**WEEK-3a**

import java.applet.Applet;

import java.awt.\*;

public class Week3aMyApplet extends Applet

{

public void init()

{

setBackground(Color.yellow);

}

public void paint(Graphics g)

{

g.drawString("Hello World",100,100);

}

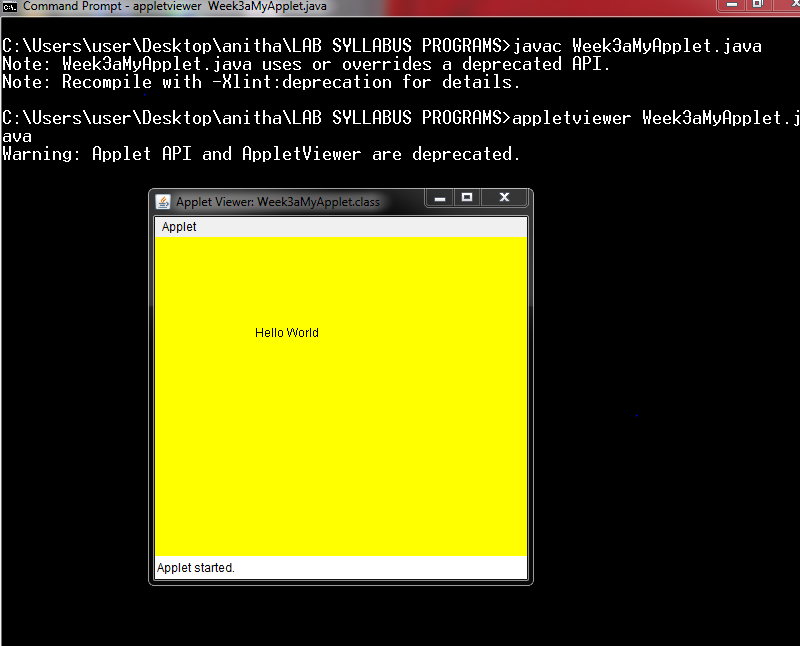
}

/\*

<applet code="Week3aMyApplet.class" width=300

height=300></applet>

\*/



**WEEK-3b**

import java.applet.Applet;

import java.awt.\*;

import java.awt.event.\*;

public class Week3bFactorial extends Applet

implements ActionListener

{

TextField t1,t2;

Label l1,l2,l3;

Button b;

int fact=1,n,i;

Week3bFactorial e;

public void init()

{

e=this;

t1=new TextField(10);

t2=new TextField(10);

l1=new Label("factorial of a number");

l2=new Label("enter a number");

l3=new Label("result");

b=new Button("compute");

add(l1);

add(l2);

add(t1);

add(b);

add(l3);

add(t2);

b.addActionListener(e);

}

public void actionPerformed(ActionEvent ae)

{

String str=t1.getText();

n=Integer.parseInt(str);

for(i=1;i<=n;i++)

{

fact=fact\*i;

}

String msg=""+fact;

t2.setText(msg);

fact=1;

}

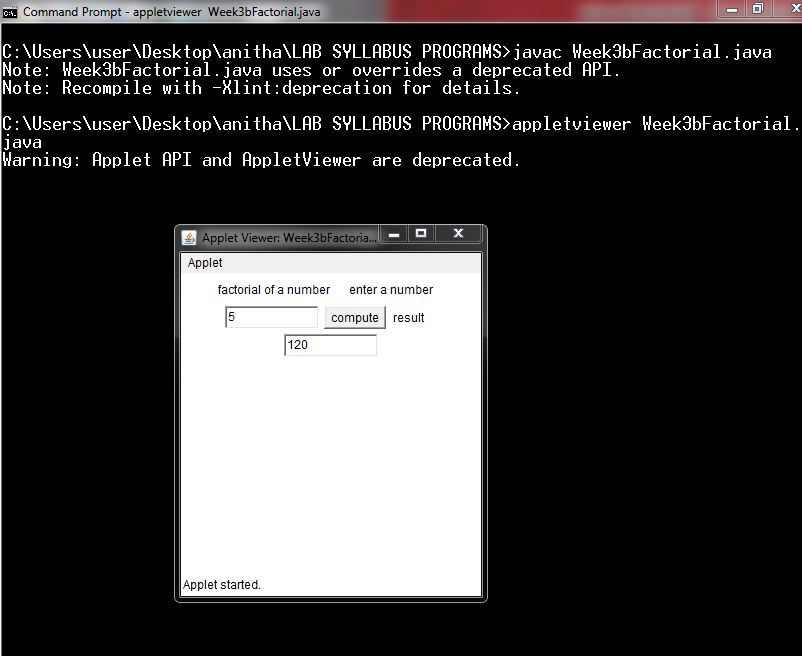
}

/\*

<applet code="Week3bFactorial.class" width=300

height=300></applet>

\*/



**WEEK-5**

import java.util.Random;

class Thread1 extends Thread

{

public void run()

{

try

{

int num=0;

Random r=new Random();

for(int i=0;i<10;i++)

{

num=r.nextInt(100);

if(num%2==0)

{

Even e=new Even(num);

e.start();

}

else

{

Odd o=new Odd(num);

o.start();

}

Thread.sleep(1000);

}

}

catch(InterruptedException e)

{

System.out.println("handled");

}

}

}

class Even extends Thread

{

int x;

Even(int x)

{

this.x=x;

}

public void run()

{

System.out.println("number is even:"+x+"its square is:"+(x\*x));

}

}

class Odd extends Thread

{

int x;

Odd(int x)

{

this.x=x;

}

public void run()

{

System.out.println("number is odd:"+x+"its cube is:"+(x\*x\*x));

}

}

class Week5MultipleThreadDemo

{

public static void main(String args[])

{

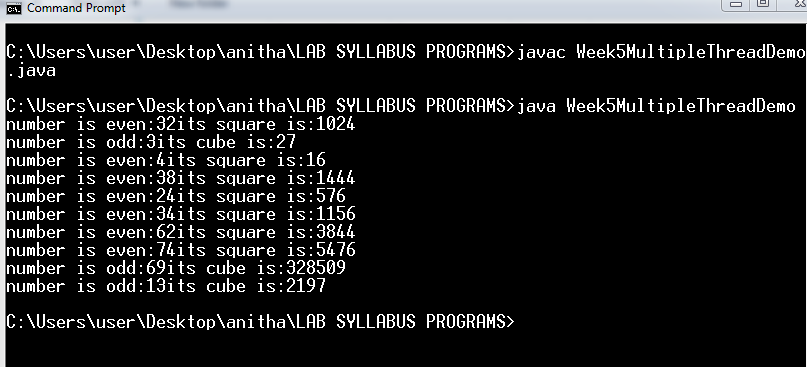
Thread1 t=new Thread1();

t.start();

}

}

OUTPUT:



**WEEK-8**

abstract class Shape

{

int dim1;

int dim2;

Shape(int a, int b)

{

dim1 = a;

dim2 = b;

}

Shape(int a)

{

dim1 = a;

}

abstract double printarea(); // printarea is now

an abstract method

}

class Rectangle extends Shape

{

Rectangle (int a, int b)

{ super (a, b);

}

double printarea () // override area for

rectangle

{

System.out.println ("Inside Rectangle.");

return dim1 \* dim2;

}

}

class Triangle extends Shape

{

Triangle (int a, int b)

{

super (a, b);

}

double printarea() // override area for triangle

{

System.out.println ("Inside Triangle.");

return dim1 \* dim2 / 2;

}

}

class Circle extends Shape

{

Circle(int a)

{

super(a);

}

double printarea()

{

System.out.println("Inside Circle.");

return 3.14\*dim1\*dim1;

}

}

class Week8AbstractAreas

{

public static void main(String args[])

{

Rectangle r = new Rectangle(9, 5);

Triangle t = new Triangle(10, 8);

Circle c=new Circle(5);

System.out.println("Area of rectangle is :" +

r.printarea());

System.out.println("Area of triangle is :" +

t.printarea());

System.out.println("Area of circle is :" +

c.printarea());

}}

OUTPUT: