DAY-49

------

There are two types clonings in Java:

-------------------------------------

1.Shallow Cloning/Shallow Copy

2.Deep Cloning/Deep Copy

1.Shallow Cloning/Shallow Copy:

-------------------------------

In this cloning mechanism,while cloning an object if any associated object is encountered then JVM

will not duplicate associated object along with data duplication,where duplicated object is also refer

the same associated object which was refereed by original object.

NOTE:Shallow Cloning is default cloning mechanism in Java.

EXAPLE:1

---------

// SHALLOW COPY/SHALLOW CLONING

class Demo implements Cloneable

{

int i = 10;

int j = 20;

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

{

Demo d1 = new Demo();

Demo d2 =(Demo)d1.clone();

System.out.println("the object is copied and new dupliacte object is created");

System.out.println("values of original object is:"+d1.i+" "+d1.j);

System.out.println("values of duplicate object is:"+d2.i+" "+d2.j);

}

}

OUTPUT:

--------

the object is copied and new dupliacte object is created

values of original object is:10 20

values of duplicate object is:10 20

EXAPLE:2

--------

class sample1

{

int j;

sample1(int j)

{

this.j=j;

}

}

class sample2 implements Cloneable

{

sample1 p;

int i;

sample2(sample1 p, int i)

{

this.p=p;

this.i=i;

}

public Object clone() throws CloneNotSupportedException

{

return super.clone();

}

}

class shallowCopy

{

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

{

sample1 s1 = new sample1(20);

sample2 s2 = new sample2(s1,10);

sample2 s3 = (sample2)s2.clone();

System.out.println(s2.i+" "+s2.p.j);

s3.i=1234;

s3.p.j=9686;

System.out.println(s2.i+" "+s2.p.j);

}

}

OUTPUT:

-------

10 20

10 96

2. Deep Cloning/Deep Copy:

--------------------------

In this cloning mechanism,while cloning an object if JVM encounter any associated object then JVM

will duplicate associated object also along with data duplication.In this cloning mechanism,both

original object and cloned object are having their own duplicated associated objects copy,both are not

referring a single associated object.

EXAPLE:

---------

class Bike

{

int j;

Bike(int j)

{

this.j=j;

}

}

class Car implements Cloneable

{

Bike b;

int i;

Car(Bike b,int i)

{

this.b=b;

this.i=i;

}

public Object clone() throws CloneNotSupportedException

{

Bike b1 =new Bike(b.j);

Car c1 = new Car(b1,i);

return c1;

}

}

class DeepCopy

{

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

{

Bike b2 = new Bike(20);

Car c2 = new Car(b2,25);

System.out.println(c2.i+" "+c2.b.j);

Car c3 = (Car)c2.clone();

c3.i=8989;

c3.b.j=7676;

System.out.println(c2.i+" "+c2.b.j);

}

}

OUTPUT:

--------

20 25

20 25

Difference b/w extends and implements

-------------------------------------

extends:

--------

1. A class can inherit other class using the extends keyword.An interface can inherit another interface using the extends keyword.

2. A subclass which extends super class may or may not override all the methods of superclass

3. An inertface can extends any number of interface

4. A class can extends only one super class.

implements:

----------

1. A class can be implemented other inteface uisng implements keyword.

2. class implementing interface should provide body for all methods of interface

3. An interface can never implements other interface

4. A class can implement multiple interface simultaneously.

Difference b/w interface and abstract class

--------------------------------------------

interface:

---------

1.100% abstraction is achieved

2. interface are used when we don't know complete implementation

3.only abstract methods are allowed

4.Every methods inside interface are by default public and abstract

5.Every variables inside interface are by default public static final

6. following modifiers are not allowed for abstract methods static, private,strictfp,protected,final

7.interface cannot have static blocks

8.interface cannot have instance blocks

9.constructors are not allowed

abstract class

--------------

1.0-100% abstraction can be achieved

2.abstract class is used when we know parent implementation

3.both abstract and concrete methods are allowed

4.every method present in abstract class need not to be public and abstract

5.every variables present in abstract class need not to be public static final

6.no such restructions for methods

7.abstract class can have static block

8.abstract class can have instance block

9.construced are allowed