**JAVA SHORT QUIZES**

**JAVA CONTROL STATEMENTS**

1. public class ifelse {

public static void main(String[] args) {

int number=13;

if(number%2==0){

System.out.println("even number");

}else{

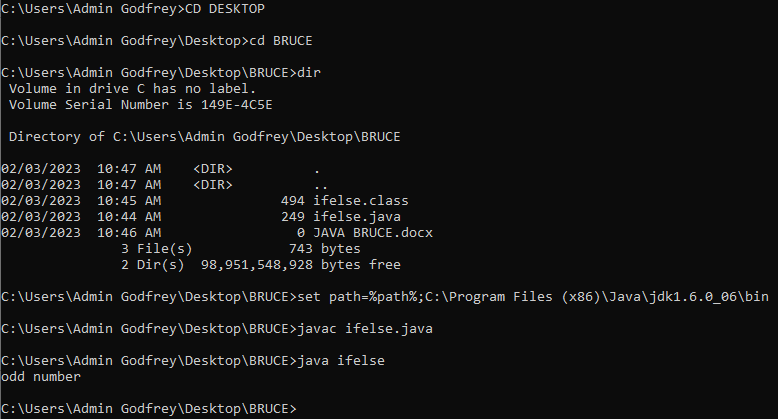
System.out.println("odd number");

}

}

}

OUTPUT



2. public class SwitchMonth {

public static void main(String[] args) {

int month=7;

String monthString="";

switch(month){

case 1: monthString="1 - January";

break;

case 2: monthString="2 - February";

break;

case 3: monthString="3 - March";

break;

case 4: monthString="4 - April";

break;

case 5: monthString="5 - May";

break;

case 6: monthString="6 - June";

break;

case 7: monthString="7 - July";

break;

case 8: monthString="8 - August";

break;

case 9: monthString="9 - September";

break;

case 10: monthString="10 - October";

break;

case 11: monthString="11 - November";

break;

case 12: monthString="12 - December";

break;

default:System.out.println("Invalid Month!");

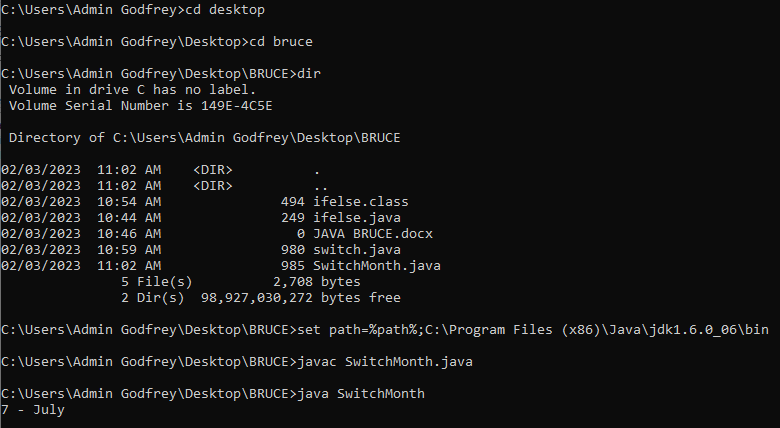
}

System.out.println(monthString);

}

}

OUTPUT



3. public class While {

public static void main(String[] args) {

int i=1;

while(i<=10){

System.out.println(i);

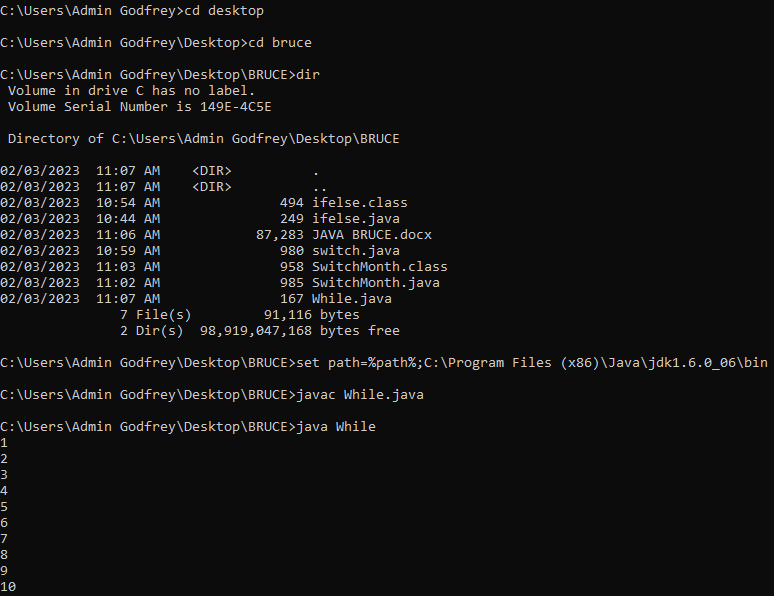
i++;

}

}

}

OUTPUT



**JAVA OBJECT CLASS**

1. class Student{

int id;

String name;

}

class TestStudent5{

public static void main(String args[]){

Student s1=new Student();

s1.id=101;

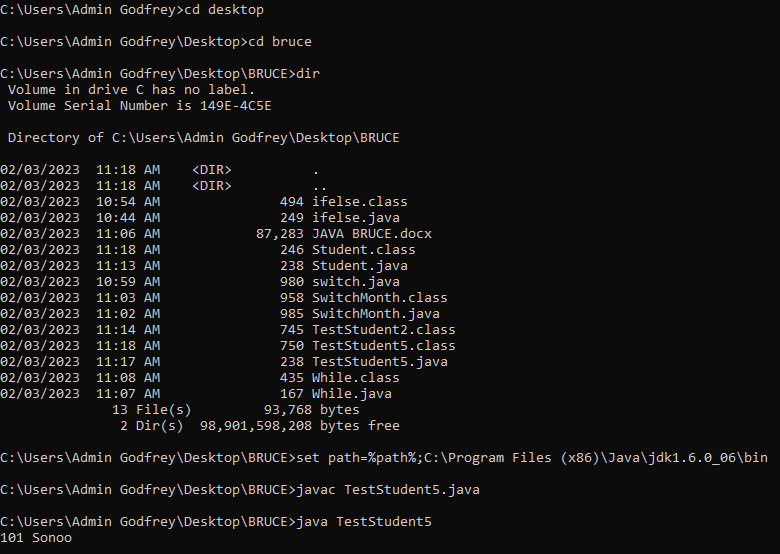
s1.name="Sonoo";

System.out.println(s1.id+" "+s1.name);

}

}

OUTPUT



2. public class Demo

{

public static void main(String[] args)

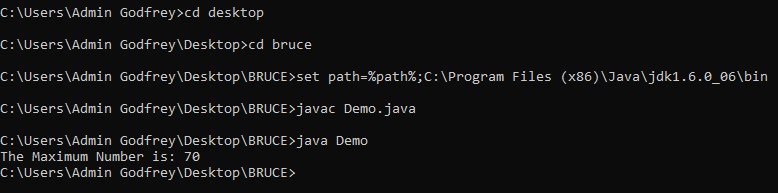
{

System.out.print("The Maximum Number is: " + Math.max(70,5));

}

}

OUTPUT



3. class Student5{

int id;

String name;

Student5(int i,String n){

id = i;

name = n;

}

void display(){System.out.println(id+" "+name);}

public static void main(String args[]){

Student5 s1 = new Student5(111,"Bruce");

Student5 s2 = new Student5(222,"Aron");

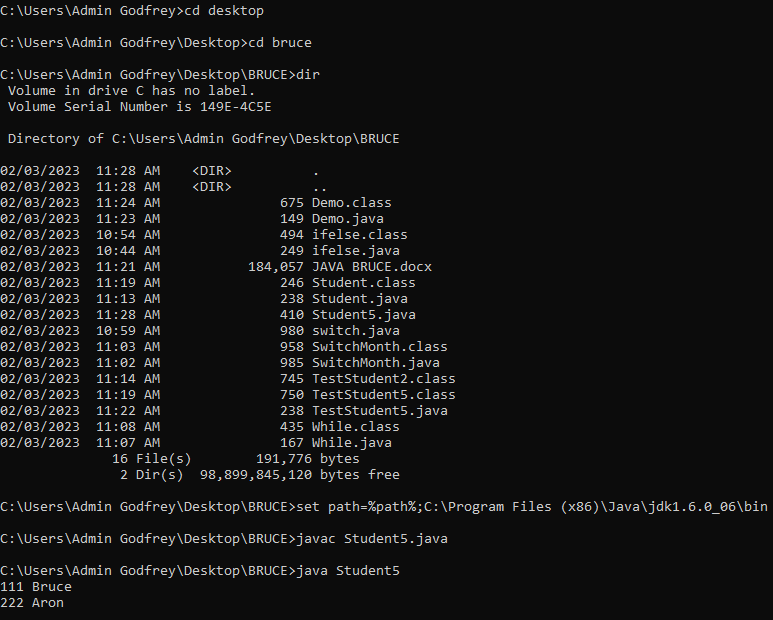
s1.display();

s2.display();

}

}

OUTPUT



**JAVA INHERITENCE**

1. class Employee{

float salary=50000;

}

class IT extends Employee{

int bonus=10000;

public static void main(String args[]){

IT p=new IT();

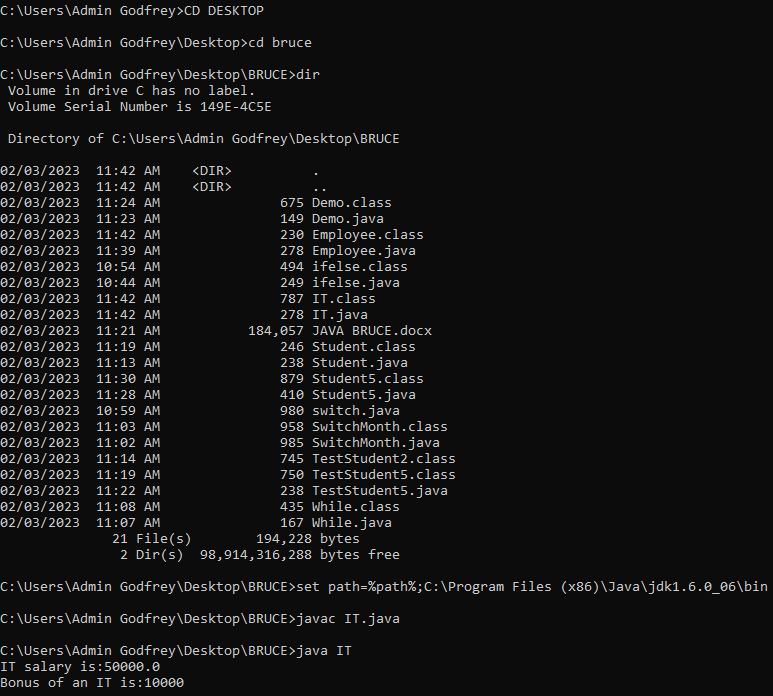
System.out.println("IT salary is:"+p.salary);

System.out.println("Bonus of an IT is:"+p.bonus);

}

}

OUTPUT



2. class Animal{

void eat(){System.out.println("eating...");}

}

class Dog extends Animal{

void bark(){System.out.println("barking...");}

}

class BabyDog extends Dog{

void weep(){System.out.println("weeping...");}

}

class TestInheritance5{

public static void main(String args[]){

BabyDog d=new BabyDog();

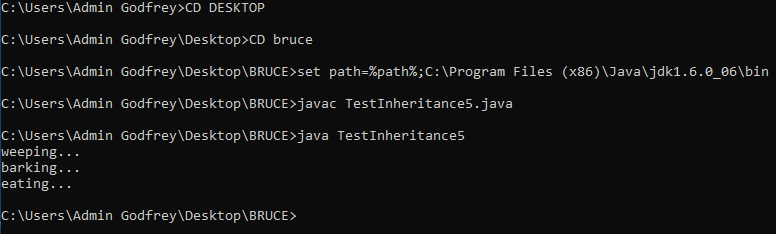
d.weep();

d.bark();

d.eat();

}}

OUTPUT



3. class Operation{

int square(int n){

return n\*n;

}

}

class Circle{

Operation op;

double pi=3.14;

double area(int radius){

op=new Operation();

int rsquare=op.square(radius);

return pi\*rsquare;

}

public static void main(String args[]){

Circle c=new Circle();

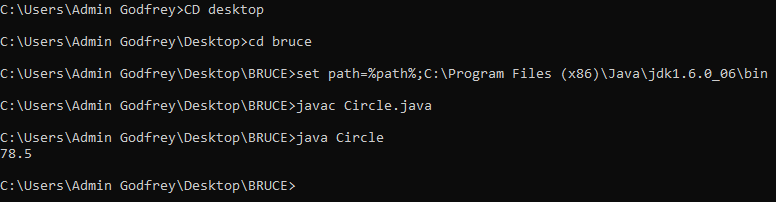
double result=c.area(5);

System.out.println(result);

}

}

OUTPUT



**JAVA POLYMORPHISM**

1. class Adder{

static int add(int a,int b){return a+b;}

static int add(int a,int b,int c){return a+b+c;}

}

class TestOverloading5{

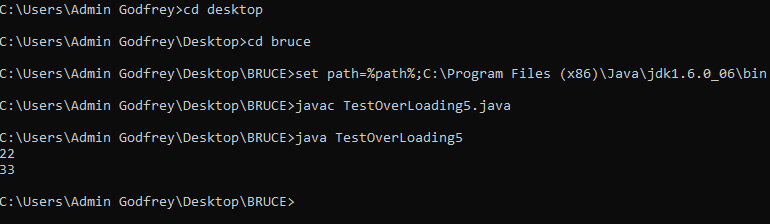
public static void main(String[] args){

System.out.println(Adder.add(11,11));

System.out.println(Adder.add(11,11,11));

}}

OUTPUT



2. class Vehicle{

void run(){System.out.println("Vehicle is MOVING");}

}

class Bike extends Vehicle{

public static void main(String args[]){

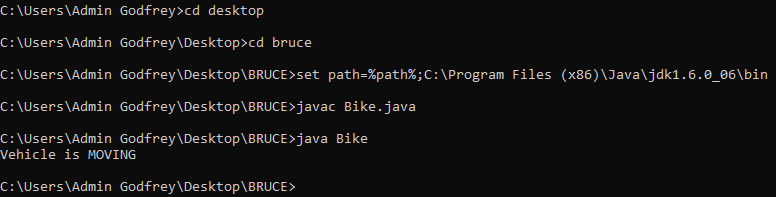
Bike obj = new Bike();

obj.run();

}

}

OUTPUT



3. class A{

A get(){return this;}

}

class B1 extends A{

@Override

B1 get(){return this;}

void message(){System.out.println("welcome to covariant return type");}

public static void main(String args[]){

new B1().get().message();

}

}

OUTPUT

