## PROGRAM

## //Program to create a simple class in java.

class myclass

{

public static void main(String args[])

{

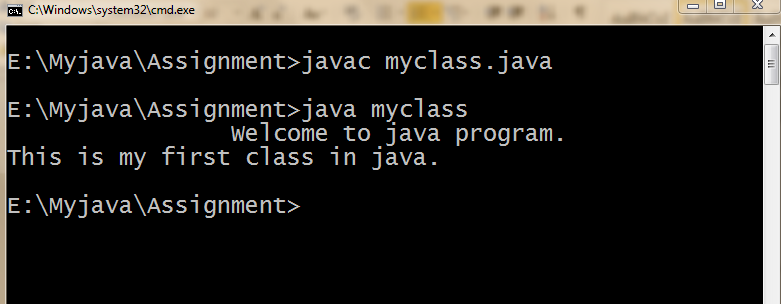
System.out.println("\t\tWelcome to java program.");

System.out.println("This is my first class in java.");

}

}

## OUTPUT:-



## PROGRAM

## //Program to create an object in java

class welcome

{

void method()

{

System.out.println("This is simple method of object ob.");

}

}

class objectclass

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to java program.");

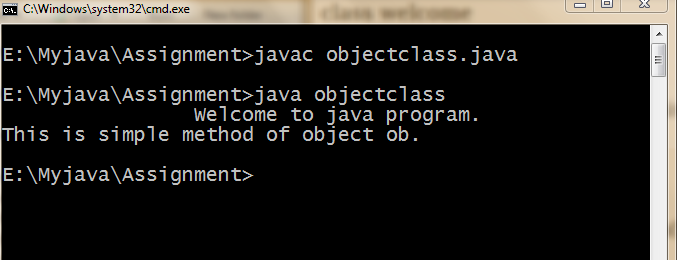
welcome ob=new welcome();//creates a object of class of welcome.

ob.method();//calling the method of class welcome.

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a simple class taking command line argument in java.

class cmd

{

String n;

void method(String name)

{

n=name;

System.out.println("The value of comandline argument:"+n);

}

}

class cmdargs

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to java program.");

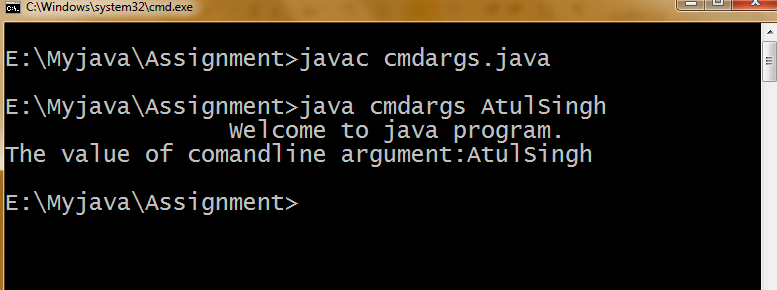
cmd ob=new cmd();

ob.method(args[0]);//command line argument.

}

}

## OUTPUT:-



## PROGRAM

## //Program to implement method overloading in a simple class in java.

class addition

{

int i,j,s=10;

double k,l;

void sum()//without argument and without returning type.

{

System.out.println("Sum is:"+s);

}

void sum(int x,int y)//argument without returning type.

{

i=x;

j=y;

System.out.println("Sum is:"+(s+i+j));

}

void sum(double p,double q)//double argument.

{

k=p;

l=q;

System.out.println("Sum is:"+(s+k+l));

}

int sum(int x)//returning type with argument.

{

return(s+x);

}

}

class methodoverloading

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to java program.");

addition ob=new addition();

ob.sum();

ob.sum(1,2);

ob.sum(3.5,4.5);

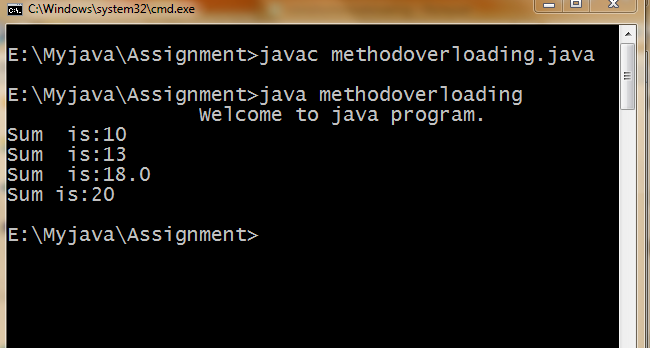
int call=ob.sum(10);

System.out.println("Sum is:"+call);

}

}

## OUTPUT:-



## PROGRAM

## //Constructure overloading in a simple class in java.

Class sum

{

int i,j,s=10;

double k,l;

sum()//no argument constructor.

{

System.out.println("Sum is:"+s);

}

sum(int x)//one argument constructor.

{

i=x;

System.out.println("Sum is:"+(s+i));

}

sum(double p,double q)//multi argument constructor.

{

k=p;

l=q;

System.out.println("Sum is:"+(s+k+l));

}

sum(sum copy)//copy constructor.

{

i=copy.i;

j=copy.j;

System.out.println("Now sum is:"+(i+j));

}

}

class constructoroverloading

{

public static void main(String args[])

{

System.out.println("\t\tWelcome to the java program.");

sum ob=new sum();

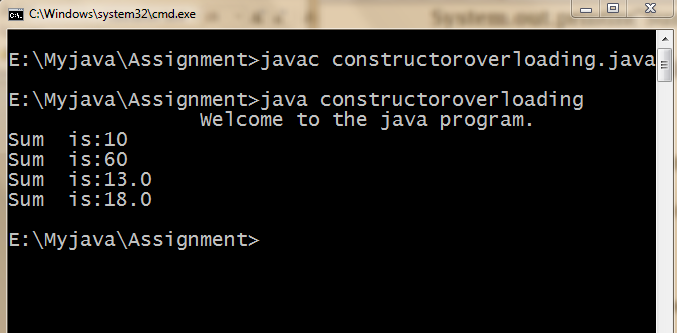
sum ob1=new sum(50);

sum ob2=new sum(1,2);

sum ob3=new sum(3.5,4.5) ;

}

}

OUTPUT:-

## PROGRAM

## //Program to create a 2-D array in java.

import java.io.\*;

class array

{

int n[][]=new int[100][100];

int i,j,row,col;

array(int r,int c)

{

row=r;

col=c;

}

void input(int m[][])throws IOException

{

n=m;

BufferedReader br=new BufferedReader(new InputStreamReader

(System.in));

for(i=0;i<row;i++)

{

for(j=0;j<col;j++)

{

System.out.println("Enter a number.");

n[i][j]=Integer.parseInt(br.readLine());

}

}

}

void display()

{

System.out.println("The two dimantional array is:");

for(i=0;i<row;i++)

{

for(j=0;j<col;j++)

{

System.out.print(n[i][j]+"\t");

}

System.out.println("");

}

}

}

class twod

{

public static void main(String args[])throws IOException

{

BufferedReader br=new BufferedReader(new InputStreamReader

(System.in));

System.out.println("\t\tWelcome to the java program.");

int r,c;

System.out.println("Enter the size of array(row/column)");

r=Integer.parseInt(br.readLine());

c=Integer.parseInt(br.readLine());

array ob=new array(r,c);

int m[][]=new int[r][c];

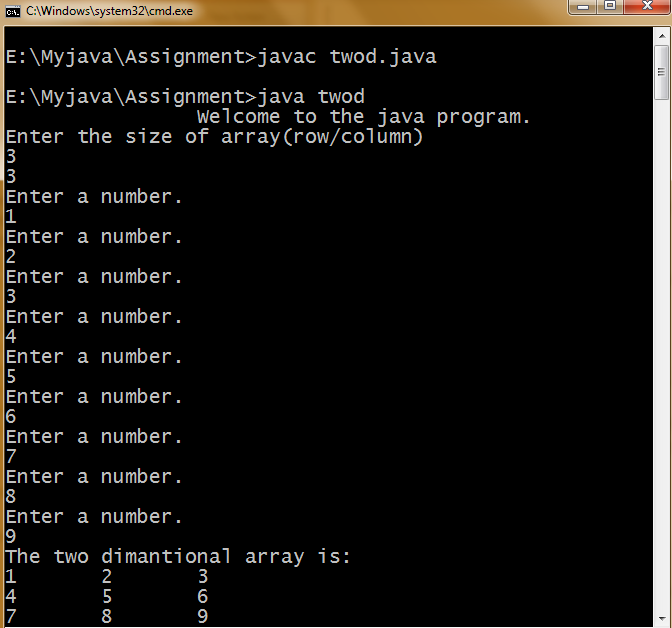
ob.input(m);

ob.display();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a simple class to show the use of this keyword in java.

class Rectangle

{

int length,breadth;

void show(int length,int breadth)

{

this.length=length;

this.breadth=breadth;

}

void mul()

{

System.out.println(length\*breadth);

}

}

class uthis

{

public static void main(String[] args)

{

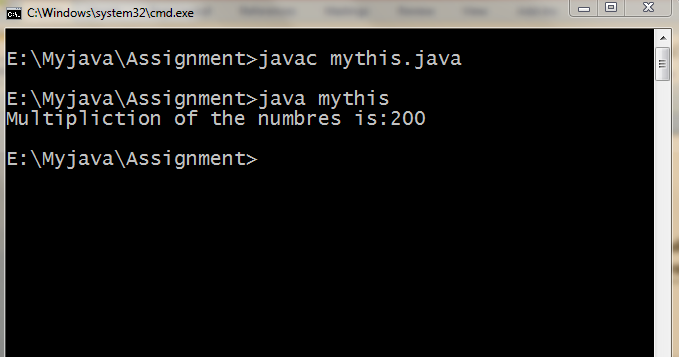
Rectangle ob=new Rectangle();

ob.show(10,20);

ob.mul();

} }

## OUTPUT:-



## PROGRAM

## //Program to create a simple class to show single level inheritance in java.

class rectangle

{

int a,b,c;

void rect(int m,int n)

{

a=m;

b=n;

c=a\*b;

}

}

class area extends rectangle

{

void show()

{

System.out.println("The area of the rectangle is:"+c);

}

}

class singleinh

{

public static void main(String args[])

{

area ob=new area();

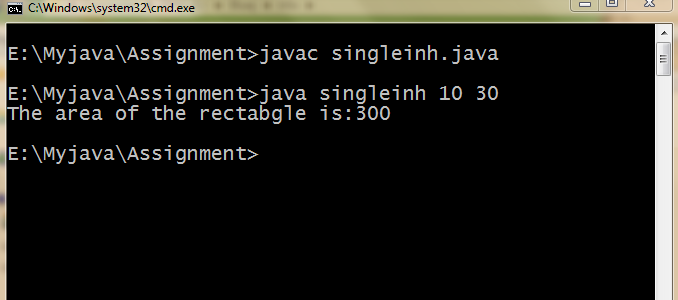
ob.rect(Integer.parseInt(args[0]),Integer.parseInt(args[1]));

ob.show();

}

}

## OUTPUT:-



## PROGRAM

## //Program to create a simple class to show multilevel inheritance in java.

class radius

{

int r;

void rad(int rd)

{

r=rd;

}

}

class area extends radius

{

int ar;

void calculate()

{

ar=22/7\*r\*r;

}

}

class show extends area

{

void display()

{

System.out.println("The area of the circle is:"+ar);

}

}

class multilevelinh

{

public static void main(String args[])

{

show ob=new show();

ob.rad(Integer.parseInt(args[0]));

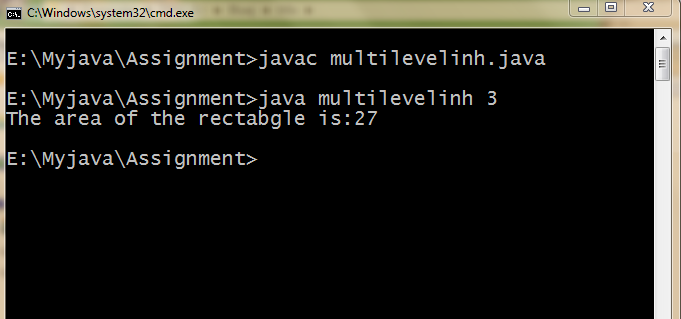
ob.calculate();

ob.display();

}

}

## OUTPUT:-



## PORGRAM

## //Program to create a simple class to show method overriding in java.

class base

{

public void display()

{

System.out.println("This is the base class method.");

}

}

class derive

{

public void display()

{

System.out.println("This is the derive class method.");

}

}

class overri

{

public static void main(String args[])

{

base ob=new base();

derive ob1=new derive();

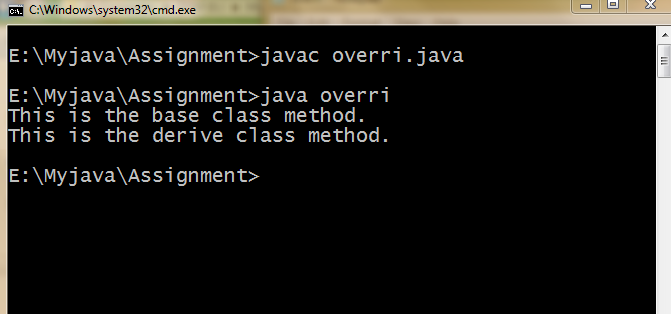
ob.display();

ob1.display();

}

}

## OUTPUT:-



## PORGRAM

## //Pogram to create a class to implement static variables in java.

class usestatic

{

static int a=30;

static int b;

static void meth(int y)

{

System.out.println("Welcome to java program.");

System.out.println("y="+y);

System.out.println("a="+a);

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

}

static

{

System.out.println("static block initialized.");

b=a+10;

}

public static void main(String args[])

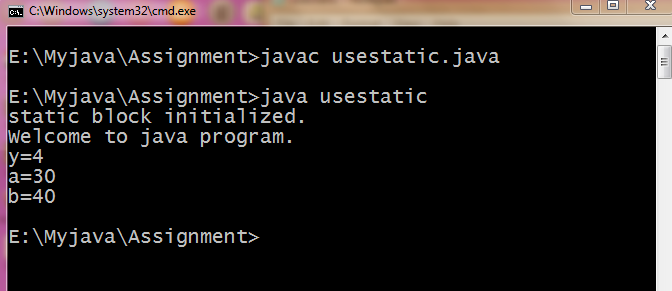
{

meth(4);

}

}

## OUTPUT:-



## PORGRAM

## //Program to create a class to implement static method in java.

class st

{

int i;

static int j;

public static void staticMethod()

{

System.out.println("you can access a static method this way");

}

public void nonStaticMethod()

{

i=100;

j=1000;

System.out.println("Don't try to access a non static method");

}

public static void main(String[] args)

{

//nonStaticMethod();//Error occured.

staticMethod();

stati ob=new stati();

ob.kitchen();

ob.display();

stati.display();//can be call by class name.

}

}

class stati

{

static int price=10;

void kitchen()

{

System.out.println("The price of the onion is:"+"Rs."+price);

}

static void display()

{

price++;

System.out.println("This is static method.");

System.out.println("Now price is:"+price);

}

}

## OUTPUT:-

## E:\Myjava\output\12.PNG