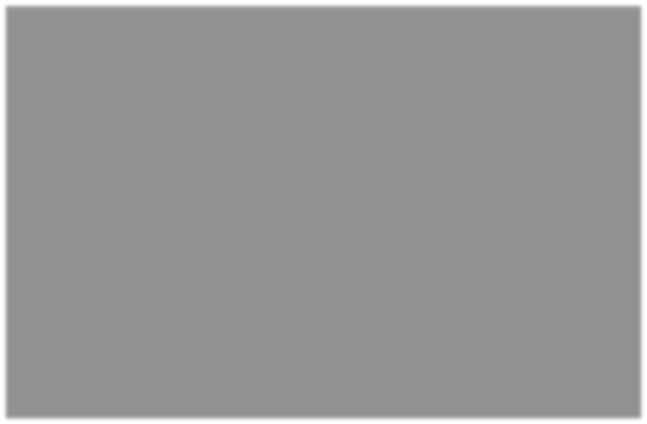
System.***out***.println("Exception caught:"+e);

}

}

}

# **O/P –**



**Q.5 - Accept Student names and marks as key-value of a treeMap and display them in ascending order of marks**

**Ans –**

**import** java.util.\*;

**class** Student **implements** Comparable<Student>

{

**public** Student(String name, **double** grade)

{

**this**.name = name; **this**.grade = grade;

}

String name; **double** grade;

#### @Override

**public** **int** compareTo(Student o)

{ **if** (o == **null**)

{

**return** -1;

}

**int** c = Double.*valueOf*(grade).compareTo(o.grade);

**if** (c != 0)

{ **return** c; }

**return** name.compareTo(o.name);

}

#### @Override

**public** String toString()

{

**return** String.*format*("%s has grade %.2f \n", name, grade);

}

**public** **static** **void** main(String[] args)

{

List<Student> al = **new** ArrayList<>(); al.add(**new** Student("Jadoo", 70)); al.add(**new** Student("Tokyo", 50)); al.add(**new** Student("Ayan", 82)); al.add(**new** Student("Ashutosh", 75)); al.add(**new** Student("Manali", 55));

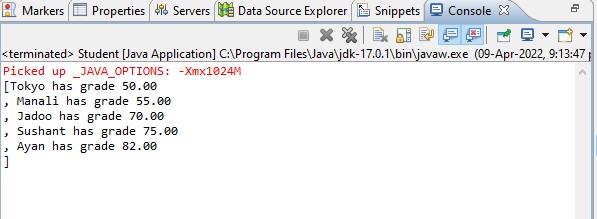
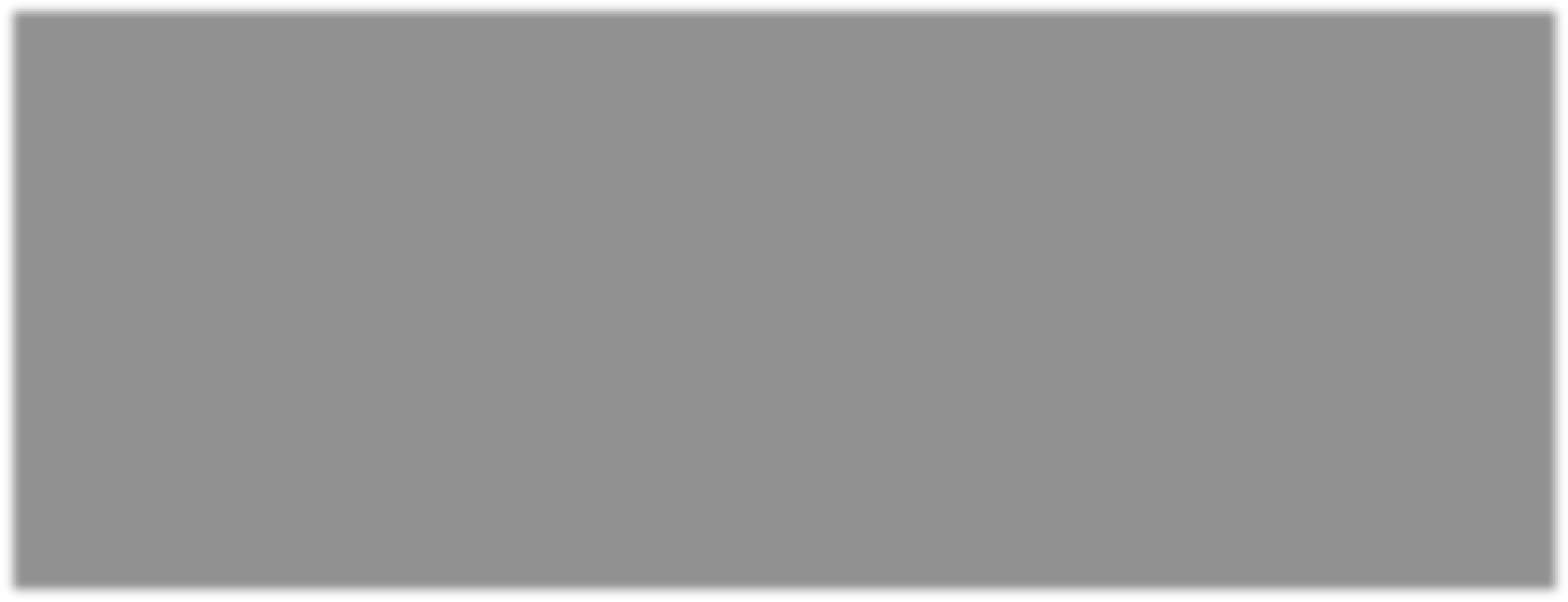
Collections.*sort*(al);

System.***out***.println(al);

}

}

**O/P -**



**Q.7 - Write a threaded application to display pyramid of stars. Accept number of stars from user**

# Ans –

**import** java.util.Scanner; **public** **class** ThreadEg **extends** Thread{

**public** **static** **int** *amount* = 0; **int** num = 0;

**public** **static** **void** main(String[] args) {

ThreadEg thread = **new** ThreadEg();

Scanner sc = **new** Scanner(System.***in***); System.***out***.println("Enter Number"); **int** num = sc.nextInt();

thread.run(num);

}

**public** **void** run(**int** num) { **for**(**int** i=0; i<num; i++)

{

**for**(**int** j=0; j<=i; j++)

{

System.***out***.print("\* ");

}

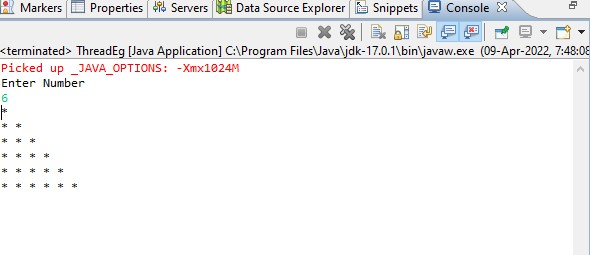
System.***out***.println();

}

}

}

## **O/P –**



**Q.8- Write a code to create calculator application, which will calculate simple Arithmetic operations. (Accept numbers from different text boxes and print result in textbox. Take 4 buttons for arithmetic operations)**

## Ans –

**import** java.awt.event.\*; **import** javax.swing.\*; **import** java.awt.\*;

**class** calculator **extends** JFrame **implements** ActionListener { **static** JFrame *f*; **static** JTextField *l*; String s0, s1, s2; calculator()

{

s0 = s1 = s2 = "";

}

**public** **static** **void** main(String args[])

{

*f* = **new** JFrame("calculator");

**try** {

UIManager.*setLookAndFeel*(UIManager.*getSystemLookAndFeelClassName*());

}

**catch** (Exception e) {

System.***err***.println(e.getMessage());

}

calculator c = **new** calculator(); *l* = **new** JTextField(16); *l*.setEditable(**false**);

JButton b0, b1, b2, b3, b4, b5, b6, b7, b8, b9, ba, bs, bd, bm, be, beq, beq1;

b0 = **new** JButton("0"); b1 = **new** JButton("1"); b2 = **new** JButton("2"); b3 = **new** JButton("3"); b4 = **new** JButton("4"); b5 = **new** JButton("5"); b6 = **new** JButton("6"); b7 = **new** JButton("7"); b8 = **new** JButton("8"); b9 = **new** JButton("9"); beq1 = **new** JButton("="); ba = **new** JButton("+"); bs = **new** JButton("-"); bd = **new** JButton("/"); bm = **new** JButton("\*");

beq = **new** JButton("C");

be = **new** JButton("."); JPanel p = **new** JPanel();

bm.addActionListener(c); bd.addActionListener(c); bs.addActionListener(c); ba.addActionListener(c); b9.addActionListener(c); b8.addActionListener(c); b7.addActionListener(c); b6.addActionListener(c); b5.addActionListener(c); b4.addActionListener(c); b3.addActionListener(c); b2.addActionListener(c); b1.addActionListener(c); b0.addActionListener(c); be.addActionListener(c); beq.addActionListener(c); beq1.addActionListener(c);

p.add(*l*);

p.add(ba);

p.add(b1);

p.add(b2);

|  |  |  |
| --- | --- | --- |
|  |  | p.add(b3); |
|  |  | p.add(bs); |
|  |  | p.add(b4); |
|  |  | p.add(b5); |
|  |  | p.add(b6); |
|  |  | p.add(bm); |
|  |  | p.add(b7); |
|  |  | p.add(b8); |
|  |  | p.add(b9); |
|  |  | p.add(bd); |
|  |  | p.add(be); |
|  |  | p.add(b0); |
|  |  | p.add(beq); |
|  |  | p.add(beq1); |
|  |  | p.setBackground(Color.***blue***); |
|  |  | *f*.add(p); |
|  |  | *f*.setSize(200, 220); |
|  |  | *f*.~~show~~(); |

}

**public** **void** actionPerformed(ActionEvent e)

{

String s = e.getActionCommand();

**if** ((s.charAt(0) >= '0' && s.charAt(0) <= '9') || s.charAt(0) ==

'.') {

**if** (!s1.equals("")) s2 = s2 + s; **else**

s0 = s0 + s;

*l*.setText(s0 + s1 + s2);

}

**else** **if** (s.charAt(0) == 'C') { s0 = s1 = s2 = "";

*l*.setText(s0 + s1 + s2);

}

**else** **if** (s.charAt(0) == '=') {

**double** te;

**if** (s1.equals("+"))

te = (Double.*parseDouble*(s0) + Double.*parseDouble*(s2)); **else** **if** (s1.equals("-"))

te = (Double.*parseDouble*(s0) - Double.*parseDouble*(s2)); **else** **if** (s1.equals("/"))

te = (Double.*parseDouble*(s0) / Double.*parseDouble*(s2));

**else**

te = (Double.*parseDouble*(s0) \* Double.*parseDouble*(s2));

*l*.setText(s0 + s1 + s2 + "=" + te); s0 = Double.*toString*(te);

s1 = s2 = "";

}

**else** {

**if** (s1.equals("") || s2.equals(""))

s1 = s; **else** { **double** te;

**if** (s1.equals("+"))

te = (Double.*parseDouble*(s0) +

Double.*parseDouble*(s2));

**else** **if** (s1.equals("-")) te = (Double.*parseDouble*(s0) -

Double.*parseDouble*(s2));

**else** **if** (s1.equals("/"))

te = (Double.*parseDouble*(s0) /

Double.*parseDouble*(s2));

**else**

te = (Double.*parseDouble*(s0) \*

Double.*parseDouble*(s2));

s0 = Double.*toString*(te);

s1 = s;

s2 = "";

}

*l*.setText(s0 + s1 + s2);

}

}

}

## **O/P –**

