1. **BOX –**

**class** Box

{

**double** width,height,depth;

Box()

{height=10; width=10; depth=20; }

Box(**double** l)

{

width=height=depth=l;

}

Box(**double** w,**double** h,**double** d)

{ width=w;

height=h;

depth=d;

}

Box(Box ob)

{ width=ob.width;

height=ob.height;

depth=ob.depth;

}

**double** volume()

{ System.***out***.print("The volume of Box is:");

**return** (width\*height\*depth);

}

}

**class** BoxWeight **extends** Box

{

**double** weight;

BoxWeight()

{**super**(); }

BoxWeight(**double** l)

{**super**(l);

weight=l;

}

BoxWeight(**double** w,**double** h,**double** d,**double** w1)

{ **super**(w,h,d);

weight=w1;

}

BoxWeight(BoxWeight ob)

{ **super**(ob);

weight=ob.weight;

}

}

**class** BoxShiftment **extends** BoxWeight

{ **double** cost;

BoxShiftment()

{**super**(); }

BoxShiftment(**double** l)

{**super**(l);

cost=l; }

BoxShiftment(**double** w,**double** h,**double** d,**double** w1,**double** c)

{ **super**(w,h,d,w1);

cost=c; }

BoxShiftment(BoxShiftment ob)

{ **super**(ob);

cost=ob.cost; }}

**class** BoxEx

{ **public** **static** **void** main(String a[])

{ BoxShiftment bs1=**new** BoxShiftment();

System.***out***.println(bs1.volume());

BoxWeight bw1=**new** BoxWeight(10);

System.***out***.println(bw1.volume());

//bs1=bw1;

bw1=bs1;

System.***out***.println(bw1.volume());

}}

1. **Interface –**

**public** **interface** int1 {

**int** ***regno***=101;

**void** input();

String compute();

**void** display(String s1);

}

**interface** int2 **extends** int1

{

**void** display();

}

**import** java.util.Scanner;

**public** **class** stuint **implements** int2

{

**int** jm,cm,pm,tot,avg;

String remarks;

Scanner s;

**public** **void** input()

{

s=**new** Scanner(System.***in***);

System.***out***.println("Enter the Java mark, Cm, PM");

jm=s.nextInt();

cm=s.nextInt();

pm=s.nextInt();

}

**public** String compute()

{ tot=jm+cm+pm;

avg=tot/3;

**if**(avg>=75)

remarks="Passed with Distinction";

**else** **if**(avg>=60 && avg<75)

remarks="Passed with I class";

**else** **if**(avg>=50 && avg<60)

remarks="Pass";

**else** remarks="Fail";

**return** remarks;

}

**public** **void** display(String s)

{ System.***out***.println("Reg no is:"+***regno***);

System.***out***.println("Total is:"+tot);

System.***out***.println("Avg is:"+avg);

System.***out***.println("Remarks"+s);

}

**public** **static** **void** main(String[] args)

{ int1 i1=**new** stuint();

i1.input();

String s1=i1.compute();

i1.display(s1); }

}

1. Polymor.

import java.util.Scanner;

class Bank

{ //int roi;

int getroi()

{ return 2;

}

}

class SBI extends Bank

{ int getroi()

{

return 4;

}

}

class HDFC extends Bank

{ int getroi()

{ return 5;

}

}

class poly {

public static void main(String[] args)

{

double p,n,r,r1,r2,si;

double ci;

Scanner s=new Scanner(System.in);

System.out.println("Enter p,n,r");

p=s.nextInt();

n=s.nextInt();

Bank b=new Bank();

r=b.getroi();

si=(p\*n\*r)/100;

double amt=p+si;

ci=p\*Math.pow((1+r/100),n);

System.out.println("Amt is:"+amt+"\n"+"CI is:"+ci);

SBI sb=new SBI();

r1=sb.getroi();

si=(p\*n\*r1)/100;

ci=p\*Math.pow((1+r1/100),n);

System.out.println("SI is:"+si+"\n"+"CI is:"+ci);

HDFC h=new HDFC();

r2=h.getroi();

si=(p\*n\*r2)/100;

ci=p\*Math.pow((1+r2/100),n);

System.out.println("SI is:"+si+"\n"+"CI is:"+ci);

}

}

1. Upcasting / Downcasting

**class** Alliance

{

}

**class** ACED **extends** Alliance

{

**static** **void** method(Alliance a)

{

**if**(a **instanceof** ACED)

{

ACED d=(ACED)a;

System.***out***.println("Student of ACED");

}

}

}

**class** ASOB **extends** Alliance

{

**static** **void** method(Alliance a)

{

**if**(a **instanceof** ASOB)

{

ASOB d=(ASOB)a;

System.***out***.println("Student of ASOB");

}

}

}

**class** downcast {

**public** **static** **void** main(String[] args)

{

Alliance a1=**new** ACED();

Alliance a2=**new** ASOB();

ACED.*method*(a1);

ASOB.*method*(a2);

}

}

1. User defined Exception

**import** java.util.Scanner;

**class** fund **extends** Exception

{ Scanner s;

**int** bal,amt;

String m;

fund(String s)

{ bal=500;

m=s;

System.***out***.println(m);

}

**void** compute()**throws** fund

{

**try**

{

System.***out***.println("Enter Amount");

s=**new** Scanner(System.***in***);

amt=s.nextInt();

**if**(bal<amt)

**throw** **new** fund("Insufficient");

**else**

bal=bal-amt;

System.***out***.println(bal);

}

**catch**(fund e)

{

System.***out***.println(e.getMessage());

}

**finally**{

}

}

}

**class** ex1 {

**public** **static** **void** main(String[] args) {

**try**

{

fund f=**new** fund("User Define Exception:");

f.compute();

}**catch**(Exception e)

{

System.***out***.println(e.getMessage());

}

}

}

1. Reflection

**import** java.lang.reflect.\*;

**public** **class** ref {

**public** **static** **void** main(String[] args) **throws** ClassNotFoundException

{

**try**

{

Class c=Class.*forName*("java.lang.StringBuffer");

Constructor con[]=c.getConstructors();

**for**(Constructor c1:con)

System.***out***.println(c1.getName());

Method met[]=c.getMethods();

**for**(Method m1:met)

System.***out***.println(m1.getName());

Field fie[]=c.getFields();

**for**(**int** i=0;i<fie.length;i++)

System.***out***.println(fie[i].getName());

}**catch**(Exception e)

{

System.***out***.println(e.getMessage());

}

}

}

**import** java.io.IOException;

**import** java.util.\*;

**class** ex1 {

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

{

**int** len=4;

System.***out***.println(*pin*(len));

}

**static** **char**[] pin(**int** leng)

{

**char**[] pwd=**new** **char**[leng];

**try**

{

String num="0123456789";

Random r=**new** Random();

**for**(**int** i=0;i<leng;i++)

pwd[i]=num.charAt(r.nextInt(num.length()));

//return pwd;

}

**catch**(Exception e)

{

System.***out***.println("Error:"+e.getMessage());

}

//return pwd;

**return** pwd;

}

}