***What is cloning:*** The process of creating exactly the same copy of an object is called cloning.

***What is the purpose of cloning***?

1. To keep a Backup object for future use

2. To preserve the initial state of an object, so that we can compare it with new one

***How to do cloning:***

By using the Object class clone method we can perform object cloning.

Or we can override our own cloning method. The full name of cloning method is like

protected native Object clone() throws CloneNotSupportedException

This method is protected. So we can use it within the same package and outside the package by extending the class.

It is a native method. Means the code is implemented in other language.

The method returns Object.

The method throws CloneNotSupportedException

***There are two kind of Cloning***

1. Shallow cloning

2. Deep cloning.

Shallow cloning means using the Object class clone() method without overriding the method. In this case we cannot get the proper cloning.

Deep cloning means override the cloning method to get the full cloning of an object

***Examples Of cloning:***

// Examples of object cloning

class TestClone implements Cloneable //for cloning must implements the Cloneable interface. This interface has no method, so it is a markup interface

{ int i=10;

int j=10;

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

{

TestClone t1= new TestClone();

System.out.println(t1.i +"......." + t1.j); // Prints 10.....10. It is the original object

TestClone t2= (TestClone) t1.clone(); // object class clone() method returns Object. So we must type cuts it to our 'TestClone' class

System.out.println(t2.i +"......." +t2.j); // the cloning is completed. So the output will be 10.....10 like t1 object

t2.i=20; t2.j=30; // new value assign to t2 object. now the t2 object will change but t1 remain the same

System.out.println(t2.i +"......." +t2.j); // print 20.....30... It has changed its value

System.out.println(t1.i +"......." +t1.j); // It prints 10.....10 means remain unchanged.

}

}