‘instanceof’ operator :--

Eg: 1:--

class First{}

class Second extends First{}

class Third extends Second{}

class Demo

{

public static void main(String args[ ])

{

First f=new First();

Second s=new Second();

Third t=new Third();

System.out.println(f instanceof First);

System.out.println(f instanceof Second);

System.out.println(f instanceof Third);

System.out.println(s instanceof First);

System.out.println(s instanceof Second);

System.out.println(s instanceof Third);

System.out.println(t instanceof First);

System.out.println(t instanceof Second);

System.out.println(t instanceof Third);

}

}

o/p:--

true

false

false

true

true

false

true

true

true

**Eg: 2:--**

class First{}

class Second extends First{}

class Third extends Second{}

class Demo

{

public static void main(String args[ ])

{

First f=new First();

Second s=new Second();

Third t=new Third();

f=s;

System.out.println(f instanceof First);

System.out.println(f instanceof Second);

System.out.println(f instanceof Third);

System.out.println(s instanceof First);

System.out.println(s instanceof Second);

System.out.println(s instanceof Third);

System.out.println(t instanceof First);

System.out.println(t instanceof Second);

System.out.println(t instanceof Third);

}

}

o/p:--

true

true

false

true

true

false

true

true

true

Eg:3:--

class First{}

class Second extends First{}

class Third extends Second{}

class Demo

{

public static void main(String args[ ])

{

First f=new First();

Second s=new Second();

Third t=new Third();

s=t;

System.out.println(f instanceof First);

System.out.println(f instanceof Second);

System.out.println(f instanceof Third);

System.out.println(s instanceof First);

System.out.println(s instanceof Second);

System.out.println(s instanceof Third);

System.out.println(t instanceof First);

System.out.println(t instanceof Second);

System.out.println(t instanceof Third);

}

}

o/P;--

true

false

false

true

true

true

true

true

true

Eg:4 :--

class First{}

class Second extends First{}

class Third extends Second{}

class Demo

{

public static void main(String args[ ])

{

First f=new First();

Second s=new Second();

Third t=new Third();

f=s=t;

System.out.println(f instanceof First);

System.out.println(f instanceof Second);

System.out.println(f instanceof Third);

System.out.println(s instanceof First);

System.out.println(s instanceof Second);

System.out.println(s instanceof Third);

System.out.println(t instanceof First);

System.out.println(t instanceof Second);

System.out.println(t instanceof Third);

}

}

o/p:--

true

true

true

true

true

true

true

true

true

**‘Equals’ method :--**

**Method of ‘Object’ class :--**

**public boolean equals(Object obj)**

**Eg: 1**

class Box

{

double width,height,length;

Box(double w, double h, double l)

{

width=w;

height=h;

length=l;

}

}

class Demo

{

public static void main(String args[ ])

{

Box b1=new Box(10,11,12);

Box b2=new Box(10,11,12);

System.out.println(b1.equals(b2));

b1=b2;

System.out.println(b1.equals(b2));

Integer i1=new Integer(10);

Integer i2=new Integer(10);

System.out.println(i1.equals(i2));

}

}

**Eg:2:--**

class Wrestler

{

private String wname;

private int weight;

Wrestler(String name, int wt)

{

wname=name;

weight=wt;

}

String getWname()

{

return wname;

}

int getWeight()

{

return weight ;

}

public boolean equals(Object obj)

{

if((obj instanceof Wrestler) && (((Wrestler)obj)).getWeight()==weight)

return true;

else

return false;

}

}

class Demo

{

public static void main(String args[ ])

{

Wrestler w1=new Wrestler("Satish",80);

Wrestler w2=new Wrestler("Anil",80);

Wrestler w3=new Wrestler("Satish",84);

System.out.println(w1.equals(w2));

System.out.println(w1.equals(w3));

}

}

o/p:--

true

false