NILGIRI COLLEGE OF ARTS AND SCIENCE

Accredited with A++ grade by NAAC

(Affiliated to Bharathiar University)

DEPARTMENT OF COMPUTER SCIENCE

JAVA PROGRAMMING - LAB

P R A C T I C A L R E C O R D

2023—2024

NAME . . . . . .

REGISTER No. . . . . . .

CLASS . . . . . .

SEMESTER . . . . . .

NILGIRI COLLEGE OF ARTS AND SCIENCE

Accredited with A++ grade by NAAC

(Affiliated to Bharathiar University)

DEPARTMENT OF COMPUTER SCIENCE

JAVA PROGRAMMING - LAB

P R A C T I C A L R E C O R D

NAME . . . . . **CLASS** . . . . . .

REGISTER No. . . . . . .

Certified that this is the bonafide record of work done by the above student of Bachelor of Computer Science in the Java Programming Laboratory during the year 2023- 2024.

Staff in-charge **Head of the Department** **Principal**

Submitted for the Practical Examination held on . . . . . .

Internal Examiner **External Examiner**

INDEX

Exp No: **Date:**	

EXTRACTING A STRING

```
import java.io.*;
class sample
{
public static void main(String args[])
{
String s=new String("Hello java");
System.out.println("\nstring extraction");
System.out.println("\n the given string is"+s);
System.out.println("The extracted string1 is:  "+s.substring(0,5));
System.out.println("The extracted string1 is:  "+s.substring(6));
}
}
```

Output:

string extraction

the given string is Hello java

The extracted string1 is: Hello

The extracted string1 is: java

Result:

Exp No: **Date:**	

MULTIPLE INHERITANCE USING INTERFACES

```
import java.io.*;
class student
{
int rollno;
void getnumber(int n)
{
rollno=n;
}
void  putnumbers()
{
System.out.println("roll no:"+rollno);
}
}
class test extends student
{
float part1,part2;
void getmarks(float m1,float m2)
{
part1=m1;
part2=m2;
}
void putmarks()
{
System.out.println("\t marks obtained");
System.out.println("part1="+part1);
```

```
System.out.println("part2="+part2);
}
}
interface sports
{
float sportwt=6.0f;
void putwt();
}
class result extends test implements sports
{
float total;
public void putwt()
{
System.out.println("spotwt="+sportwt);
}
void display()
{
total=part1+part2+sportwt;
putnumbers();
putmarks();
putwt();
System.out.println("Total score="+total);
}
}
class list1
{
public static void main(String args[])
```

```
{
result student1=new result();
student1.getnumber(123);
student1.getmarks(27.5f,33.0f);
student1.display();
}
}
```

Output:

roll no:123

marks obtained

part1=27.5

part2=33.0

spotwt=6.0

Total score=66.5

Result:

Exp No: **Date:**	

EXCEPTION

```
import java.lang.Exception;
class MyException extends Exception
{
MyException(String message)
{
super(message);
}
}
class list2
{
public static void main(String args[])
{
int x=5,y=1000;
try
{
float z=(float)x/(float)y;
if(z<0.01)
{
throw new MyException("number is too small");
}
}
catch(MyException e)
{
System.out.println("caught my exception");
System.out.println(e.getMessage());
```

```
}
finally
{
System.out.println("I am always here");
}
}
}
```

Output:

caught my exception

number is too small

I am always here

Result:

Exp No: **Date:**	

MULTITHREADING

```
import java.io.*;
class A extends Thread
{
public void run()
{
for(int i=1;i<=5;i++)
{
System.out.println(i + "*" + 5 + "="  + (i*5));
}
System.out.println("End of the First Thread");
}
}
class B extends Thread
{
public void run()
{
for(int j=1;j<=7;j++)
{
System.out.println(j + "*" + 7 + "="  + (j*7));
}
System.out.println("End of the Second Thread");
}
}
class C extends Thread
{
public void run()
```

```
{
for(int k=1;k<=13;k++)
{
System.out.println(k + "*" + 13 + "=" + (k*13));
}
System.out.println("End of the Third Thread");
}
}
public class Multithread
{
public static void main(String args[])
{
new A().start();
new B().start();
new C().start();
}
}
```

Output:

1*5=5

1*7=7

2*5=10

2*7=14

3*5=15

3*7=21

4*5=20

4*7=28

5*5=25

5*7=35

End of the First Thread

6*7=42

7*7=49

End of the Second Thread

1*13=13

2*13=26

3*13=39

4*13=52

5*13=65

6*13=78

7*13=91

8*13=104

9*13=117

10*13=130

11*13=143

12*13=156

13*13=169

End of the Third Thread

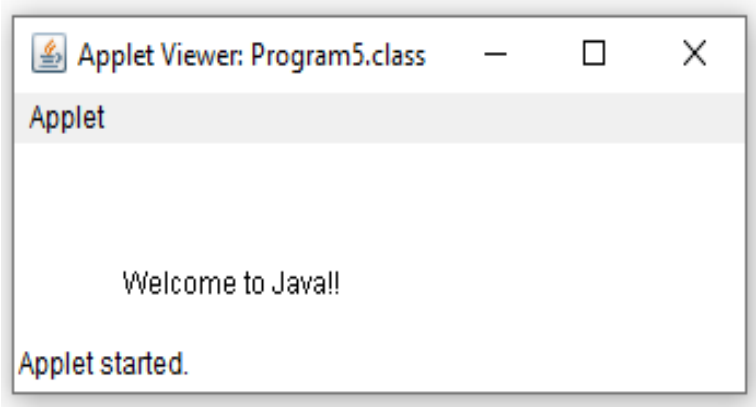
Result:

Exp No: **Date:**	

DIFFERENT SHAPES

```
import java.awt.*;
import java.applet.*;
import java.io.*;
public class Program5 extends Applet
{
public void paint(Graphics g)
{
g.drawLine(10,10,75,70);
g.drawRect(10,60,40,30);
g.drawRoundRect(10,100,80,50,10,10);
g.fillRoundRect(20,110,60,30,5,5);
g.drawOval(100,100,120,80);
g.fillOval(110,105,100,70);
}
}
```

Output:



Result:

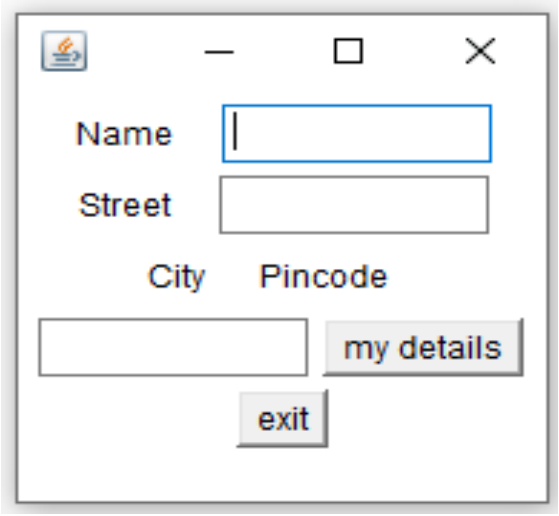
Exp No: **Date:**	

TO GET THE DETAILS BY CLICKING A BUTTON

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
class btn extends Frame implements ActionListener
{
Button b1,b2;
TextField t1,t2,t3,t4;
Label l1,l2,l3,l4;
btn()
{
b1=new Button("my details");
b2=new Button("exit");
b1.addActionListener(this);
b2.addActionListener(this);
l1=new Label("Nmae");
l2=new Label("Street");
l3=new Label("City");
l4=new Label("Pincode");
t1=new TextField(10);
t2=new TextField(10);
t3=new TextField(10);
t4=new TextField(10);
add(l1);
add(t1);
add(l2);
```

```
add(t2);
add(l3);
add(t4);
add(l4);
add(t4);
add(b1);
add(b2);
setLayout(new FlowLayout());
}
public static void main(String args[])
{
btn b=new btn();
b.setSize(200,200);
b.setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
if(ae.getSource()==b1)
{
t1.setText("ALIA");
t2.setText("Gandhi street");
t3.setText("Delhi");
t4.setText("60001");
}
else
{
System.exit(0);}}}
```

Output:



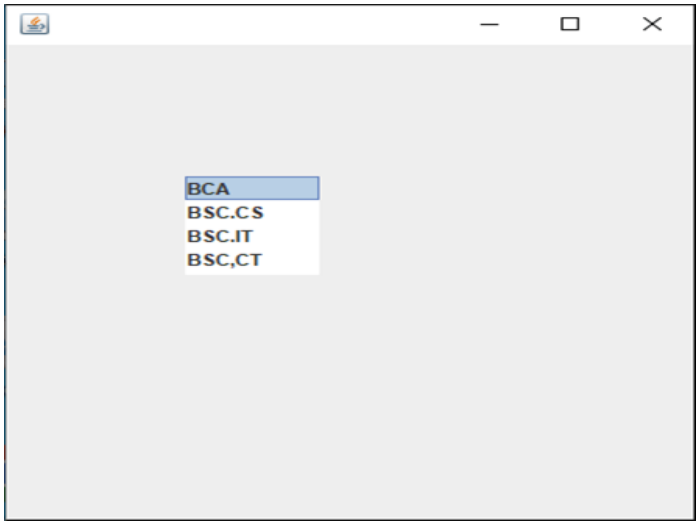
Result:

Exp No: **Date:**	

MULTIPLE SELECTION BOX

```
import javax.swing.*;
public class Listex
{
Listex()
{
JFrame f=new JFrame();
DefaultListModel l1=new  DefaultListModel();
l1.addElement("BCA");
l1.addElement("BSC.CS");
l1.addElement("BSC.IT");
l1.addElement("BSC,CT");
l1.addElement("BSC");
JList  list= new JList(l1);
list.setBounds(100,100,75,75);
f.add(list);
f.setSize(400,400);
f.setLayout(null);
f.setVisible(true);
}public static void main(String args[])
{new Listex();
}}
```

Output:



Result:

Exp No: **Date:**	

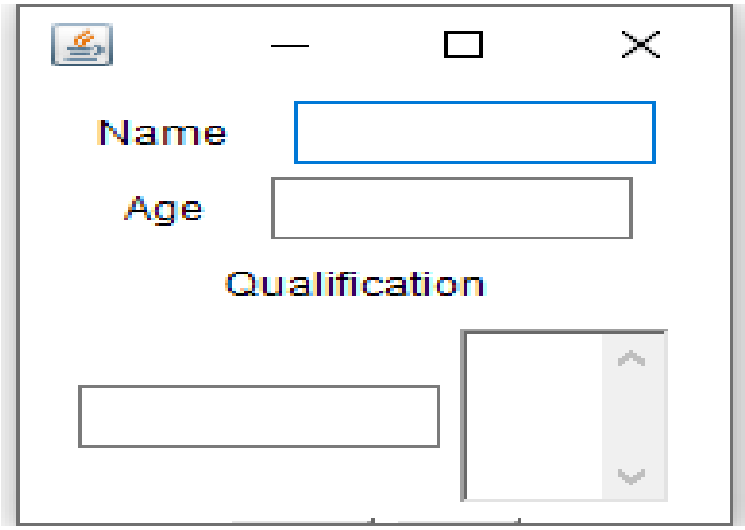
FRAMES

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
class btnnaction extends Frame implements ActionListener
{
Button b1,b2;
TextField t1,t2,t3;
TextArea ta;
Label l1,l2,l3,l4;
btnnaction()
{
b1=new Button("click");
b2=new Button("Exit");
b1.addActionListener(this);
b2.addActionListener(this);
t1=new TextField(10);
t2=new TextField(10);
t3=new TextField(10);
ta=new TextArea(3,5);
l1=new Label("Name");
l2=new Label("Age");
l3=new Label("Qualification");
l4=new Label("Address");
```

```
add(l1);
add(t1);
add(l2);
add(t2);
add(l3);
add(t3);
add(ta);
add(b1);
add(b2);
setLayout(new FlowLayout());
}
public static void main(String args[])
{
btnnaction b=new btnnaction();
b.setSize(200,200);
b.setVisible(true);
}
public void actionPerformed(ActionEvent ae)
{
if(ae.getSource()==b1)
{
t1.setText("Bindhu");
t2.setText("20");
t3.setText("Bsc.IT");
ta.setText("Charring cross,ooty");
```

```
}
else if(ae.getSource()==b2)
{
System.exit(0);
}
}
}
```

Output:



Result:

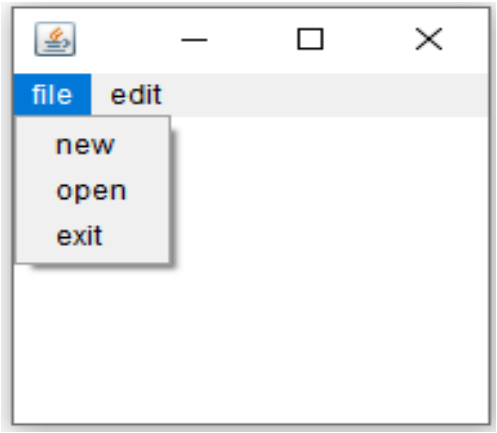
Exp No: **Date:**	

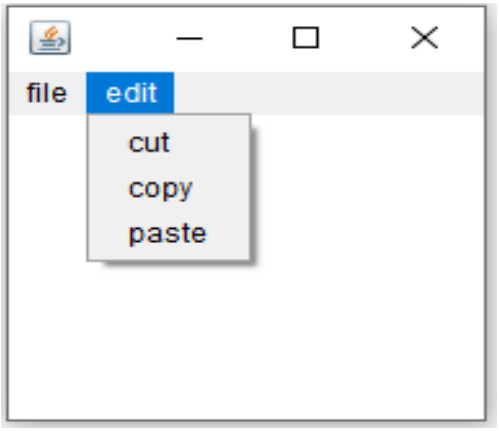
MENU EVENTS

```
import java.awt.*;
import java.applet.*;
class mbar extends Frame
{
MenuBar mb;
Menu mfile,medit;
MenuItem minew,miopen,miexit,micut,micopy,mipaste;
mbar()
{
mb=new MenuBar();
mfile=new Menu("file");
medit=new Menu("edit");
minew=new MenuItem("new");
miopen=new MenuItem("open");
miexit=new MenuItem("exit");
micut=new MenuItem("cut");
micopy=new MenuItem("copy");
mipaste=new MenuItem("paste");
mfile.add(minew);
mfile.add(miopen);
mfile.add(miexit);
medit.add(micut);
medit.add(micopy);
medit.add(mipaste);
```

```
mb.add(mfile);
mb.add(medit);
setMenuBar(mb);
}
public static void main(String args[])
{
mbar m=new mbar();
m.setSize(200,200);
m.setVisible(true);
}
}
```

Output:





Result:

Exp No: **Date:**	

MOUSE EVENTS

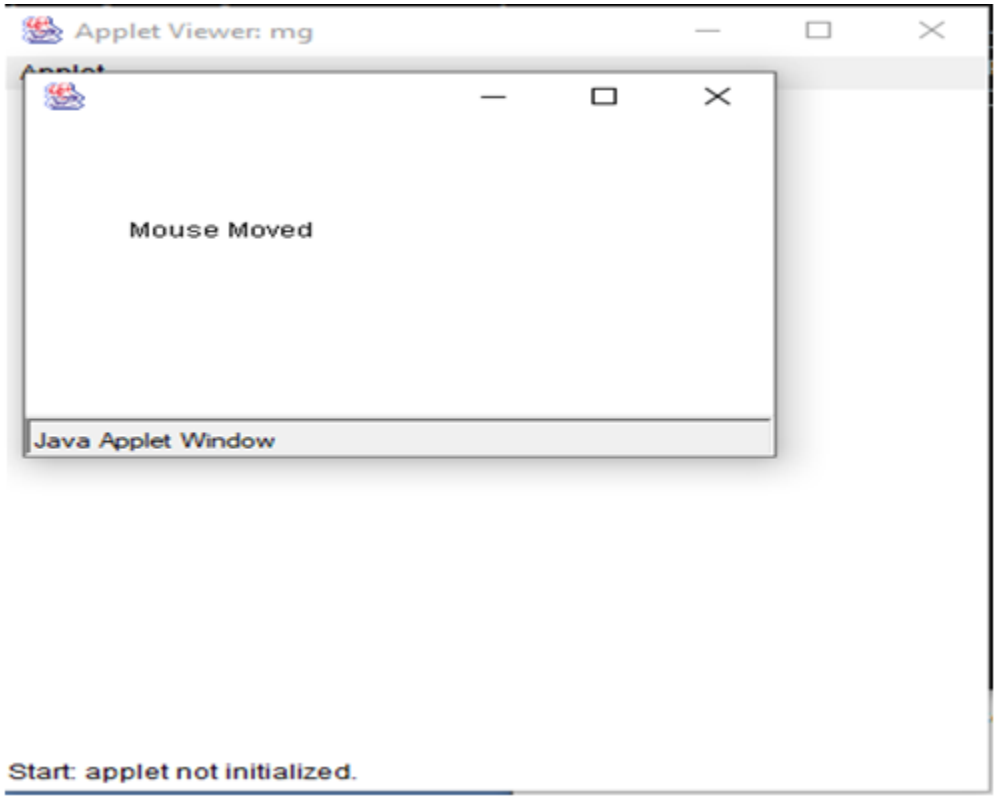
```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class mg extends Frame
implements MouseListener ,MouseMotionListener
{
String s=" ";
 public mg()
{
addMouseListener(this);
addMouseMotionListener(this);
setSize(320,240);
setVisible(true);
}
public void mouseClicked(MouseEvent e)
{
s="Mouse Clicked";
repaint();
}
public void mouseReleased(MouseEvent e)
{
s="Mouse Up";
repaint();
```

```
}
public void mouseEntered(MouseEvent e)
{
s="Mouse Entered";
repaint();
}
public void mouseDragged(MouseEvent e)
{
s="Mouse Dragged";
repaint();
}
public void mouseMoved(MouseEvent e)
{
s="Mouse Moved";
repaint();
}

public void mouseExited(MouseEvent e)
{
s="Mouse Exited";
repaint();
}
public void mousePressed(MouseEvent e)
{
s="Mouse Pressed";
```

```
repaint();
}
public void paint (Graphics g)
{
g.drawString(s,50,100);
}
}
<html>
<applet code="mg" width=400 height=400>
</applet>
</html>   // save the file” mg.html”
```

Output:



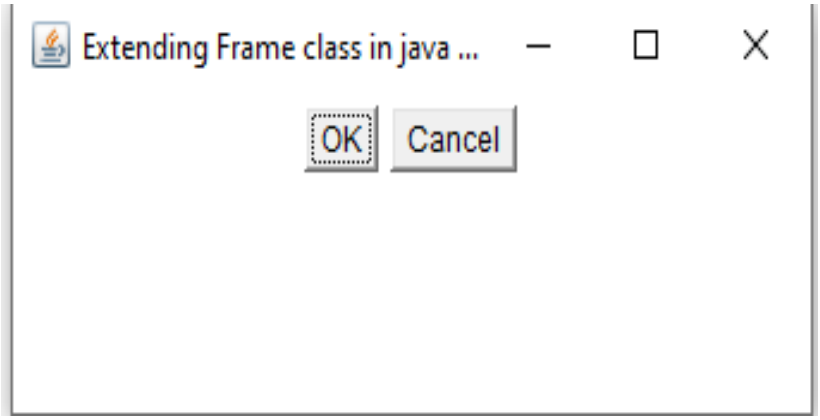
Result:

Exp No: **Date:**	

MOUSE CLICK POSITIONS

```
import java.awt.*;
class Myframe extends Frame
{
Myframe()
{
FlowLayout layout=new  FlowLayout();
setLayout(layout);
Button ok= new Button("OK");
Button cancel= new Button("Cancel");
add(ok);
add(cancel);
}
}
class ExtendingFrameClass
{
public static void main(String args[])
{
Myframe frame=new Myframe();
frame.setTitle("Extending Frame class in java Example");
frame.setSize(350,150);
frame.setVisible(true);
}}
```

Output:



Result:

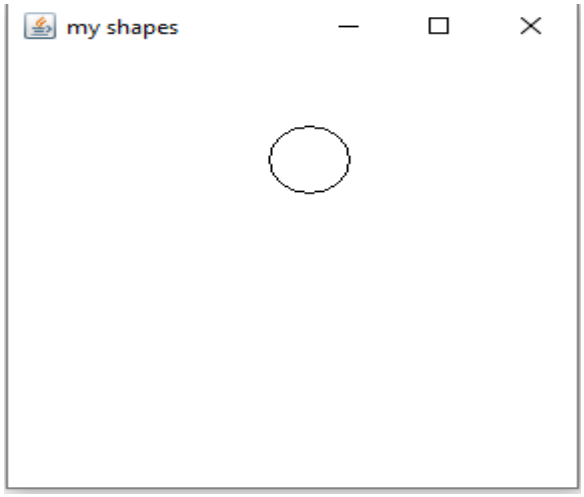
Exp No: **Date:**	

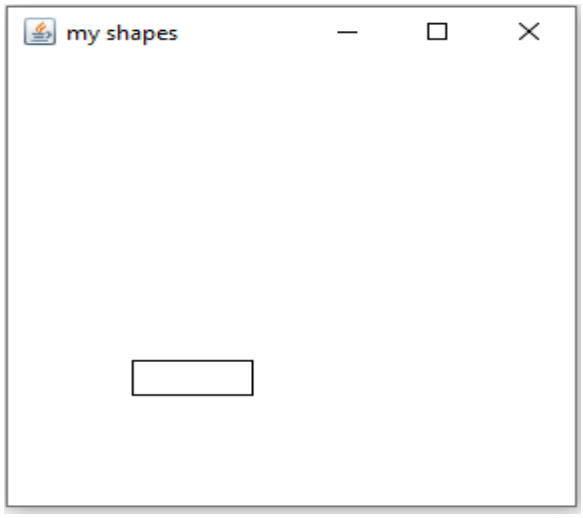
TO APPEND TEXT FILE

```
import java.io.*;
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
class fshp extends Frame implements MouseListener
{
String msg="";
int x=10,y=40;
fshp(String title)
{
super(title);
addMouseListener(this);
}
public void mouseClicked(MouseEvent ke){}
public void mouseDragged(MouseEvent ke){}
public void mouseExited(MouseEvent ke){}
public void mouseEntered(MouseEvent ke){}
public void mouseReleased(MouseEvent ke){}
public void mouseMoved(MouseEvent ke){}
public void mousePressed(MouseEvent ke)
{
x=ke.getX();
y=ke.getY();
msg="shapes";
```

```
repaint();
}
public void paint(Graphics g)
{
if((x>20)&&(x<50))
g.drawOval(x,y,25,30);
else if((x>=60)&&(x<90))
g.drawRect(x,y,60,20);
else if((x>120))
g.drawOval(x,y,40,40);
else if(x>10)
g.drawRect(x,y,20,20);
}
public static void main(String args[])
{
fshp f=new fshp("my shapes");
f.setSize(300,300);
f.setVisible(true);
}
}
```

Output:





Result: