Assignment answers

1 a) Create an AWT application which works as a Counter with Increment and Clear buttons.

```
import java.awt.*;
import java.awt.event.*;

class Counter extends Frame implements ActionListener{
	Label lbl;
	TextField txt;
	Button btnInc,btnClr;

	Counter(){
		setLayout(new FlowLayout());
		lbl=new Label("Value:");
		txt=new TextField("0");
		btnInc=new Button("Increment");
		btnClr=new Button("Clear");
		add(lbl);
		add(txt);
		add(btnInc);
		add(btnClr);
		btnInc.addActionListener(this);
		btnClr.addActionListener(this);
		setSize(200,100);
		setVisible(true);
		addWindowListener(new WindowAdapter(){
			public void windowClosing(WindowEvent e){
				System.exit(0);
			}
		});
	}
```

```
	public void actionPerformed(ActionEvent e){
		if(e.getSource()==btnInc){
			int i=Integer.valueOf(txt.getText());
			++i;
			String s=String.valueOf(i);
			txt.setText(s);
		}
		else
			txt.setText("0");
	}
	public static void main(String[] args){
		new Counter();
	}
}
```

2 b) Create an AWT application which shows a list of states and the capital is displayed when a state is selected.

```
import java.awt.event.*;
import java.awt.*;
class States extends Frame implements ActionListener,ItemListener{
	Label s;
	Button c;
	TextField t;
	Choice l;
	WEHandler wh;
	String[] f={"AP","TELANGANA","GOA","KARNATAKA"};
	States(){
		super("States");

		setLayout(new FlowLayout());
		s=new Label("Selectstate");
		c=new Button("Clear");
		t=new TextField(20);
```

```
      l=new Choice();
      for(int i=0;i<4;i++)
            l.add(f[i]);
      l.addItemListener(this);
      c.addActionListener(this);
      wh=new WEHandler();
      addWindowListener(wh);
      add(s);
      add(t);
      add(l);
      add(c);
      setSize(300,300);
      setVisible(true);
      setLocation(500,200);
}
public void actionPerformed(ActionEvent e){
      t.setText(" ");
}
public void itemStateChanged(ItemEvent e){
      if(l.getSelectedItem()==f[0])
            t.setText("Amaravathi");
      else if(l.getSelectedItem()==f[1])
            t.setText("Hyderabad");
      else if(l.getSelectedItem()==f[2])
            t.setText("Panaji");
      else if(l.getSelectedItem()==f[3])
            t.setText("Banglore");
}
class WEHandler extends WindowAdapter{

      public void windowClosing(WindowEvent e){
            System.exit(0);
```

```
            }
      }
}

class ListDemo{
      public static void main(String[] args){
            States l=new States();
      }
}
```

3 b) Create an AWT application which acts as a scribble pad.

```
import java.awt.event.*;
import java.awt.*;

class Draw extends Frame{

      Draw(){
            super("ScriblePad");
            setLayout(new FlowLayout());
            ScriblePad c=new ScriblePad();
            add(c);
            c.setPreferredSize(new Dimension(200,200));
            c.setBackground(Color.gray);
            addWindowListener(new WindowAdapter(){
                  public void windowClosing(WindowEvent e){
                        System.exit(0);
                  }
            });
            setSize(300,300);
            setVisible(true);
            setLocation(500,200);
      }
```

```
	class ScriblePad extends Canvas{
		int x,y,prevx,prevy;
		ScriblePad(){
			addMouseListener(new MouseAdapter(){
				public void mousePressed(MouseEvent e){
					prevx=e.getX();
					prevy=e.getY();
				}
			});
			addMouseMotionListener(new MouseMotionAdapter(){
				public void mouseDragged(MouseEvent e){
					x=e.getX();
					y=e.getY();
					repaint();
				}
			});
		}
		public void update(Graphics g){
			g.drawLine(x,y,prevx,prevy);
			prevx=x;
			prevy=y;
		}
	}
}
class CanvasDemo{
	public static void main(String[] args){
		new Draw();
	}
}
```

4 b) Create a Sample editor using FileDialog class.

```
import java.awt.*;
```

```
import java.awt.event.*;
import java.io.*;

class SimpleEditor extends Frame{
	Panel p;
	Button btnO,btnS,btnC;
	TextArea ta;
	SimpleEditor(){
		p=new Panel();
		p.add(btnO=new Button("Open"));
		p.add(btnS=new Button("Save"));
		p.add(btnC=new Button("Clear"));
		add(p,"North");
		add(ta=new TextArea());
		btnO.addActionListener(new ActionListener(){
			public void actionPerformed(ActionEvent e){
				FileDialog fd=new
FileDialog(SimpleEditor.this,"Choose File",FileDialog.LOAD);
				fd.setVisible(true);
				String s=fd.getDirectory()+fd.getFile(),l;
				try{
					BufferedReader br=new BufferedReader(new
FileReader(s));
					while((l=br.readLine())!=null)
						ta.append(l+"\n");
					br.close();
				}
				catch(Exception k){}
			}
		});
		btnS.addActionListener(new ActionListener(){
			public void actionPerformed(ActionEvent e){
				FileDialog fs=new
FileDialog(SimpleEditor.this,"Choose File",FileDialog.SAVE);
```

```
                        fs.setVisible(true);
                        String s=fs.getDirectory()+fs.getFile(),l;
                        try{
                              BufferedWriter bw=new BufferedWriter(new
FileWriter(s));
                              l=ta.getText();
                              bw.append(l+"\n");
                              bw.close();
                        }
                        catch(Exception k){}
                  }
            });
            btnC.addActionListener(new ActionListener(){
                  public void actionPerformed(ActionEvent e){
                        ta.setText("");
                  }
            });
            addWindowListener(new WindowAdapter(){
                  public void windowClosing(WindowEvent e){
                        System .exit(0);
                  }
            });
            setSize(300,300);
            setVisible(true);
      }
      public static void main(String[] args){
            new SimpleEditor();
      }
}
```

5 b) Create an AWT application with Menus.

```
import java.awt.*;
import java.awt.event.*;
```

```
import java.io.*;

class MenuBarDemo extends Frame{
      MenuBar mb;
      Menu fileMenu,exitMenu;
      MenuItem OMenuItem,SMenuItem,CMenuItem,exitMenuItem;
      TextArea ta;
      MenuBarDemo(){
            super("MenuBar");
            mb=new MenuBar();
            setMenuBar(mb);
            mb.add(fileMenu=new Menu("File"));
            mb.add(exitMenu=new Menu("exit"));
            exitMenu.add(exitMenuItem=new MenuItem("Exit"));
            fileMenu.add(OMenuItem=new MenuItem("Open"));
            fileMenu.add(SMenuItem=new MenuItem("Save"));
            fileMenu.addSeparator();
            fileMenu.add(CMenuItem=new MenuItem("Clear"));
            add(ta=new TextArea());
            exitMenuItem.addActionListener(new ActionListener(){
                  public void actionPerformed(ActionEvent e){
                        System.exit(0);
                  }
            });
            OMenuItem.addActionListener(new ActionListener(){
                  public void actionPerformed(ActionEvent e){
                        FileDialog fd=new
FileDialog(MenuBarDemo.this,"Choose File",FileDialog.LOAD);
                        fd.setVisible(true);
                        String s=fd.getDirectory()+fd.getFile(),l;
                        try{
                              BufferedReader br=new BufferedReader(new
FileReader(s));
```

```
                        while((l=br.readLine())!=null)
                              ta.append(l+"\n");
                        br.close();
                  }
                  catch(Exception k){}
            }
      });
      SMenuItem.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                  FileDialog fs=new
FileDialog(MenuBarDemo.this,"Choose File",FileDialog.SAVE);
                  fs.setVisible(true);
                  String s=fs.getDirectory()+fs.getFile(),l;
                  try{
                        BufferedWriter bw=new BufferedWriter(new
FileWriter(s));
                        l=ta.getText();
                        bw.append(l+"\n");
                        bw.close();
                  }
                  catch(Exception k){}
            }
      });
      CMenuItem.addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                  ta.setText("");
            }
      });
      addWindowListener(new WindowAdapter(){
            public void windowClosing(WindowEvent e){
                  System.exit(0);
            }
      });
```

```
            setSize(300,300);
            setVisible(true);
      }
      public static void main(String[] args){
            new MenuBarDemo();
      }
}
```

6 b) Create a Swing application with JTable and JTree.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.applet.*;
import javax.swing.tree.*;
//<applet code="SwingDemo" width=650 height=400></applet>
public class SwingDemo extends Applet{
      String[][]
data={{"Anil","10","M"},{"Anusha","9","F"},{"Jagadeesh","12","M"},{"
Mounica","8","F"},{"Mahidhar","5","M"},{"Padmini","3","F"}};
      String[] col={"Name","Age","Gender"};
      JTable table;
      JTree tree;
      JLabel jlab;
      public void init(){
            table=new JTable(data,col);
            JScrollPane jsp = new JScrollPane(table);
            add(jsp);

            DefaultMutableTreeNode top,a,a1,a2,b,b1,b2,b3;
            top= new DefaultMutableTreeNode("Options");
            a = new DefaultMutableTreeNode("A");
            top.add(a);
            a1 = new DefaultMutableTreeNode("A1");
```

```
            a.add(a1);
            a2 = new DefaultMutableTreeNode("A2");
            a.add(a2);
            b = new DefaultMutableTreeNode("B");
            top.add(b);
            b1 = new DefaultMutableTreeNode("B1");
            b.add(b1);
            b2 = new DefaultMutableTreeNode("B2");
            b.add(b2);
            b3 = new DefaultMutableTreeNode("B3");
            b.add(b3);
            tree = new JTree(top);
            JScrollPane jsp1 = new JScrollPane(tree);
            add(jsp1);
            setVisible(true);
      }
}
```