JAVA ALL PROGRAM

LAB 1

Program: 1

// make a simple java program

```
class Lines
{
public static void main(String args[])
{
System.out.println("hello.... \n how are you");
}
}
```

Output:-

Hello……

How are you

Program: 2

```
//make a program for Addition


class Add
{
	public static void main(String s[])
	{
		int a,b,c;
		a=Integer.parseInt(s[0]);
		b=Integer.parseInt(s[1]);
		c=a+b;
		System.out.println(c);
	}
}
```

Output:-

```
Java Add 10 20
30
```

Program: 3

```
//division program


class Div
{
	public static void main(String args[])
	{
	float a=12,b=5,c;
	c=a/b;
	System.out.println("Division of " +a+ " and " +b+ " is " +c);
	}
}
```

Output:-

```
Division of 12.0 and 5.0 is 2.4
```

Program: 4

```
//w.a.p to find grade of given value
class Grade
{
        public static void main(String s[])
        {
                for(int i=0;i<s.length;i++)
               int n=Integer.parseInt(s[i]);
                if(n>=80)
                  System.out.println("grad A");
                if (n>= 60 && n<80)
                  System.out.println("grad B");
                  else if(n>=50 && n<60)
                          System.out.println("grad C");
                  else if(n>=40 && n<50)
                          System.out.println("grad D");
                  else if(n>=30 && n<40)
                          System.out.println("grad E");
```

```
            else if(n>=100)
                System.out.println("Invalid entry");


else
                System.out.println("grad F");
        }
 }
```

Output:-

Java Grade 77

Grade B

Lab 2

Program: 1

```
// intro to more than one classes
class A1
{
 int a;
}
Public class B1
{
 public static void main(String s[])
 {
  A ob;
  ob=new A();
  System.out.println(ob.a);
 }
}
```

Output:->java A1

Exception in thread "main" java.lang.NoSuchMethodError: main

Program: 2

//API programs

```
class A
{
	int a;
}

public class B
{
	public static void main(String s[])
	{
		A ob;
		ob=new A();
		System.out.println(ob.a);
	}
}
```

Output:-

0

Program: 3

```
//API perform with 3 classes


class A
{
        int a;
}


class B
{
        int b;
}


public class C_main
{
        public static void main(String s[])
        {
                int c;
                A ob1;
                ob1=new A();
                B ob2;
                ob2=new B();
                ob1.a= 10;
                ob2.b=20;
```

```
            c=ob1.a+ob2.b;
            System.out.println("Sum of "  +ob1.a + " and " +ob2.b+ " is " +c);
        }
}
```

Output:-

Sum of 10 and 20 is 30

Program: 4

```
// three classes A,B,C. 2 having integers. assign value to int through void main.
//find max and min.


class A2
{
	int a;
}


class B2
{
	int b;
}


public class C1_main
{
	public static void main(String s[])
	{
		A2 ob1=new A2();
		ob1.a=5;
		B2 ob2=new B2();
		ob2.b=10;
		if(ob1.a>ob2.b)
		{
```

```
        System.out.println("Integer " +ob1.a+ " is greater" +ob1.a+ " and " +ob2.b );
                }
                else if(ob1.a==ob2.b)
                {
                        System.out.println("Integer a and b are equal" +ob1.a+ " and "
+ob2.b );
                }
                else
                {
                System.out.println("Integer " +ob2.b+" is greater from " +ob1.a+ "
and " +ob2.b );
                }
        }
}
```

Output:-

Integer 10 is greater from 10 and 5

Program: 5

```
//2 classes.rectangle and triangle.rectangle has l,b.triangle has base and height.
//3rd class area initialize values using command line arguments.

class rectangle
{
        int length,breath;
}

class triangle
{
        int base,heigth;
}

class Area_main
{
        public static void main(String s[])
        {
                float A1,A2;
                rectangle ob=new rectangle();
                triangle ob1=new triangle();
                ob.length=Integer.parseInt(s[0]);
                ob.breath=Integer.parseInt(s[1]);
                ob1.base=Integer.parseInt(s[2]);
```

```
            ob1.heigth=Integer.parseInt(s[3]);

            A1=ob.length*ob.breath;

            A2=(float)(ob1.base*ob1.heigth)/2;

            System.out.println("Area of Rectangle with length " +ob.length+ "
and width " +ob.breath+ " is " +A1);

            System.out.println("Area of Triangle with base " +ob1.base+ " and
heigth " +ob1.heigth+ " is " +A2);

        }

}
```

Output:-

Java Area_main 10 20 30 40

Area of rectangle with length 10 and width 20 is 200.0

Area of triangle with base 30 and height 40 is 600.0

Program: 6

//Create class rectangle.it has three int parameters. height,width,length.

//create other class volume which holds void main.Enter data into height,width and length using command line arguments.

//Print volume.

```
class Rectangle
{
        int height,width,length;
}


class volume_main
{
        public static void main(String s[])
        {
                int v;
                Rectangle ob=new Rectangle();
                ob.height=Integer.parseInt(s[0]);
                ob.width=Integer.parseInt(s[1]);
                ob.length=Integer.parseInt(s[2]);
                v=ob.height*ob.width*ob.length;
```

```
            System.out.println("Volume of Rectangle with height " +ob.height+ "
width " +ob.width+ " length " +ob.length+ " is " +v);

        }

}
```

Output:-

Java volume_main 1 2 3

Volume of rectangle with height 1 width 2 and length 3 is 6

Lab 3

Program: 1

```
/*
create 2 classes rectangle and triangle.retc has length,breath and height.
triangle has base and altitude.all integers.
create another class area that has 2 static methods area_rect  and area_tri.
both has 2 arguments.creat another that has void main.cal area.
*/
class rect
{
	int length,breath,heigth;
}

class triangle
{
	int base,altitude;
}


class area
{
	static void area_rect(int a,int b)
```

```
        {       int c;
                c=a*b;
                System.out.println("Area of rectangle is: " +c);
        }
        static void area_tri(int a,int b)
        {       float d;
                d=(float)(a*b)/2;
                System.out.println("Area of triangle is: " +d);
        }
}

public class Area_main
{
        public static void main(String s[])
        {
                rect ob=new rect();
                ob.length=10;
                ob.breath=20;
                triangle t=new triangle();
                t.base=10;
                t.altitude=20;
                area.area_rect(ob.length,ob.breath);
                area.area_tri(t.base,t.altitude);
```

```
    }
}
```

Output:-

Area of rectangle is 200

Area of triangle is 100.0

Program: 2

```
// create a class which has 2 methods hello() and good()
//hello prints statement giving hello msg and good prints
//good morning.create another class that holds void main() and calls 2
//methods.


class fine
{
	static void hello()
	{
		System.out.println("Hello ...How are you");
	}
	static void good()
	{
		System.out.println("good morning");
	}
}



public class good_main
{
```

```
    public static void main(String s[])
    {
        fine.hello();
        fine.good();
    }
}
```

Output:-

Hello…how are you….

good morning

Program: 3

```
// create a class which has 2 methods hello() and good()
//hello prints statement giving hello msg and good prints
//good morning.create another class that holds void main() and calls 2
//methods.


class hello
{
	void hello()
	{
		System.out.println("Hello ...How are you");
	}
	void good()
	{
		System.out.println("good morning");
	}
}
```

```
public class hello_main
{
	public static void main(String s[])
	{
		hello ob;
		ob=new hello();
		ob.hello();
		ob.good();
	}
}
```

Output:-

Hello…how are you

Good morning

Program: 4

```
/* write a program using Arguments
*/
//created by haresh vaviya
class sum
{
	int c;
	void add(int a,int b)
	{
		c=a+b;
		System.out.println("addition value is:  " +c);
	}
	void sub(int a,int b)
	{
		c=a-b;
		System.out.println("Subtraction is :" +c);
	}
	void mul(int a,int b)
	{
		c=a*b;
		System.out.println("Multiplication is is :" +c);
	}
	}
```

```
public class maths_main
{
	public static void main(String s[])
	{
		int a,b;
		sum ob=new sum();
		a=15;
		b=10;
		ob.add(15,10);
		ob.sub(15,10);
		ob.mul(15,10);
	}
}
```

Output:-

Addition value is:25

Subtraction is:5

Multiplication is:125

Program: 5

/*create a class that gives a personal info. that has foll. methods name,age,en_nos,dept,address.make 1st 3 methods as static and other 3 as it is. create another class that has void main and calls all this methods.
*/

```
class Name
{
	static void name()
	{
		System.out.println("Name:Dhruv Joshi");
	}
}

class Age
{
	static void age()
	{
		System.out.println("Age: 20");

	}
}
```

```
class En_nos
{
	static void en_no()
	{
		System.out.println("Nmbr is:090010107052");
	}
}

class Dept
{
	void dept()
	{
		System.out.println("Dept: CP");
	}
}

class Address
{
	void address()
	{
		System.out.println("ADIT hostel");
	}
```

```
}


public class personal_info_main
{
        public static void main(String s[])
        {
                Dept ob=new Dept();
                Address ob1=new Address();
                Name.name();
                Age.age();
                En_nos.en_no();
                ob.dept();
                ob1.address();
        }
}
```

Output:-

Name: Adit joshi

Age:20

Number is:9428599986

Dept: CP

ADIT hoste

Lab 4

Program: 1

```
/* create class with 2 variables.
create another class with method compare.
create third class which contains main.
initialize 2 obj of class A and pass value into parameter
using class. */


class variables
{
        int a,b;
}



class compare
{
        void compare(int a1,int b1,int a2,int b2)
        {
                if(a1==a2 && b1==b2)
                {
                        System.out.println("Both Numbers are equal" );
```

```
            }
            else
            {
                System.out.println("Both numbers are not equal");
            }
        }
}

class compare_main
{
        public static void main(String s[])
        {
            variables ob1,ob2;
            ob1=new variables();
            ob2=new variables();
            ob1.a=Integer.parseInt(s[0]);
            ob1.b=Integer.parseInt(s[1]);
            ob2.a=Integer.parseInt(s[2]);
            ob2.b=Integer.parseInt(s[3]);
            compare b;
            b=new compare();
            b.compare(ob1.a,ob1.b,ob2.a,ob2.b);
        }
```

}

Output:-

java compare_main 1 1 1 1

Both numbers are equal

Program: 2

```
/* create class with 2 variables.
create another class with method compare.
create third class which contains main.
initialize 2 obj of class A and pass value into parameter
using cla. */
class variables
{
	int a,b;
	void compare(int a1,int b1)
	{

		if(a==a1 && b==b1)
		{
			System.out.println("Both are equal");
		}
		else
		{
			System.out.println("Both are not equal");
		}
	}

}
```

```
public class compare_main1
{
	public static void main(String s[])
	{
		variables ob1,ob2;
		ob1=new variables();
		ob2=new variables();
		ob1.a=10;
		ob1.b=5;
		ob2.a=10;
		ob2.b=5;
		ob1.compare(ob2.a,ob2.b);
	}
}
```

Output:-

Both are equal

Program: 3

```
/* create 3 classes.square and cube.
*/


class variable
{
	int a;
}


class Square
{
	static void square(int a)
	{
	System.out.println("Square is :" +a*a);
	}
}


class Cube
{
	static cube(int a)
	{
		System.out.println("Cube is :" +a*a*a);
	}
```

```
}


class math_main
{
        public static void main(String s[])
        {
                variable ob=new variable();
                ob.a=Integer.parseInt(s[0]);
                Square.square(ob.a);
                Cube.cube(ob.a);
        }
}
```

Output:-

```
Java math_main 4
Square is: 16
Cube is: 64
```

Program: 4

```
/* create 3 classes.square and cube.
*/


class variable
{
	int a;
	void square()
	{
		System.out.println("Square is :" +a*a);
	}
	void cube()
	{
		System.out.println("Cube is :" +a*a*a);
	}
}


class math_main1
{
	public static void main(String s[])
	{
		variable ob=new variable();
		ob.a=Integer.parseInt(s[0]);
```

```
            ob.square();
            ob.cube();
        }
}
```

Output:-

Java math_main 5

Square is: 25

Cube is: 125

Lab 5

Program: 1

```
/*
 constructor ADD.
*/
//Created on 09-08-2011
//by haresh vaviya
class assign
{
	int a,b;
	assign(int x,int y)
	{
		a=x;
		b=y;
	}

	void add()
	{
		int c;
		c=a+b;
		System.out.println("Addition is :" +c);
```

```
    }
}


public class add_main
{
    public static void main(String s[])
    {
        assign aa=new assign(5,10);
        aa.add();

    }
}
```

Output:-

Addition is 15

Program: 2

```
/*      Create a class that has variable a.
        Initialize using constructor.
*/
//by haresh
class cons
{       int a;
        cons(int x)
        {
        a=x;
        }
}
public class c_main
{
        public static void main(String s[])
        {
        cons c=new cons(15);
        System.out.println("Default value set is " +c.a);
        }
}
```

Output:-

```
Default value set is 15
```

Program: 3

```
// Construtor
class sample
{
	sample()
	{
		System.out.println("haresh");
	}
}

public class constructor_main
{
	public static void main(String s[])
	{
		sample ss=new sample();
	}
}
```

Output:-

haresh

Program: 4

```
// multiplication and subtraction with constructor
class assign
{
	static int a,b,c,d;
	assign(int x,int y)
	{
		a=x;
		b=y;
	}

	static void sub()
	{
		c=a-b;
		System.out.println("Subtraction is :" +c);
	}

	static void mul()
	{

		d=a*b;
		System.out.println("Multiplication is :" +d);
	}
```

```
}
public class mt1_main
{
        public static void main(String s[])
        {
                assign aa=new assign(5,10);
                assign.sub();
                assign.mul();
                assign ab=new assign(10,3);
                assign.sub();

        }
}
```

Output:-

```
Subtraction is: -5
Multiplication is: 50
Subtraction is: 7
```

Program: 5

```
// multiplication and subtraction with constructor


class assign
{
        int a,b;
        assign(int x,int y)
        {
                a=x;
                b=y;
        }

        void sub()
        {
                int c;
                c=a-b;
                System.out.println("Subtraction is :" +c);
        }

        void mul()
        {
                int d;
                d=a*b;
```

```
            System.out.println("Multiplication is :" +d);
        }
}



public class mt_main
{
        public static void main(String s[])
        {
                assign aa=new assign(5,10);
                aa.sub();
                aa.mul();
        }
}
```

Output:-

Subtraction is -5

Multiplication is 50

Lab 6

Program: 1

```
/*
        constructor implicitly called.
*/


class A
{
        A()
        {
                System.out.println("This is A");
        }
}

class B extends A
{
        B()
        {
                System.out.println("This is B");
        }
}
```

```
class C
{
        public static void main(String s[])
        {
                B ob=new B();
        }
}
```

Output:-

```
This is A
This is B
```

Program: 2

```
/*
        super keyword
*/


class A1
{
        int i;
}



class B1 extends A1
{

        int i=5;
        B1()
        {
                super.i=10;


        }

              void print()
                {
```

```
                System.out.println("Super value is:" +super.i);
                System.out.println("normal value is:" +i);
        }
}
public class C1
{
    public static void main(String s[])
    {
        B1 ob=new B1();
        ob.print();
    }
}
```

Output:-

```
Super value is: 10
normal value is: 5
```

Program: 3

```
/*
        method overriding
*/



class A2
{
        void hello()
        {
                System.out.println("A");
        }
}

class B2 extends A2
{
        void hello()
        {
                System.out.println("B");
                super.hello();
        }
}
```

```
class C2
{
	public static void main(String s[])
	{
		B2 ob=new B2();
		ob.hello();
	}
}
```

Output:-

```
B
A
```

Program: 4

```
/* 1 level Inheritance.....
create a superclass A in which there a integer i,
and create a class B Extends Aand have integer j,
create method add in B which adds this 2 integers.
and prints them.create another class with void main and
initialize them usind command line arguments.
*/


class A
	{
		int i;
	}
class B extends A
	{
		int j;
		public int add()
		{
			int c=j+i;
			return c;
		}
	}
```

```
public class C_main
{
	public static void main(String s[])
	{
		B ob=new B();
		ob.i=Integer.parseInt(s[0]);
		ob.j=Integer.parseInt(s[1]);
		int m=ob.add();
		System.out.println("Addition is :" +m);
	}
}
```

Output:-

Javac C_main 10 20

Addition is: 30

Program: 5

```
/*
create a class rectangle having 3 variables l,b,h.
create another class area that contain method rect_area
which calculates area of rectangle and prints it.
create another class called volume which contain
rect_vol that calculates volume of rect and prints it.
both area and volume uses rectangle as super class.
create another class that contains main.

*/



class rectangle
{
        int l,b,h;
}



class area extends rectangle
{
        void rect_area()
        {
```

```
            System.out.println("Area is:" +l*b);
        }
}

class volume extends rectangle
{
        void rect_vol()
        {
            System.out.println("Volume is: " +l*b*h);
        }
}


class rect_main
{
        public static void main(String s[])
        {
            rectangle r=new rectangle();
            r.l=Integer.parseInt(s[0]);
            r.b=Integer.parseInt(s[1]);
            r.h=Integer.parseInt(s[2]);
            area ob=new area();
            ob.rect_area();
```

```
            volume v=new volume();
            v.rect_vol();
        }
}
```

Output:-

java rect_main 1 2 3

area is: 2

volume is: 6

Lab 7

Program: 1

```
/* INTERFACE
 create interface that has 3 methods.addition,subtraction,multiplication.
create another class which imlements interface overides all 3 methods.
create another class containnig void main.create object and call 3 methods.
*/
interface maths
{
        void add(int a,int b);
        void sub(int a,int b);
        void mul(int a,int b);
}

class A implements maths
{
        void add(int a,int b)
        {
                System.out.println(a+b);
        }
        void sub(int a,int b)
        {
```

```
            System.out.println(a-b);
        }
        void mul(int a,int b)
        {
            System.out.println(a*b);
        }
}

class B_main
{
        public static void main(String s[])
        {
            A ob=new A();
            ob.add(10,5);
            ob.sub(10,5);
            ob.mul(10,5);
        }
}
```

Output:-

```
15
5
50
```

Program: 2

//Dynamic method dispatch

class A

```
{
        void print()
        {
                System.out.println("hello, i am in A");
        }
}

class B extends A
{
        void print()
        {
                System.out.println("hello, i am in B");
        }
}



class C extends A
{
        void print()
        {    System.out.println("hello, i am in C");
        }
}
```

```
class D_main
{
	public static void main(String s[])
	{
		A ob=new A();
		B ob1=new B();
		C ob2=new C();
		ob.print();
		ob=ob1;
		ob.print();
		ob=ob2;
		ob.print();
	}
}
```

Output:-

```
Hello, I am in A
Hello, I am in B
Hello, I am in C
```

Program: 3

```
//created by haresh vaviya
//on 20-9-11
//exception handling using try and catch
class Sampl_main
{
	public static void main(String s[])
	{
		try
```

```
            {
                int a,x,y;
                x=4;
                y=0;
                a=x/y;
            }
            catch(Exception e)
            {
                System.out.println("error occured");
            }
        }
}
```

Output:- Error occurred

Program: 4

```
//exception handling
class Sampl_main
{
    public static void main(String s[])
    {
```

```
        try
        {
            int a,x,y;
            x=4;
            y=0;
            System.out.println("hello1");
            a=x/y;
            System.out.println("hello2");
        }

        catch(Exception e)
        {
            System.out.println("exception occured");
        }
    }
}
```

Output:-

Hello1

exception occurred

Lab 8

Program: 1

```
//create a program in which there are 3 threads 1 thread prints all even no.s
//between 0 to 1000 and 2nd thread prints all odds nos. between 0 to 1000 and thrid
//thread prints all divisible by 5 from 0 to 1000

class counter implements Runnable

{
```

```
Thread t1;
int i,x;
counter(int y)
{
        x=y;
        t1=new Thread(this,"adit");
        t1.start();
}
public void run()
{
        if(x==1)
        {
                for(i=0;i<=1000;i++)
                {
                        if(i%2==0)
                        {
                                System.out.println(i);
                        }
                }
        }
        else if(x==2)
        {
                for(i=0;i<=1000;i++)
```

```
                {
                        if(i%2!=0)
                        {
                                System.out.println(i);
                        }
                }
        }
        else
        {
                for(i=0;i<=1000;i++)
                {
                        if(i%5==0)
                        {
                                System.out.println(i);
                        }
                }

        }
    }

}
class counter_main
{
```

```
    public static void main(String s[])
    {
        counter c1,c2,c3;
        c1=new counter(1);
        c2=new counter(2);
        c3=new counter(3);
    }
}
```

Output:-

405(+2)

.

.

.

997

999

Program: 2

```
//multi threding  operation in java


class sample implements Runnable
{
    Thread t1;
    int x;
```

```
sample(int y)
{
        t1=new Thread(this,"adit");
        t1.start();
        x=y;
}
public void run()
{
        int i;
        if(x==1)
        {

                for(i=1;i<=10;i++)
                {
                        try
                        {
                                Thread.sleep(200);
                                System.out.println(i);
                                System.out.println("hello, i am thread 1");
                        }
                        catch(Exception e)
                        {
                                System.out.println("from1");
```

```
                }
            }
        }
        else if(x==2)
        {

            for(i=1;i<=10;i++)
            {
                try
                {
                    Thread.sleep(500);
                    System.out.println(i);
                    System.out.println("hello, iam thread 2");
                }
                catch(Exception e)
                {
                    System.out.println("from 2");
                }
            }
        }
    }
}
```

```
class sample_main
{
	public static void main(String s[])
	{
		sample s2,s1;
		s1=new sample(1);
		s2=new sample(2);
		System.out.println("process dies");
	}
}
```

Output:-

```
process dies
1
hello, i am thread 1
2
hello, i am thread 1
1
```

hello, iam thread 2

3

hello, i am thread 1

4

hello, i am thread 1

5

hello, i am thread 1

2

hello, iam thread 2

6

hello, i am thread 1

7

hello, i am thread 1

3

hello, iam thread 2

8

hello, i am thread 1

9

hello, i am thread 1

10

hello, i am thread 1

4

hello, iam thread 2

5

hello, iam thread 2

6

hello, iam thread 2

7

hello, iam thread 2

8

hello, iam thread 2

9

hello, iam thread 2

10

hello, iam thread 2

Lab 9

Program: 1

```
//this program created by haresh vaviya

//sample program for thread programming.

//write a prog in which there are 2 threads. 1 thread creates multiples of 3
//between 0 to 10 and another thread creates multiple of 5 between 0 to 10.

//1st thread generates data at 200 ms and another generates data at 500 ms.
```

```
class Multiple implements Runnable
{
        Thread t;
        int a;
        Multiple(int b)
        {
                t=new Thread(this,"adit");
                t.start();
                a=b;
        }
        public void run()
        {
                int i;
                if(a==1)
                {
                        for(i=0;i<=10;i+=3)
                        {
                                try
                                {
                                        Thread.sleep(200);
                                        System.out.println("thread 1");
```

```
                }
                catch(Exception e)
                {
                        System.out.println("error");
                }
        }
}
else
{
        for(i=0;i<=10;i+=5)
        {
                try
                {
                        Thread.sleep(500);
                        System.out.println("thread 2");
                }
                catch(Exception e)
                {
                        System.out.println("error");
                }

        }
}
```

```
        }
}

class Multiple_main
{
        public static void main(String s[])
        {
                Multiple m1,m2;
                m1=new Multiple(1);
                m2=new Multiple(2);
        }
}
```

Output:-

```
Thread 1
Thread 1
Thread 2
Thread 1
Thread 1
```

Thread 2

Thread 2

Program: 2

//producer and consumer problem using threading

```
import java.util.Random;
class Unit
{
        int cake;
```

```
    void produce()
    {
        if(cake==50)
        {
            System.out.println("rack is full");
        }
        else
        {
            cake=cake+1;
            System.out.println("cake in rack are:"+cake);
        }
    }

void consume()
    {
        if(cake==0)
        {
            System.out.println("rack is empty");
        }
        else
        {
            cake=cake-1;
            System.out.println("total cake in rack after consumed:"+cake);
```

```
            }
        }

}

class P_C implements Runnable
{
        Thread t;
        int x;
        Random r;
        int i;
        Unit ob;
        P_C(int y,Unit u)
        {
                r=new Random();
                x=y;
                ob=u;
                t=new Thread(this);
                t.start();
        }
        public void run()
        {
                if(x==1)
```

```
{
        for(i=0;i<=1;i++)
        {
                ob.produce();
                try
                {
                        Thread.sleep(100+r.nextInt(50));
                }
                catch(Exception e)
                {
                        System.out.println(e);
                }
                System.out.println("producer produce "+i+" cake");
        }
}
else if(x==2)
{
        for(i=0;i<=1;i++)
        {
                ob.consume();
                try
                {
                        Thread.sleep(100+r.nextInt(50));
```

```
                }
                catch(Exception e)
                {
                        System.out.println("e");
                }
                System.out.println("consumer consumes "+i+" cake");
        }
}

else if(x==3)
{
        for(i=0;i<=1;i++)
        {
                ob.consume();
                try
                {
                        Thread.sleep(150+r.nextInt(100));
                }
                catch(Exception e)
                {
                        System.out.println("e");
                }
                System.out.println("consumer consumes "+i+" cake");
```

```
                }
            }
        }
}
class P_C_main
{       public static void main(String s[])
        {
            Unit un=new Unit();
            P_C ob1=new P_C(1,un);
            P_C ob2=new P_C(2,un);
            P_C ob3=new P_C(3,un);
        }
}
```

Output:-

```
cake in rack are:1
rack is empty
total cake in rack after consume
consumer consumes 0 cake
rack is empty
producer produce 0 cake
```

cake in rack are:1

consumer consumes 0 cake

total cake in rack after consume

consumer consumes 1 cake

producer produce 1 cake

consumer consumes 1 cake

Lab 10

//GUI in applet

Program: 1

```
/*  create a sample class.make it public.Applet demo
Simple check with appletviewer
*/
import java.awt.*;
```

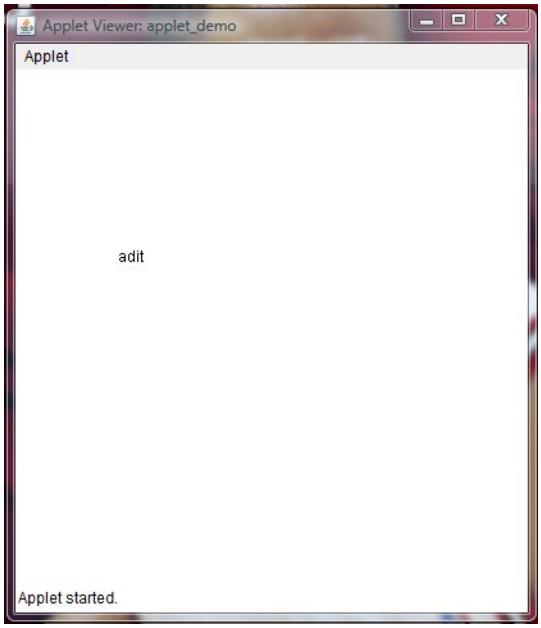
```
import java.applet.*;
/*
  <applet code="applet_demo" width=400 height=400>
  </applet>
*/
public class applet_demo extends Applet
{
public void init()
        {
                System.out.println("init Called");
        }
public void start()
        {
                System.out.println("Start Called");
        }


public void stop()
        {
                System.out.println("Stop Called");
        }

        public void destroy()
```

```
        {
                System.out.println("Destroy Called");
        }
        public void paint(Graphics g)
        {
                g.drawString("adit",80,150);
        }
}
```

Output:-



Init called

Start called

Stop called

Destroy called

Program: 2

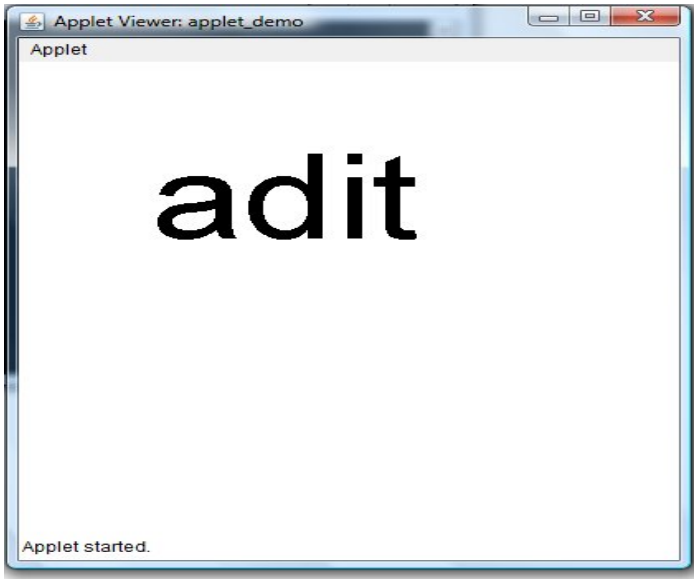
/* create a sample class.make it public.Applet demo

Change the size and type of the font used in applet

```
*/
import java.awt.*;
import java.applet.*;
/*
  <applet code="applet_demo" width=400 height=400>
  </applet>
*/
public class applet_demo extends Applet
{
      public void init()
         {
                  System.out.println("init Called");
 Font f;
    f=new Font("Helvetica",Font.PLAIN,100);
     setFont(f);
    }
      public void start()
         {
     System.out.println("Start Called");
         }
public void stop()
         {                System.out.println("Stop Called");
         }
```

```
        public void destroy()
        {
                System.out.println("Destroy Called");
        }
public void paint(Graphics g)
        {               g.drawString("adit",80,150);
        }
}
```

Output:-



Program: 3

/* create a sample class.make it public.Applet demo

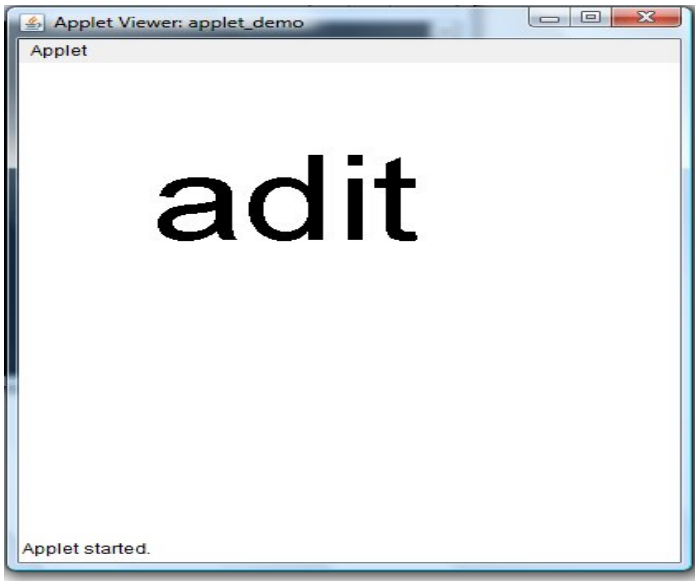
```
Change the font size and color .
*/
import java.awt.*;
import java.applet.*;
/*
  <applet code="applet_demo" width=400 height=400>
  </applet>
*/
public class applet_demo extends Applet
{
      public void init()
         {
                System.out.println("init Called");
    Font f;
    Color c;
 f=new Font("Helvetica",Font.PLAIN,100);
    setFont(f);
}
      public void start()
         {
      System.out.println("Start Called");
         }
```

```
public void stop()
        {
                System.out.println("Stop Called");
        }

        public void destroy()
        {
                System.out.println("Destroy Called");
        }

        public void paint(Graphics g)
        {
             Color c=new Color(250,149,170);
             g.setColor(c);
            g.drawString("adit",80,150);
        }
}
```

Output:-



Here the color of adit is change