PRACTICAL LIST for JAVA

1 WAP to find the average and sum of the N numbers Using Command line argument.
2 WAP to Demonstrate Type Casting.
3 WAP to find the number of arguments provide at runtime.
4 WAP to Test the Prime number.
5 WAP to calculate the Simple Interest and Input by the user.
6 WAP to create a Simple class to find out the Area and perimeter of rectangle and box using super and this keyword.
7 WAP to find G.C.D of the number.
8 WAP to design a class account using the inheritance and static that show all function of bank (withrowal, deposite).
9 WAP to find the factorial of a given number using Recursion.
10 WAP to design a class using abstract Methods and Classes.
11 WAP to design a String class that perform String Method(Equal,Reverse the string,change case).
12 WAP to handle the Exception using try and multiple catch block.
13 WAP that Implement the Nested try Statements.
14 WAP to Create a package that access the member of external class as well as same package.
15 WAP that import the user define package and access the Member variable of classes that Contained by Package.
16 WAP that show the partial implementation of Interface.
17 WAP to Handle the user defined Exception using throw keyword.
18 WAP to create a thread that Implement the Runable interface.
19 WAP to Implement Interthread communication.
20 WAP to create a class component that show controls and event handling on that controls.(math calc).
21 WAP to Draw the line, Rectangle, oval,text using the graphics method.
22 WAP to create a Menu using the frame.
23 WAP to create a Dialogbox.
24 WAP to Implement the flow layout And Border Layout.
25 WAP to Implement the GridLayout, CardLayout.
26 Wap of Awtdemo2 given by me.
27 WAP to demonstrate System clock.
28 WAP to create Frame that display the student information.

Java Lab Solution

Program 1: Print Message.

```
class firstprogram
{
public static void main(String arg[])
{
System.out.println("this is java's first program");
}
}
```

Execution steps:

Javac firstprogram.java (File name)
Java firstprogram (class name)

Program 2: WAP to find the average,sum,min and max of the N numbers Using user Input.

```
import java.util.*;
class Average{
	public static void main(String args[])

	{	Scanner sc= new Scanner(System.in);// to take user input
		int choice;
		int a=0,min=0,max=0,x;
		int n =args.length;
		System.out.println("1-sum");
		System.out.println("2-Average");
		System.out.println("3-Minimum");
		System.out.println("4-Maximum");
		System.out.println("Enter Ur Choice : ");
		choice=sc.nextInt();
		for(int i=0;i<n;i++){
			a+=Integer.parseInt(args[i]);//to convert string into Integer
		}
		switch(choice)
		{
			case 1 :System.out.println("The sum is : "+a);
			break;
			case 2 :System.out.println("The Average is  : "+a/n);
			break;
			case 3 :for(int i=0;i<n-1;i++)
			{	x=Integer.parseInt(args[i]);
				if(x<Integer.parseInt(args[i+1]))
```

```
                        min=x;
                        else min=Integer.parseInt(args[i+1]);
                }
                System.out.println("The minimum is : "+min);
                break;
                case 4 :
                for(int i=0;i<n-1;i++)
                {       x=Integer.parseInt(args[i]);
                        if(x>Integer.parseInt(args[i+1]))
                                max=x;
                                else
                                        max=Integer.parseInt(args[i+1]);
                }
                System.out.println("The maximum is : "+max);

                break;
            }
        }
}
```

Program 3: WAP to Demonstrate Type Casting.

```
class typecast
{
        public static void main(String args[])
        {
                byte h=127;
                 int a=300;
                float a1=12.222f;
                 float  g;
                 short b=200;
                 long c=999999;
                 float e=345.89F;
                 double f=45645.78222222222222;
                        g= (float)f;


                 System.out.println("short b ="+g);

                 System.out.println("short b ="+b);
                 System.out.println("long c ="+c);
                 System.out.println("float e="+e);
                 System.out.println("double f="+f);
                 System.out.println("short b="+b);
                 System.out.println("short to byte "+(byte)b);
```

```
                System.out.println("int to byte "+(byte)a);
                System.out.println("int to float"+(float)a);
                System.out.println("long to byte "+(byte)c);
                System.out.println("double to long "+(long)f);
                System.out.println("double to int "+(int)f);
                System.out.println("double to byte "+(byte)f);
                System.out.println("double to short "+(short)f);
                System.out.println("double to float "+(float)f);
                System.out.println("float to int "+(int)e);
                System.out.println("float to byte "+(byte)e);
                System.out.println("float to short "+(short)e);
                System.out.println("float to long "+(long)e);
                System.out.println("float to double ="+(double)e);
                System.out.println("long to int"+(int)c);
                System.out.println("byte to int ="+(int)h);
        }
}
```

Program 4: WAP to Test the Prime num.

```
 import java.util.*;
class prime
{
        public static void main(String args[])
        {       int flag,x,i;
                flag=0;
                int a[]=new int[7];
                for(x=0;x<args.length;x++)
                {

                        a[x]=Integer.parseInt(args[x]);
                        for(i=2;i<(a[x]/2);i++)
                        {
                                if((a[x]%i)==0)
                                {
                                        break;

                                }
                                else flag=1;

                        }
                        if(flag==1)
                        System.out.println(a[x]+" is a prime no ");
                        else
                        System.out.println(a[x]+" is not a prime no ");
                        flag=0;
                }
```

```
	}
}
```

Program 5: WAP to find out the HCF and LCF.

```
import java.util.*;
class hcf
{
	public static void main(String args[])
	{
		int a,b;
		Scanner sc= new Scanner(System.in);
		System.out.println("Enter two nos :");
		a=sc.nextInt();
		b=sc.nextInt();
		int big;
		int small;
		if(a>b)
		{
			big=a;
			small=b;
		}
		else
		{
			big=b;
			small=a;
		}
		for(int i=1;i<=big;i++)
		{
			if(((big*i)%small)==0)
			{
			int lcm=big*i;
			System.out.println("The least common multiple is "+(lcm));
			break;
			}
		}
		int temp=1;
		while(temp!=0)
		{
		temp=big%small;
		if(temp==0)
		{
			System.out.println("GCD is "+small);
		}
		else
		{
			big=small;
			small=temp;}
```

}}}

Program 6: WAP to calculate the Simple Interest and Input by the user.

```
import java.util.*;
class si
{
	int p,t;
	float si,r;
	public si()
	{
	r=0;
	p=0;
	}
	public void getdata()
	{
	Scanner sc =new Scanner(System.in);
	System.out.println("Enter principle : ");
	p=sc.nextInt();
	System.out.println("Enter rate : ");
	r=sc.nextFloat();
	System.out.println("Enter time period : ");
	t=sc.nextInt();
	}
	public void cal()
	{
	si=(p*r*t)/100;
	}
	public void display()
	{
	System.out.println("Principle : Rs"+p);
	System.out.println("Rate : "+r);
	System.out.println("Time period : "+t);
	System.out.println("Simple Interest : Rs"+si);
	}
	public static void main(String args[])
	{
	si s = new si();
	s.getdata();
	s.cal();
	s.display();
	}
}
```

Program 7:

WAP to create a Simple class to find out the Area and perimeter of rectangle and box using super and this keyword .

```
class rect
{
        int l,b;
        public rect(int l,int b)
        {       this.l=l;
                this.b=b;
        }
        public int area()
        {
                return l*b;
        }
}
class box extends rect
{
        int d;
        public box(int l,int b,int d)
        {
                super(l,b);
                this.d=d;
        }
        public int volume()
        {
                int vol = area()*d;
                return vol;
        }

        public static void main(String args[])
        {       int vol ,area;
                System.out.println("derived object in derived reference");
                rect r= new rect(10,20);
                area=r.area();
                System.out.println("area is "+area+"\n");

                System.out.println("base object in base reference");
                box b = new box(10,20,30);
                vol=b.volume();
                area=b.area();
                System.out.println("area is "+area);
                System.out.println("volume is "+vol+"\n");

                System.out.println("derived object in base reference");
                rect b1= new box(10,90,70);
                area = b1.area();
```

```
                //vol=b1.volume(); as with refernce of base class we can't call derived's
method
                System.out.println("area is "+area);

                //as super class doesn't knw abt the base class but reference can be
assigned
                /*System.out.println("base object in derived reference");
                box b2=(new rect (10,20));
                vol = b2.area();
                System.out.println("area is "+area);*/

                r=b;
                System.out.println(r.area());
                System.out.println(r.volume());
        }
}
```

Program 8:

WAP to design a class account using the inheritance and static that show all function of bank(withrowal,deposite) and generate account number dyanamically.

```
import java.util.*;
class bank
{       static int acc_no =10001;
        float amt;

        public void display()
        {
                System.out.println("Account no :"+acc_no );
                System.out.println("Current Amount :"+amt );
        }

        public bank()
        {
                amt=1000;
                System.out.println("Ur account no is "+acc_no);
                acc_no++;
        }
        public void getamt()
        {
        System.out.println("Current balance :"+amt);
        }
        public void withdraw(float x)
        {
                if(amt==1000 || amt<=x )
```

```
                {
                System.out.println("Sorry u can't withdraw");
                }

                else
                {
                amt=amt-x;
                System.out.println("amount withdrawn :"+x);
                System.out.println("After withdrawl");
                getamt();

                }
        }
        public void deposit(float x)
        {
        if(x==0.0)
        System.out.println("OOPS 0 can't be deposited");
        else {
        amt+=x;
        System.out.println("After deposition");
        getamt();}

        }
        public static void main(String args[])
        {       Scanner sc = new Scanner(System.in);
                bank b1 = new bank();
                b1.deposit(0);
                b1.withdraw(120.5f);
                b1.display();
                System.out.println("\n");
                bank b2 = new bank();
                b2.deposit(1000.0f);
                b2.withdraw(150.5f);

        }
}
```

Program 9: WAP to design a class Shape (Implement Runtime polymorphim) using abstract Methods and Classes.

```
class AbstractDemo1
{
        public static void main(String args[])
        {
```

```
                Shape  shape;
                Rectangle r = new Rectangle();
                 r.setDimensions(40,20);

                shape  = r;
                System.out.println(shape.getArea());
                System.out.println(shape.getPerimeter());
        }
}
```

abstract class Shape

```
{
        void someMethod()
        {
                System.out.println("This is some method");
        }
        abstract float getArea();
        abstract float getPerimeter();
}
```

class Square extends Shape

```
{
        float side;
        Square()
        {
                side = 0;
        }
        Square(float side)
        {
                this.side = side;
        }
        void setSide(float side)
        {
                this.side = side;
        }
        float getArea()
        {
                return side * side;
        }
        float getPerimeter()
        {
                return 4 * side;
        }
}
import java.awt.*;
```

```
class MyCircle extends MyShape
{
	private float radius;
	static float pi;

	static
	{
		pi = 22 / 7.0f;
	}

	MyCircle()
	{
		super("circle");
		radius = 0;
	}

	MyCircle(float radius)
	{
		super("circle");
		this.radius = radius;
	}

	void setDimensions(float radius)
	{
		this.radius = radius;
	}

	void showDimensions()
	{
		System.out.println("radius  : " + radius);
	}

	float getArea()
	{
		return radius * radius * pi;
	}
}
class Rectangle extends Shape
{
	private float length;
	private float breadth;
	Rectangle()
	{
		length = breadth = 0;
	}
```

```
	Rectangle(float length, float breadth)
	{
		setDimensions(length, breadth);
	}
	void setDimensions(float length, float breadth)
	{
		this.length = length;
		this.breadth = breadth;
	}
	float getArea()
	{
		return length * breadth;
	}
	float getPerimeter()
	{
		return (2 * (length + breadth));
	}
}
```

Program 10:WAP to design a String class that perform String Method(Equal, Reverse the string, change case, trim etc.)

```
public class StringDemo
{
	public static void main(String args[])
	{
		String str = "This is some sample String with some words that have been
		repeated some times";
		System.out.println("Total no. of characters : " + str.length());
		System.out.println("To Upper Case : " + str.toUpperCase());
		System.out.println("To Lower Case : " + str.toLowerCase());
		System.out.println("Original String : " + str);

		System.out.println(str.substring(8));
		System.out.println(str.substring(8,19));

		System.out.println(str.indexOf("some"));


		String s = "        " + str  + "     ";
		System.out.println(s);
		System.out.println("[" + s.trim() + "]");

		System.out.println(str.replace("s","$$##"));
```

```
            String sh = "parth is a good boy";
            System.out.println(sh + " -> " + new StringBuffer(sh).reverse());
        }}
```

Program 11: WAP to handle the Exception using try and multiple catch block.

```
class exception
{
        public static void main(String args[]){

    try{
        int d=42;
        int a =0;
        int c=d/a;
    }
    catch(ArithmeticException e){
    System.out.println("Division by zero error");
    }
        }
}
```

Other Example:

```
public class ExceptionHandling
{
        public static  void main(String args[])
        {
                String num[]={"123","456","abc","789"};
                int sum=0;
                int i;
                for(i=0;i<=num.length;i++)
                {
                        try{
                                sum+=Integer.parseInt(num[i]);
                        }
                        catch(NumberFormatException e)

                        { System.out.println("NUMBER FORMAT ERROR");
                        }
                        catch(ArrayIndexOutOfBoundsException e)

                        {System.out.println("ARRAY ERROR");
                                }
                                finally
                                { System.out.println("i = "+i);
                                        }

                }
```

```
                System.out.println("sum is"+sum);
        }

}
```

Program 12:WAP that Implement the Nested try Statements.

```
class NestedTry
{
        public static void main(String args[])
        {       int a=args.length;
                try{
                        int d=42/a;
                        try
                        {
                                if(a==1){
                                        int c= a/(a-a);}
                                if(a==2)
                                {
                                        int c[]={2,3,4};
                                        c[5]=90;
                                }
                        }
                        catch(ArrayIndexOutOfBoundsException e)
                        {e.printStackTrace();
                                }
                }
                catch(ArithmeticException e)
                {
                        e.printStackTrace();
                }
        }
}
```

Program 13:WAP that Implement Throw and Throws.

```
class ThrowDemo
{
        ThrowDemo()
        {
                try
                { throw new NullPointerException();
                }
                catch(NullPointerException e)
                {
```

```
                System.out.println("Caught in constructor");
                throw e;
        }
    }
    public static void main(String args[])
    {
        try{
                ThrowDemo td=new ThrowDemo();
        }
        catch(NullPointerException e)
        {
                System.out.println("Caught in Main");
        }
    }
}

class ThrowsDemo
{
    ThrowsDemo() throws NullPointerException
    { System.out.print("in constructor");
        throw new NullPointerException();


    }
    public static void main(String args[])
    {
        try{

                ThrowsDemo td=new ThrowsDemo();
        }
        catch(NullPointerException e)
        {
                System.out.println("Caught in Main");
        }
    }
}
```

Program 14: WAP that Implement Custom Exception.

```
import java.util.*;
class MyException  extends Exception
{
        private int e;
    MyException (int a )
    {
        e=a;
```

```
	}
	public String toString()
	{
		return ("Error in entry"+e);
	}
}

public class mine
{	public void  compute(int a) throws MyException
	{
		int age=a;
		if(age>150)
		throw new MyException (age);
		System.out.println("COrrect age");
	}

	public static void main(String args[])
	{
		mine m=new mine();
		try{
			m.compute(1);
			m.compute(789);
		}
		catch(MyException  e)
		{
			System.out.println(e);
		}
	}
}
```

Program 15: WAP to Create a package that access the member of external class as well as same package.

```
package pack;
class base
{
	public static void main(String arg[])
	{
		System.out.println("Base class(p1)");
		p1 w=new p1();
		//w.f1();
		System.out.println("Derived  class(p2)");
		p2 x=new p2();
	//	x.f2();
		System.out.println("Simple class(p3)");
		p3 y=new p3();
```

```
	//	y.f3();
	}
}


package pack;
public class p1
{
	int a=1;
	public int b=2;
	private int c=3;
	protected int d=4;
	public p1()
	{
		System.out.println("Value of a="+a);
		System.out.println("Value of b="+b);
		System.out.println("Value of c="+c);
		System.out.println("Value of d="+d);
	}
}

package pack;
class p2 extends p1
{
	p2()
	{
		System.out.println("Value of a="+a);
		System.out.println("Value of b="+b);
		//System.out.println("Value of c="+c);
		System.out.println("Value of d="+d);
	}
}

package pack;
class p3
{
	p1 p=new p1();
	p3()
	{
		System.out.println("Value of a="+(p.a));
		System.out.println("Value of b="+(p.b));
		//System.out.println("Value of c="+(p.c));
		System.out.println("Value of d="+(p.d));
	}
}
```

```
package pack1;
class simple extends pack.p1
{
	public simple()
	{

	//	System.out.println("Value of a="+a);
		System.out.println("Value of b="+b);
	//	System.out.println("Value of c="+c);
		System.out.println("Value of d="+d);
	}
}

package pack1;
class s2
{
	public static void main(String arg[])
	{
		simple s=new simple();
		s1 p=new s1();
	}
}

package pack1;
class s1
{
	 s1()
	{
		pack.p1 z=new pack.p1();
	//	System.out.println("Value of a="+(z.a));
		System.out.println("Value of b="+(z.b));
	//	System.out.println("Value of c="+(z.c));
	//	System.out.println("Value of d="+(z.d));
	}
}
```

Program16: WAP that show the partial implementation of Interface.(calculation of Salary of Employee).

```
import java.util.*;
interface salary
{
	int  getsal();
}
```

```
abstract class employee
{	String name;
	int age;
	String sex;
	int sal;
	employee(String name,int age,String sex,int sal)
	{
		this.name=name;
		this.age=age;
		this.sex=sex;
		this.sal=sal;
	}
	abstract void display();
}
class labour extends employee implements salary
{	int wage;
	int hrs;
	labour(String name,int age,String sex,int sal,int hrs)
	{
		super(name,age,sex, sal);
		this.hrs=hrs;
	}

	public int getsal()
	{
		wage=sal*hrs;
		return wage;
	}
	void display()
	{
		System.out.println("name :"+name);
		System.out.println("Age :"+age);
		System.out.println("Sex :"+sex);
		System.out.println("salary : Rs"+sal);
		System.out.println("Hours worked :"+hrs);
		System.out.println("Wage of the daily labour :Rs"+getsal());
	}
}
class staff extends employee implements salary
{	int hra,da,ta;
	staff(String name,int age,String sex,int sal,
	int hra,int da,int ta)
	{
		super(name,age,sex, sal);
		this.da=da;
		this.ta=ta;
```

```
		this.hra=hra;
	}

	public int getsal()
	{
		int wage=sal+ta+da+hra;
		return wage;
	}
	void display()
	{
		System.out.println("name :"+name);
		System.out.println("Age :"+age);
		System.out.println("Sex :"+sex);
		System.out.println(" basic salary :Rs"+sal);
		System.out.println("Daily allowance : Rs"+da);
		System.out.println("Travel allowance : Rs"+ta);
		System.out.println("Household allowance : Rs"+hra);
		System.out.println("total salary  :Rs"+getsal());
	}
}
class sal
{
	public static void main(String args[])
	{	Scanner sc= new Scanner (System.in);
		int ch,da,ta,hra,sal,hrs,age;
		String name;
		String sex;
		System.out.println("Enter ur choice for salary calculation");
		System.out.println("1-labour");
		System.out.println("2-Staff");
		ch=sc.nextInt();
		switch(ch)
		{
			case 1 :
				System.out.println("Enter the following for a lobour");
				System.out.print("Name :");
				name=sc.next();
				System.out.println("age :");
				age=sc.nextInt();
				System.out.println("Sex : ");
				sex=sc.next();
				System.out.println("salary :");
				sal=sc.nextInt();
				System.out.println("daily working hours :");
				hrs=sc.nextInt();
				labour l = new labour(name,age,sex,sal,hrs);
```

```
                    l.display();
                    break;
                    case 2 :
                    System.out.println("Enter the following for a Staff");
                    System.out.println("Name :");
                    name=sc.next();
                    System.out.println("age :");
                    age=sc.nextInt();
                    System.out.println("Sex : ");
                    sex=sc.next();
                    System.out.println("salary : ");
                    sal=sc.nextInt();
                    System.out.println("daily allowance :");
                    da=sc.nextInt();
                    System.out.println("travel allowance :");
                    ta=sc.nextInt();
                    System.out.println("household allowance :");
                    hra=sc.nextInt();
                    staff s = new staff(name,age,sex,sal,hra,da,ta);
                    s.display();
                    break;

            }
        }
```

Program 17:

WAP to create Arithmetic Math Calculator Using Applet Class ant Event Handling.

```
/*<APPLET  CODE ="calc.class" WIDTH =300 HEIGHT =400>
</APPLET>*/

import java.awt.event.*;
import java.awt.*;
import java.applet.Applet;
public class calc extends Applet  implements ActionListener
{       Button add,sub,divide,multi;
        Label result,no1,no2;
        TextField tf,ip1,ip2;
        Panel p1,p2,p3;
        public  void init()
        {
                add=new Button("ADD");
                sub=new Button("SUBTRACT");
                divide=new Button("DIVIDE");
                multi=new Button("MULTIPLY");
```

```
        result = new Label("Result = ");
        no1=new Label ("NUMBER 1:");
        no2=new Label ("NUMBER 2:");
        tf=new TextField(20);
        ip1=new TextField(10);
        ip2=new TextField(10);
        p1=new Panel();
        p2=new Panel();
        p3=new Panel();

        tf.setEditable(false);
        add.setSize(20,40);
        sub.setSize(20,40);
        divide.setSize(20,40);

        add.addActionListener(this);
        sub.addActionListener(this);
        divide.addActionListener(this);
        multi.addActionListener(this);

        setLayout(new FlowLayout());

        p1.add(no1);
        p1.add(ip1);
        p1.add(no2);
        p1.add(ip2);
        p2.add(add);
        p2.add(sub);
        p2.add(divide);
        p2.add(multi);
        p3.add(result);
        p3.add(tf);
        add(p1);
        add(p2);
        add(p3);

        setSize(400,200);
        setVisible(true);

}
public void actionPerformed(ActionEvent e)
{       int a,b;
        int result;
        a =Integer.parseInt(ip1.getText());
        b=Integer.parseInt(ip2.getText());
        if(e.getSource()==add)
```

```
                {System.out.println("ADD");
                result=(a+b);
                tf.setText("Addition :"+String.valueOf(result));
                }
                if(e.getSource()==sub)
                {
                result=(a-b);
                tf.setText("Subtraction : "+String.valueOf(result));
                }
                if(e.getSource()==multi)
                {
                result=(a*b);
                tf.setText("Multiplication : "+String.valueOf(result));
                }
                if(e.getSource()==divide)
                {
                        try{
                        if(b==0)
                        {
                                result=(a/b);
                                tf.setText("Division :"+String.valueOf(result));
                        }
                        }
                        catch(ArithmeticException ae )
                        {
                        tf.setText("Division can't be performed");
                        }
                }

        }
}
```

Program 18: WAP to Draw the line, Rectangle, oval, text etc using the graphics method.

```
/*<applet code= "AppletDemo.class" width  = "500"          height  = "300">
</applet>*/

import java.applet.Applet;
import java.awt.*;
public class AppletDemo extends Applet
{
        public void init()
        {setBackground(Color.cyan);
        }
        public void paint(Graphics g)
        {
```

```
                Font f=new Font("TIMES NEW ROMAN ",Font.ITALIC,32);
                g.setFont(f);
                g.setColor(Color.orange);
                g.drawString("WELCOME TO APPLET ",30,30);
                g.fillOval(60,60,150,150);
                g.setColor(Color.black);
                g.fillOval(90,100,20,20);
                g.fillOval(160,100,20,20);
                g.setColor(Color.RED);
                g.drawLine(120,150,150,150);
                g.drawLine(120,150,140,130);
                g.drawArc(90,130,90,60,0,-180);

        }
}
```

Program 19: WAP to create a frame Window Using Frame Class.

```
  public class AWT1
{
        public static void main(String args[])
        {
                MyFrame mf = new MyFrame();
        }
}


import java.awt.event.*;

/*
class MyWindowListener implements WindowListener
{
        public void windowActivated(WindowEvent we){}
        public void windowDeactivated(WindowEvent we){}
        public void windowOpened(WindowEvent we){}
        public void windowClosed(WindowEvent we){}
        public void windowIconified(WindowEvent we){}
        public void windowDeiconified(WindowEvent we){}

        public void windowClosing(WindowEvent we)
        {
                System.exit(0);
        }

}
*/
```

```
class MyWindowAdapter extends WindowAdapter
{
	public void windowClosing(WindowEvent we)
	{
		System.exit(0);
	}
}


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


class MyFrame extends Frame //implements WindowListener
{
	MyFrame()
	{
		super("Sample Java Frame");
		//MyWindowListener mwl = new MyWindowListener();
		//MyWindowAdapter mwa = new MyWindowAdapter();
		//addWindowListener(mwa);

		addWindowListener(new MyWindowAdapter());
		//addWindowListener(this);

		setSize(400,300);
		setResizable(true);
		//setUndecorated(true);
		setVisible(true);
	}
}
```

Program 19: WAP to create UI component on Frame Window Using Frame Class.

```
  public class AWT1
{
	public static void main(String args[])
	{
		MyFrame mf = new MyFrame();
	}
}
```

```
import java.awt.event.*;
class MyWindowAdapter extends WindowAdapter
{
	public void windowClosing(WindowEvent we)
	{
		System.exit(0);
	}
}



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

class MyFrame extends Frame
{
	Label lbl, l2;

	TextField t1;

	MyFrame()
	{
		super("Sample Java Frame");
		addWindowListener(new MyWindowListener());
		setSize(500,400);
		addControls();
		setVisible(true);
	}

	private void addControls()
	{
		setLayout(null);

		lbl = new Label("Sample Label",Label.CENTER);


		lbl.setSize(250,22);
		lbl.setLocation(10,40);


		//lbl.setText("This is the text in the label control");
		lbl.setBackground(Color.YELLOW);
		lbl.setForeground(Color.RED);
```

```
                //lbl.setAlignment(Label.RIGHT);

                add(lbl);

                l2 = new Label(lbl.getText());
                l2.setSize(lbl.getSize());
                l2.setLocation(lbl.getLocation().x, lbl.getLocation().y + 30);
                l2.setAlignment(lbl.getAlignment());
                l2.setBackground(lbl.getForeground());
                l2.setForeground(lbl.getBackground());
                //l2.setVisible(false);
                add(l2);

                t1 = new TextField("This is some initial text in the text box control");
                t1.setSize(200,22);
                t1.setLocation(10,160);

                t1.setEchoChar('^');
                if (t1.echoCharIsSet())
                {
                        System.out.println("Input has been masked");
                        System.out.println("Mask character is "  + t1.getEchoChar());
                }
                add(t1);

                t1.setEnabled(false);
        //      t1.setEditable(false);
        }
}
```

Program 20: WAP to implement ListBox.

```
  public class AWT1
{
        public static void main(String args[])
        {
                MyFrame mf = new MyFrame();
        }
}
import java.awt.*;
import java.awt.event.*;

public class MyFrame extends Frame implements ActionListener
{
        List lst;
        Button btn;
        MyFrame()
```

```
	{
		super("Sample Java Frame");
		addWindowListener(new MyWindowAdapter());
		setSize(500,400);
		addControls();
		setVisible(true);
	}

	private void addControls()
	{
		setLayout(null);

		lst = new List();
		lst.setLocation(30,50);
		lst.setSize(200,300);
		lst.setMultipleMode(true);

		add(lst);

		lst.add("sfsdf");
		lst.add("55656");
		lst.add("dfgdfg");
		lst.add("sfsdf");
		lst.add("cvb");
		lst.add("sfcvbcbcvbcvbsdf");
		lst.add("bmmbnm");
		lst.add("ioouo");
		lst.add("qeqwe");
		lst.add(".m,.m,.");

		btn = new Button("Click Me");
		btn.addActionListener(this);
		btn.setSize(100,24);
		btn.setLocation(250,50);

		add(btn);

	}

	public void actionPerformed(ActionEvent ae)
	{
		System.out.println("Total Selected Items : " +
lst.getSelectedItems().length);
		String aItem[] = lst.getSelectedItems();
		int i;
		for (i=0;i<aItem.length;i++)
```

```
                        System.out.println(aItem[i]);
        }
}
```

```
import java.awt.event.*;
class MyWindowAdapter extends WindowAdapter
{
        public void windowClosing(WindowEvent we)
        {
                System.exit(0);
        }
}
```

Program 21: WAP to implement Choice, Checkbox, radio button With event handling.

```
  public class AWT1
{
        public static void main(String args[])
        {
                MyFrame mf = new MyFrame();
        }
}

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

class MyFrame extends Frame implements ItemListener
{
        Checkbox c1, c2, c3, c4, c5, c6;
        CheckboxGroup cbg1, cbg2;
        Choice cbo;
        MyFrame()
        {
                super("Sample Java Frame");
                addWindowListener(new MyWindowAdapter());
                setSize(500,400);
                addControls();
                setVisible(true);
        }

        private void addControls()
        {
```

```
setLayout(new FlowLayout());

cbg1 = new CheckboxGroup();
cbg2 = new CheckboxGroup();

c1 = new Checkbox("C", true);

c2 = new Checkbox("C++",cbg2,true);
c3 = new Checkbox("Java",cbg2, true);

c4 = new Checkbox("Prolog", cbg1,false);
c5 = new Checkbox("Lisp", true, cbg1);

c6 = new Checkbox("Fortran");

c1.addItemListener(this);
c2.addItemListener(this);
c3.addItemListener(this);
c4.addItemListener(this);
c5.addItemListener(this);
c6.addItemListener(this);

//c2.setState(true);
//c5.setState(false);

add(c1);
add(c2);
add(c3);
add(c4);
add(c5);
add(c6);

cbo = new Choice();
cbo.add("Delhi");
cbo.add("Ajmer");
cbo.add("Jaipur");
cbo.add("Mumbai");
cbo.insert("Beawar",0);
cbo.add("Chandigarh");
cbo.add("Jalandhar");
cbo.add("Nasirabad");
cbo.add("Bharatpur");

cbo.select(3);
cbo.addItemListener(this);
add(cbo);
```

```
	}

	public void itemStateChanged(ItemEvent ie)
	{
		if (ie.getSource() instanceof Checkbox)
		{
			Checkbox c = (Checkbox) ie.getSource();
			System.out.println(c.getLabel() + " : " + c.getState());
		}
		else if (ie.getSource() instanceof Choice)
		{
			System.out.println("Selected Index : " + cbo.getSelectedIndex());
			System.out.println("Selected Item  : " + cbo.getSelectedItem());
		}
	}
}
import java.awt.event.*;
class MyWindowAdapter extends WindowAdapter
{
	public void windowClosing(WindowEvent we)
	{
		System.exit(0);
	}
}
```

Program 22: WAP to implement Layout Manager.

```
  public class AWT1
{
	public static void main(String args[])
	{
		MyFrame mf = new MyFrame();
	}
}

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

class MyFrame extends Frame implements ActionListener
{
	Panel mainPanel, p1, p2, p3, p4, p5, topPanel;
	CardLayout cl;

	TextField txt[];
	Button btn[], b1, b2, b3, b4, b5;
```

```
TextArea ta[];
Label lbl[];
Choice choice[];

MyFrame()
{
	super("Sample Java Frame");
	addWindowListener(new MyWindowAdapter());
	setSize(400,300);
	addControls();
	setVisible(true);
}
private void addControls()
{
	cl = new CardLayout();
	mainPanel = new Panel();
	mainPanel.setLayout(cl);

	int i;
	GridLayout gl = new GridLayout(5,10,5,5);

	p1 = new Panel();
	p1.setLayout(gl);
	txt = new TextField[50];
	for (i=0;i<txt.length;i++)
	{
		txt[i] = new TextField("Text " + (i+1));
		p1.add(txt[i]);
	}

	p2 = new Panel();
	p2.setLayout(gl);
	btn = new Button[50];
	for (i=0;i<btn.length;i++)
	{
		btn[i] = new Button("Button " + (i+1));
		btn[i].addActionListener(this);
		p2.add(btn[i]);
	}

	p3 = new Panel();
	p3.setLayout(gl);
	ta = new TextArea[50];
	for (i=0;i<ta.length;i++)
	{
		ta[i] = new TextArea("Text " + (i+3));
```

```
        p3.add(ta[i]);
}

p4 = new Panel();
p4.setLayout(gl);
lbl = new Label[50];
for (i=0;i<lbl.length;i++)
{
        lbl[i] = new Label("Label " + (i+4));
        p4.add(lbl[i]);
}

p5 = new Panel();
p5.setLayout(gl);
choice = new Choice[50];
for (i=0;i<choice.length;i++)
{
        choice[i] = new Choice();
        p5.add(choice[i]);
}

mainPanel.add(p1,"panel1");
mainPanel.add(p2,"panel2");
mainPanel.add(p3,"panel3");
mainPanel.add(p4,"panel4");
mainPanel.add(p5,"panel5");

add(mainPanel);

b1 = new Button("Card 1");
b2 = new Button("Card 2");
b3 = new Button("Card 3");
b4 = new Button("Card 4");
b5 = new Button("Card 5");

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);

topPanel = new Panel();
topPanel.setLayout(new FlowLayout());

topPanel.add(b1);
topPanel.add(b2);
```

```
		topPanel.add(b3);
		topPanel.add(b4);
		topPanel.add(b5);

		add(topPanel, BorderLayout.NORTH);
	}

	public void actionPerformed(ActionEvent ae)
	{
		if (ae.getSource() == b1)
			cl.show(mainPanel,"panel1");
		else if (ae.getSource() == b2)
			cl.show(mainPanel,"panel2");
		else if (ae.getSource() == b3)
			cl.show(mainPanel,"panel3");
		else if (ae.getSource() == b4)
			cl.show(mainPanel,"panel4");
		else if (ae.getSource() == b5)
			cl.show(mainPanel,"panel5");
		else
		{
			int i;
			boolean found = false;
			for (i=0;i<btn.length;i++)
			{
				if (ae.getSource() == btn[i])
				{
					found = true;
					break;
				}
			}

			if (found)
			{
				System.out.println("Button Clicked from panel : " +
			 btn[i].getLabel());
			}
		}
	}
}

import java.awt.event.*;
class MyWindowAdapter extends WindowAdapter
{
	public void windowClosing(WindowEvent we)
	{
```

```
                System.exit(0);
        }
}
```

Program 23: WAP to implement Dialog box.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="DialogDemo" width =250 height = 250>
</applet>
*/
class SampleDialog extends Dialog implements ActionListener
{
        SampleDialog(Frame parent,String title)
        {
         super(parent,title,false);
         setLayout(new FlowLayout());
         setSize(300,200);

         add(new Label("Press this button: "));
         Button b;
         add(b= new Button("Cancel"));
         b.addActionListener(this);
        }

        public void actionPerformed(ActionEvent ae)
        {
         dispose();
        }
        public void paint(Graphics g)
        {
         g.drawString("This is in the dialog box",10,70);
        }
}

class MenuFrame extends Frame
{
String msg = "";
CheckboxMenuItem debug,test;

MenuFrame(String title)
        {
        super(title);
        MenuBar mbar = new MenuBar();
```

```
        setMenuBar(mbar);    //Menu Bar added on applet

        Menu file = new Menu("File");                   //Menu File is created
        MenuItem item1,item2,item3,item4,item5;         //Menu items for File created
        file.add(item1=new MenuItem("New..."));         //Menu Items added in Menu
        file.add(item2=new MenuItem("Open..."));
        file.add(item3=new MenuItem("Close"));
        file.add(item4=new MenuItem("-"));
        file.add(item5=new MenuItem("Quit..."));
        mbar.add(file);                                 // Menu File added on Menu
Bar


        Menu edit = new Menu("Edit");                   //Menu Edit is created
        MenuItem item6,item7,item8,item9;           //Menu items for Edit created
        edit.add(item6=new MenuItem("Cut"));            //Menu Items added in Menu
        edit.add(item7=new MenuItem("Copy"));
        edit.add(item8=new MenuItem("Paste"));
        edit.add(item9=new MenuItem("-"));

        Menu sub = new Menu("Special",true);                    //Menu Special is
created
        MenuItem item10,item11,item12;                  //Menu items for Special
created
        sub.add(item10=new MenuItem("First"));          //Menu Items added in Menu
        sub.add(item11=new MenuItem("Second"));
        sub.add(item12=new MenuItem("Third"));
        edit.add(sub);                              //Menu Special added in Edit Menu

        debug = new CheckboxMenuItem("Debug");
        edit.add(debug);
        test = new CheckboxMenuItem("Testing");
        edit.add(test);

        mbar.add(edit);                                 //Menu Edit added on Menu
Bar


        MyMenuHandler handler = new MyMenuHandler(this);//added a handler for
menu
        item1.addActionListener(handler);           //added an Action Listener for each
item
        item2.addActionListener(handler);
        item3.addActionListener(handler);
        item4.addActionListener(handler);
        item5.addActionListener(handler);
```

```
        item6.addActionListener(handler);
        item7.addActionListener(handler);
        item8.addActionListener(handler);
        item9.addActionListener(handler);
        item10.addActionListener(handler);
        item11.addActionListener(handler);
        item12.addActionListener(handler);
        debug.addItemListener(handler);
        test.addItemListener(handler);

        MyWindowAdapter adapter = new MyWindowAdapter(this);
        addWindowListener(adapter);
        }

public void paint(Graphics g)
        {
        g.drawString(msg,10,200);
        if(debug.getState())
                g.drawString("Debug is on...",10,220);
        else
                g.drawString("Debug is off...",10,220);

        if(test.getState())
                g.drawString("Testing is on...",10,240);
        else
                g.drawString("Testing is off...",10,240);
        }
}

class MyWindowAdapter extends WindowAdapter
{
        MenuFrame menuFrame;
        public MyWindowAdapter(MenuFrame menuFrame)
        {
        this.menuFrame=menuFrame;
        }
        public void WindowClosing(WindowEvent we)
        {
        menuFrame.dispose();
        }
}

class MyMenuHandler implements ActionListener,ItemListener
{
        MenuFrame menuFrame;
        public MyMenuHandler(MenuFrame menuFrame)
```

```
	{
	this.menuFrame=menuFrame;
	}
	public void actionPerformed(ActionEvent ae)
	{
		String msg="You selected";
		String arg=(String)ae.getActionCommand();
		if(arg.equals("New..."))
		{
		msg+=" New.";
		SampleDialog d= new SampleDialog(menuFrame,"New Dialog Box");
		d.setVisible(true);
		}
		else if(arg.equals("Open..."))
			msg+=" Open.";
		else if(arg.equals("Close"))
			msg+=" Close.";
		else if(arg.equals("Quit..."))
			msg+=" Quit.";
		else if(arg.equals("Edit"))
			msg+=" Edit.";
		else if(arg.equals("Cut"))
			msg+=" Cut.";
		else if(arg.equals("Copy"))
			msg+=" Copy.";
		else if(arg.equals("Paste"))
			msg+=" Paste.";
		else if(arg.equals("First"))
			msg+=" First.";
		else if(arg.equals("Second"))
			msg+=" Second.";
		else if(arg.equals("Third"))
			msg+=" Third.";
		else if(arg.equals("Debug"))
			msg+=" Debug.";
		else if(arg.equals("Testing"))
			msg+=" Testing.";

		menuFrame.msg=msg;
		menuFrame.repaint();
	}
	public void itemStateChanged(ItemEvent ie)
	{
	menuFrame.repaint();
	}
}
```

```
public class DialogDemo extends Applet
{
Frame f;
public void init()
        {
        f=new MenuFrame("Menu Demo");
        int width=Integer.parseInt(getParameter("width"));
        int height=Integer.parseInt(getParameter("height"));
        setSize(new Dimension(width,height));
        f.setSize(width,height);
        f.setVisible(true);
        }
public void start()
        {
        f.setVisible(true);
        }

public void stop()
        {
        f.setVisible(false);
        }
}
```

Program 24: WAP to implement Smiley face Using applet.

```
import java.awt.*;
import java.applet.*;
/*<APPLET
        CODE = Face.class
        WIDTH =250
        HEIGHT = 200 >
        <param name="a" value =10>
  <param name="b" value =20>
        </APPLET>*/
public class Face extends Applet
{
        public void paint (Graphics g)
        {
                                String a;
                                 String b;
                               String c;
                            a=getParameter("a");
                         b=getParameter("b");

            int p=Integer.parseInt(a);
```

```
        int q=Integer.parseInt(b);
        int sum=p+q;
        c=Integer.toString(sum);
        g.drawString("First value :-"+a,10,210);
        g.drawString("Second value :-"+b,10,230);
        g.drawString("Total sum :-"+c,10,250);

              g.drawLine(10,212,130,212);
              g.drawLine(10,232,130,232);
              g.drawLine(10,252,130,252);









                          Color c1=new Color(25,0,0);
                    setBackground(c1);
                    setForeground(Color.green);
                    g.drawRect(200,160,100,50);
                    g.fillRect(200,40,100,50);


                              g.drawOval(40,40,120,150);
                                    g.drawOval(57,75,30,20);
                                    g.drawOval(110,75,30,20);
                                 g.fillOval(68,81,10,10);
                                    g.fillOval(121,81,10,10);
                                    g.drawOval(85,100,30,30);
                                    g.fillArc(60,125,80,40,180,180);
                                    g.drawOval(25,92,15,30);
                                    g.drawOval(160,92,15,30);


              }
    }
```

Program 25: WAP to create Frame that display the student information.

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


public class Studentinfo
{
```

```
        static StudFrame sf;
        public static void main(String args[])
        {
                sf = new StudFrame();
        }
}


class mywindowadapter extends WindowAdapter
{
  //      StudFrame sf;
//        public mywindowadapter(StudFrame sf)
//        {
//           this.sf=sf;
//        }
//
        public void windowClosing(WindowEvent we)
        {
//              sf.setVisible(false);
                System.exit(0);
        }
}

class StudFrame extends Frame implements ActionListener,ItemListener
{
        Button b1,b2,b3,b4;

   static TextField t1,t2;
   static Choice c,c1,c2,cc;
   static Label lh,l1,l2,l3,l4,l5,l6;
   //static List lb;
   static Checkbox cb1,cb2;
   static CheckboxGroup gndr=new CheckboxGroup();

   StudFrame()
   {
        super("Student Records Form");
      //mywindowadapter mw=new mywindowadapter(this);
      addWindowListener(new mywindowadapter());
      addcontrols();
      setSize(700,550);
           setResizable(true);
      setVisible(true);
        }

   void addcontrols()
```

```
{
        setLayout(null);



           lh=new Label("Student Records");
           l1=new Label("Student ID");
        l2=new Label();
        l2.setText("Name");
        l3=new Label("Gender");
        l4=new Label("Age");
        l5=new Label("Qualification");
        l6=new Label("Course");

        t1=new TextField(8);
        t2=new TextField(8);


        cb1=new Checkbox("Male",gndr,true);
        cb2=new Checkbox("Female",gndr,false);

        cc=new Choice();
        for(int i=15;i<=80;i++)
                cc.add(Integer.toString(i));

        c=new Choice();
        c.add("Under Graduate");
        c.add("Graduate");

        c1=new Choice();
        c1.add("B.A.");
        c1.add("B.B.A.");
        c1.add("B.C.A.");
        c1.add("B.Com");
        c1.add("B.E./B.Tech");
        c1.add("B.Pharma");
        c1.add("B.Sc.");

        c2=new Choice();
        c2.add("M.B.A.");
        c2.add("M.C.A.");
        c2.add("M.E./M.Tech");

        b1=new Button("OK");
        b2=new Button("Cancel");
```

```
b3=new Button("Reset");
b4=new Button("Exit");

        lh.setBounds(100,30,100,30);
        l1.setBounds(100,60,100,30);
        l2.setBounds(100,90,100,30);
        l3.setBounds(100,120,100,30);
        l4.setBounds(100,150,100,30);
        l5.setBounds(100,180,100,30);
        l6.setBounds(100,210,100,30);

        t1.setBounds(250,60,150,20);
        t2.setBounds(250,90,150,20);
        cb1.setBounds(250,120,40,20);       cb2.setBounds(310,120,60,20);
        cc.setBounds(250,150,150,20);
        c.setBounds(250,180,150,20);
        c1.setBounds(250,210,150,20);
        c2.setBounds(250,210,150,20);

        b1.setBounds(500,90,100,35);        b2.setBounds(500,180,100,35);
        b3.setBounds(125,290,100,35);       b4.setBounds(300,290,100,35);

add(lh);
add(l1);
add(l2);
add(l3);
add(l4);
add(l5);
add(l6);
add(t1);
add(t2);
add(cb1);       add(cb2);
add(cc);
add(c);
add(c1);        c1.setVisible(true);
add(c2);        c2.setVisible(false);
        add(b1);
        add(b2);
        add(b3);
        add(b4);


b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
```

```
	c.addItemListener(this);

	}

	public void actionPerformed(ActionEvent ae)
	{
		if(ae.getSource()==b1)
		{
		subframe s=new subframe("Submission","Data entered successfully.");
		String s1=t1.getText();
			String s2=t2.getText();
		if(s1.length()==0 || s1.length()==0 )
		{
			s.setSize(300,100);
		}
		else
		{
			s.setSize(350,350);
		}
		s.setVisible(true);
		StudentInfo.sf.setEnabled(false);
		}
		else if(ae.getSource()==b2)
		{
		subframe s=new subframe("Cancellation","Data is not
accepted(Cancellation done).");
		s.setSize(300,100);
		s.setVisible(true);
		StudentInfo.sf.setEnabled(false);
		}
		else if(ae.getSource()==b3)
		{
			t1.setText("");
			t2.setText("");
			gndr.setSelectedCheckbox(cb1);
			c.select(0);
			cc.select(0);
			c1.select(0);
			c1.setVisible(true);

		}
		else if(ae.getSource()==b4)
		{
			System.exit(0);
		}
	}
```

```
	public void itemStateChanged(ItemEvent ie)
	{
		String s = c.getSelectedItem();
		if(s=="Under Graduate")
		{
			c1.setVisible(true);
			c2.setVisible(false);
		}
		if(s=="Graduate")
		{
			c1.setVisible(false);
			c2.setVisible(true);
		}
	}

}

class subwindowadapter extends WindowAdapter
{
	subframe subf;
	public subwindowadapter(subframe subf)
	{
	   this.subf=subf;
	}

	public void windowClosing(WindowEvent we)
	{
	StudentInfo.sf.setEnabled(true);
	subf.setVisible(false);
	}
}

class subframe extends Frame implements ActionListener
{
	Button bsubok=new Button("OK");

	subframe()	{}
	subframe(String title) {}

	subframe(String title,String msg)
	{
		super(title);

		String s1=StudFrame.t1.getText();
		String s2=StudFrame.t2.getText();
```

```
        subwindowadapter sw=new subwindowadapter(this);
        addWindowListener(sw);

        if(title=="Cancellation")
                {
//                              StudFrame.t1.setText("");
//                              StudFrame.t2.setText("");
                                setLayout(new FlowLayout(FlowLayout.CENTER));
                                add(new Label(msg));
                                add(bsubok);
                        bsubok.addActionListener(this);
                }
                else
                {
                        if(s1.length()==0)
                        {
                                //resize(300,100);
                                setLayout(new FlowLayout(FlowLayout.CENTER));
                        add(new Label("Please fill in Student Name."));
                        add(bsubok);
                        bsubok.addActionListener(this);
                }
                else if(s2.length()==0)
                {
                        setLayout(new FlowLayout(FlowLayout.CENTER));
                        add(new Label("Please fill in Student Roll Number."));
                        add(bsubok);
                        bsubok.addActionListener(this);
                }
                else
                        //add(new Label(StudFrame.t1.getText() +", Student ID:
"+StudFrame.t2.getText()+" Accepted.",Label.CENTER));
                {

                        setLayout(null);


                        Label lhl,ll1,ll2,ll3,ll4,ll5,ll6,la1,la2,la3,la4,la5,la6,ltl;
                        lhl=new Label("Your data is:");
                        ll1=new Label("Student ID");
                        ll2=new Label();
                        ll2.setText("Name");
                        ll3=new Label("Gender");
                        ll4=new Label("Age");
                        ll5=new Label("Qualification");
                        ll6=new Label("Course");
```

```
                    ltl=new Label(msg);

                    la1=new Label(StudFrame.t1.getText());
                    la2=new Label();
                    la2.setText(StudFrame.t2.getText());
                    la3=new
Label(StudFrame.gndr.getSelectedCheckbox().getLabel());
                    la4=new Label(StudFrame.cc.getSelectedItem());
                    la5=new Label(StudFrame.c.getSelectedItem());
                    if(StudFrame.c.getSelectedItem()=="Under Graduate")
                        la6=new Label(StudFrame.c1.getSelectedItem());
                    else
                        la6=new Label(StudFrame.c2.getSelectedItem());

                        add(lhl);
                        add(ll1);
                    add(ll2);
                       add(ll3);
                    add(ll4);
                    add(ll5);
                    add(ll6);
                    add(la1);
                    add(la2);
                       add(la3);
                    add(la4);
                    add(la5);
                    add(la6);
                    add(ltl);add(bsubok);

                    lhl.setBounds(50,30,100,30);
                    ll1.setBounds(50,60,100,30);
                        ll2.setBounds(50,90,100,30);
                        ll3.setBounds(50,120,100,30);
                        ll4.setBounds(50,150,100,30);
                        ll5.setBounds(50,180,100,30);
                        ll6.setBounds(50,210,100,30);
                        la1.setBounds(200,60,100,30);
                        la2.setBounds(200,90,100,30);
                        la3.setBounds(200,120,100,30);
                        la4.setBounds(200,150,100,30);
                        la5.setBounds(200,180,100,30);
                        la6.setBounds(200,210,100,30);
                        ltl.setBounds(75,240,200,30);

                        bsubok.setBounds(100,280,100,30);
                        bsubok.addActionListener(this);
```

```
                }
        }


    }
    public void actionPerformed(ActionEvent ae)
        {
                if(ae.getSource()==bsubok)
                {
                StudentInfo.sf.setEnabled(true);
                setVisible(false);
                }

        }
}
```

Program 26: WAP to implement System Clock.

```
  public class AWT1
{
        public static void main(String args[])
        {
                MyFrame mf = new MyFrame();
        }
}
import java.util.*;

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

class MyFrame extends Frame implements ItemListener, ActionListener
{
        Choice c1;
        Button b1;

        Checkbox cb1, cb2, cb3, cb4;
        CheckboxGroup cbg1, cbg2;

        Label lblTime;

        TimeThread tt;

        MyFrame()
```

```
{
        super("Sample Java Frame");
        addWindowListener(new MyWindowAdapter());
        setSize(400,300);
        addControls();
        setVisible(true);


        tt = new TimeThread(this);
}

private void addControls()
{
        setLayout(new FlowLayout());

        lblTime = new Label("System Time Here");
        add(lblTime);

        c1 = new Choice();
        c1.add("Ajmer");
        c1.add("Jaipur");
        c1.add("Alwar");
        c1.add("Nasirabad");
        c1.add("Bikaner");
        c1.add("Kishangarh");
        c1.add("Beawar");
        c1.add("Bundi");
        c1.add("Kota");
        c1.add("Nagur");
        c1.add("Jodhpur");
        c1.add("Pali");

        c1.addItemListener(this);

        add(c1);

        b1 = new Button("Click Me");
        b1.addActionListener(this);
        add(b1);

        cbg1 = new CheckboxGroup();
        cbg2 = new CheckboxGroup();

        cb1 = new Checkbox("DOS",true,cbg1);
        cb2 = new Checkbox("Windows",cbg1,true);
```

```
		cb3 = new Checkbox("Linux",cbg2,false);
		cb4 = new Checkbox("Unix",cbg2,false);

		add(cb1);
		add(cb2);
		add(cb3);
		add(cb4);

		cb1.addItemListener(this);
		cb2.addItemListener(this);
		cb3.addItemListener(this);
		cb4.addItemListener(this);

	}

	public void itemStateChanged(ItemEvent ie)
	{
		if (ie.getSource() == c1)
		{
			if (c1.getSelectedIndex() != -1)
			{
				System.out.println(c1.getSelectedItem());
				System.out.println(c1.getSelectedIndex());
			}
		}
		else if (ie.getSource() == cb1)
		{

			c1.setEnabled(cb1.getState());
		}
		else if (ie.getSource() == cb3 || ie.getSource() == cb4)
		{
			System.out.println("Item Selected : " +
cbg2.getSelectedCheckbox().getLabel());
		}
	}

	public void actionPerformed(ActionEvent ae)
	{
		if (ae.getSource() == b1)
		{
			int i;
			for (i=0;i<c1.getItemCount();i++)
				System.out.println(c1.getItem(i));
		}
	}
```

```
	String getTime()
	{
		Calendar cal = new GregorianCalendar();

		int hour;
		int minute;
		int second;

		hour = cal.get(Calendar.HOUR);
		minute = cal.get(Calendar.MINUTE);
		second = cal.get(Calendar.SECOND);

		String timeStr = hour + ":" + minute + ":" + second;
		return timeStr;
	}
}
class TimeThread extends Thread
{
	MyFrame mf;
	TimeThread(MyFrame mf)
	{
		this.mf = mf;
		start();
	}

	public void run()
	{
		while(true)
		{
			mf.lblTime.setText(mf.getTime());
			try
			{
				Thread.sleep(970);
			}
			catch (InterruptedException e1)
			{
			}
		}
	}
}

import java.awt.event.*;
class MyWindowAdapter extends WindowAdapter
{
	public void windowClosing(WindowEvent we)
```

```
        {
                System.exit(0);
        }
}
```

Program 27: WAP to implement Interthread Communication.

```
class Consumer implements Runnable
{
        Counter counter;
        Thread t;
        Consumer(Counter counter)
        {
                this.counter = counter;
                t = new Thread(this);
                t.start();
        }

        public void run()
        {
                int i;
                while ((i = counter.getValue()) < 50);

        }
}
class Counter
{
        int value;
        boolean valueSet;
        Counter()
        {
                valueSet = false;
        }
        synchronized void setValue(int value)
        {
                try
                {
                        if (valueSet == true)
                        {
                                wait();
                        }
                        this.value = value;
                        System.out.println("Value produced : " + value);
                        valueSet = true;
                        notify();
                }
                catch (InterruptedException e1){}
```

```
	}

	synchronized int getValue()
	{
		try
		{

			if (valueSet == false)
			{
				wait();
			}
			System.out.println("Value consumed : " + value);
			valueSet = false;
			notify();
		}
		catch (InterruptedException e1){}
		return value;
	}
}
class Producer implements Runnable
{
	Counter counter;
	Thread t;
	Producer(Counter counter)
	{
		this.counter = counter;
		t = new Thread(this);
		t.start();
	}

	public void run()
	{
		int i;
		for (i=1;i<=50;i++)
			counter.setValue(i);
	}
}


public class InterThreadComm
{
	public static void main(String args[])
	{
		Counter counter = new Counter();
		Producer p = new Producer(counter);
		Consumer c = new Consumer(counter);
```

```
        }
    }
```