Week 12

a. Develop a Java application to find the maximum value from the given type of elements using a generic function.

Source code:

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import java.applet.*;
import java.awt.event.*;
import java.awt.*;
public class Test extends Applet implements KeyListener
{
String msg="";
public void init()
{
 addKeyListener(this);
}
public void keyPressed(KeyEvent k)
{
 showStatus("KeyPressed");
}
public void keyReleased(KeyEvent k)
{
 showStatus("KeyRealesed");
}
public void keyTyped(KeyEvent k)
{
```

```
msg = msg+k.getKeyChar();
 repaint();
}
public void paint(Graphics g)
{
 g.drawString(msg, 20, 40);
 }
}
```

HTML code:

```
<applet code="Test" width=300, height=100>
</applet>
```

b. Develop a Java application that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to displaythe result.

Source code:

```
importjavax.swing.*;
import java.awt.*;
import
java.awt.event.*;
//<applet code=Calculator height=300
width=200></applet> public class Calculator extends
JApplet
{
public void init()
{
CalculatorPanel calc=new CalculatorPanel();
getContentPane().add(calc);
```

```
}
}
class CalculatorPanel extends JPanel implements ActionListener
{
JButton
n1,n2,n3,n4,n5,n6,n7,n8,n9,n0,plus,minus,mul,div,dot,equal;
static JTextField result=new JTextField("0",45);
static String lastCommand=null;
JOptionPane p=new
JOptionPane(); double
preRes=0,secVal=0,res; private
static void assign(String no)
{
if((result.getText()).equals("0")
) result.setText(no);
else if(lastCommand=="=")
{
result.setText(no);
lastCommand=null;
}
else
result.setText(result.getText()+no);
}
public CalculatorPanel()
{
setLayout(new BorderLayout());
```

```
result.setEditable(false);
result.setSize(300,200);
add(result,BorderLayout.NORTH);
JPanel panel=new JPanel();
panel.setLayout(new
GridLayout(4,4));
n7=new JButton("7");
panel.add(n7);
n7.addActionListener(this);
n8=new JButton("8");
panel.add(n8);
n8.addActionListener(this);
n9=new JButton("9");
panel.add(n9);
n9.addActionListener(this);
div=new JButton("/");
panel.add(div);
div.addActionListener(this);
n4=new JButton("4");
panel.add(n4);
n4.addActionListener(this);
n5=new JButton("5");
panel.add(n5);
n5.addActionListener(this);
n6=new JButton("6");
panel.add(n6);
```

```
n6.addActionListener(this);
mul=new JButton("*");
panel.add(mul);
mul.addActionListener(this);
n1=new JButton("1");
panel.add(n1);
n1.addActionListener(this);
n2=new JButton("2");
panel.add(n2);
n2.addActionListener(this);
n3=new JButton("3");
panel.add(n3);
n3.addActionListener(this);
minus=new JButton("-");
panel.add(minus);
minus.addActionListener(this);
dot=new JButton(".");
panel.add(dot);
dot.addActionListener(this);
n0=new JButton("0");
panel.add(n0);
n0.addActionListener(this);
equal=new JButton("=");
panel.add(equal);
equal.addActionListener(this);
plus=new JButton("+");
```

```
panel.add(plus);
plus.addActionListener(this);
add(panel,BorderLayout.CENTE
R);
}
public void actionPerformed(ActionEvent ae)
{
if(ae.getSource()==n1) assign("1");
else if(ae.getSource()==n2)
assign("2"); else
if(ae.getSource()==n3) assign("3");
else if(ae.getSource()==n4)
assign("4"); else
if(ae.getSource()==n5) assign("5");
else if(ae.getSource()==n6)
assign("6"); else
if(ae.getSource()==n7) assign("7");
else if(ae.getSource()==n8)
assign("8"); else
if(ae.getSource()==n9) assign("9");
else if(ae.getSource()==n0)
assign("0"); else
if(ae.getSource()==dot)
{
if(((result.getText()).indexOf("."))==-1)
result.setText(result.getText()+".");
```

```
}
else if(ae.getSource()==minus)
{
preRes=Double.parseDouble(result.getText
()); lastCommand="-";
result.setText("0");
}
else if(ae.getSource()==div)
{
preRes=Double.parseDouble(result.getText());
lastCommand="/";
result.setText("0");
}
else if(ae.getSource()==equal)
{
secVal=Double.parseDouble(result.getText());
if(lastCommand.equals("/"))
res=preRes/secVal;
else if(lastCommand.equals("*"))
res=preRes*secVal;
else if(lastCommand.equals("-
")) res=preRes-secVal;
else if(lastCommand.equals("+"))
res=preRes+secVal;
result.setText(" "+res);
lastCommand="=";
```

```
}
else if(ae.getSource()==mul)
{
preRes=Double.parseDouble(result.getText());
lastCommand="*";
result.setText("0");
}
else if(ae.getSource()==plus)
{
preRes=Double.parseDouble(result.getText());
lastCommand="+";
result.setText("0");
}
}
}
```

C. Develop a Java application for handling mouse events

Source code:

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="Mouse" width=500 height=500>
</applet>
*/
public class Mouse extends Applet
```

```
implements MouseListener,MouseMotionListener
{
int X=0,Y=20;
 String msg="MouseEvents";
 public void init()
 {
 addMouseListener(this);
 addMouseMotionListener(this);
setBackground(Color.black);
 setForeground(Color.red);
 }
 public void
 mouseEntered(MouseEvent m)
 {
setBackground(Color.magenta);
showStatus("Mouse Entered");
 repaint();
}
 public void mouseExited(MouseEvent m)
 {
 setBackground(Color.black);
 showStatus("Mouse Exited");
 repaint();
}
 public void mousePressed(MouseEvent m)
 {
```

```
 X=10;
Y=20;
msg="NEC";
setBackground(Color.green);
 repaint();
 }
 public void mouseReleased(MouseEvent m)
 {
 X=10;
 Y=20;
 msg="Engineering";
 setBackground(Color.blue);
repaint();
 }
 public void mouseMoved(MouseEvent m)
 {
 X=m.getX();
 Y=m.getY();
 msg="College";
 setBackground(Color.white);
 showStatus("Mouse Moved");
 repaint();
 }
 public void mouseDragged(MouseEvent m)
 {
 msg="CSE";
```

```
setBackground(Color.yellow);


showStatus("MouseMoved"+m.getX()+" "+m.getY());
 repaint();
 }
 public void mouseClicked(MouseEvent m)
 {
 msg="Students";
 setBackground(Color.pink);
 showStatus("MouseClicked");
 repaint();
 }
 public voidpaint(Graphics g)
 {
 g.drawString(msg,X,Y);
}
}
```