JAVA.UTIL.VECTOR.CLONE METHOD

Description

The **clone** method is used to get another copy of an existing vector.The copy will have a **reference** to a clone of the internal data array but not a **reference** to the original internal data array.

Declaration

Following is the declaration for **java.util.Vector.clone** method

```
public Object clone()
```

Parameters

- This method does not take any parameter as input.

Return Value

Object--The method call returns a clone of this vector as an object.

Exception

- **CloneNotSupportedException** -- This exception is thrown if the object's class does not support the Cloneable interface.

Example

The following example shows the usage of java.util.Vector.clone method.

```
package com.tutorialspoint;

import java.util.Vector;

public class VectorDemo {
   public static void main(String[] args) {
   // create an empty Vector vec with an initial capacity of 4
   Vector<Integer> vec = new Vector<Integer>(4);
   Vector<Integer> vecclone = new Vector<Integer>(4);

   // use add() method to add elements in the vector
   vec.add(4);
   vec.add(3);
   vec.add(2);
   vec.add(1);

   // let us print all the elements available in vec
   System.out.println("Added numbers in vec vector are :- ");
   for (Integer number : vec) {
   System.out.println("Number = " + number);
   }

   // let us clone the vector vec
   vecclone=(Vector)vec.clone();

   // let us print all the elements available in vecclone
   System.out.println("Added numbers in vecclone vector are :-");
   for (Integer number : vecclone) {
   System.out.println("Number = " + number);
   }
   }
}
```

Let us compile and run the above program, this will produce the following result.

```
Added numbers in vec vector are :-
Number = 4
Number = 3
Number = 2
Number = 1
Added numbers in vecclone vector are :-
Number = 4
Number = 3
Number = 2
Number = 1
```