**\*\* Vector \*\***

**Vector**

1. It is internal working based on Array List.
2. It is synchronized.
3. It is thread safe.
4. It implements list, Clone-able serializable and random access.
5. It is grow-able in nature.
6. It allows random access.
7. Insertion and deletion operation are complex.
8. In Array List insertion order is preserved.
9. Internal data structure is Array.
10. It allows to Store duplicate data.
11. Null value can also be inserted.
12. We can store homogeneous type of data.

Formula of Vector

New Capacity = Current Capacity \* 2 ;

1. **Vector have Four Constructor.**
2. **Vector()**
3. **Vector(int Capacity)**
4. **Vector(int Capacity, int increase capacity)**
5. **Vector(Collection)**

|  |  |  |
| --- | --- | --- |
| **No**. | **Vector** | **Array List** |
|  | It is synchronized. | It is not synchronized . |
|  | It is thread safe | It is not thread safe |
|  | Its size increases by  Current capa \* 2. | Its size increases by  (Current capa \* 3) / 2. |
|  | Its performance is slow  Because it is synchronized. | Its performance is fast Because  It is not synchronized. |
|  | Vector contain Capacity() method. | Array List does not contain Capacity() method. |
|  | It have four Constructor. | It have Three constructor. |

|  |  |  |
| --- | --- | --- |
| **No.** | **Method** | **Description** |
|  | [add()](https://www.javatpoint.com/java-vector-add-method) | It is used to append the specified element in the given vector. |
|  | [addAll()](https://www.javatpoint.com/java-vector-addall-method) | It is used to append all of the elements in the specified collection to the end of this Vector. |
|  | [addElement()](https://www.javatpoint.com/java-vector-addelement-method) | It is used to append the specified component to the end of this vector. It increases the vector size by one. |
|  | [capacity()](https://www.javatpoint.com/java-vector-capacity-method) | It is used to get the current capacity of this vector. |
|  | [clear()](https://www.javatpoint.com/java-vector-clear-method) | It is used to delete all of the elements from this vector. |
|  | [clone()](https://www.javatpoint.com/java-vector-clone-method) | It returns a clone of this vector. |
|  | [contains()](https://www.javatpoint.com/java-vector-contains-method) | It returns true if the vector contains the specified element. |
|  | [containsAll()](https://www.javatpoint.com/java-vector-containsall-method) | It returns true if the vector contains all of the elements in the specified collection. |
|  | [copyInto()](https://www.javatpoint.com/java-vector-copyinto-method) | It is used to copy the components of the vector into the specified array. |
|  | [elementAt()](https://www.javatpoint.com/java-vector-elementat-method) | It is used to get the component at the specified index. |
|  | [elements()](https://www.javatpoint.com/java-vector-elements-method) | It returns an enumeration of the components of a vector. |
|  | [ensureCapacity()](https://www.javatpoint.com/java-vector-ensurecapacity-method) | It is used to increase the capacity of the vector which is in use, if necessary. It ensures that the vector can hold at least the number of components specified by the minimum capacity argument. |
|  | [equals()](https://www.javatpoint.com/java-vector-equals-method) | It is used to compare the specified object with the vector for equality. |
|  | [firstElement()](https://www.javatpoint.com/java-vector-firstelement-method) | It is used to get the first component of the vector. |
|  | [forEach()](https://www.javatpoint.com/java-vector-foreach-method) | It is used to perform the given action for each element of the Iterable until all elements have been processed or the action throws an exception. |
|  | [get()](https://www.javatpoint.com/java-vector-get-method) | It is used to get an element at the specified position in the vector. |
|  | [hashCode()](https://www.javatpoint.com/java-vector-hashcode-method) | It is used to get the hash code value of a vector. |
|  | [indexOf()](https://www.javatpoint.com/java-vector-indexof-method) | It is used to get the index of the first occurrence of the specified element in the vector. It returns -1 if the vector does not contain the element. |
|  | [insertElementAt()](https://www.javatpoint.com/java-vector-insertelementat-method) | It is used to insert the specified object as a component in the given vector at the specified index. |
|  | [isEmpty()](https://www.javatpoint.com/java-vector-isempty-method) | It is used to check if this vector has no components. |
|  | [iterator()](https://www.javatpoint.com/java-vector-iterator-method) | It is used to get an iterator over the elements in the list in proper sequence. |
|  | [lastElement()](https://www.javatpoint.com/java-vector-lastelement-method) | It is used to get the last component of the vector. |
|  | [lastIndexOf()](https://www.javatpoint.com/java-vector-lastindexof-method) | It is used to get the index of the last occurrence of the specified element in the vector. It returns -1 if the vector does not contain the element. |
|  | listIterator() | It is used to get a list iterator over the elements in the list in proper sequence. |
|  | [remove()](https://www.javatpoint.com/java-vector-remove-method) | It is used to remove the specified element from the vector. If the vector does not contain the element, it is unchanged. |
|  | [removeAll()](https://www.javatpoint.com/java-vector-removeall-method) | It is used to delete all the elements from the vector that are present in the specified collection. |
|  | [removeAllElements()](https://www.javatpoint.com/java-vector-removeallelements-method) | It is used to remove all elements from the vector and set the size of the vector to zero. |
|  | [remove Element()](https://www.javatpoint.com/java-vector-removeelement-method) | It is used to remove the first (lowest-indexed) occurrence of the argument from the vector. |
|  | [remove ElementAt()](https://www.javatpoint.com/java-vector-removeelementat-method) | It is used to delete the component at the specified index. |
|  | removeIf() | It is used to remove all of the elements of the collection that satisfy the given predicate. |
|  | Remove Range() | It is used to delete all of the elements from the vector whose index is between fromIndex, inclusive and toIndex, exclusive. |
|  | [replaceAll()](https://www.javatpoint.com/java-vector-replaceall-method) | It is used to replace each element of the list with the result of applying the operator to that element. |
|  | [retainAll()](https://www.javatpoint.com/java-vector-retainall-method) | It is used to retain only that element in the vector which is contained in the specified collection. |
|  | set() | It is used to replace the element at the specified position in the vector with the specified element. |
|  | setElementAt() | It is used to set the component at the specified index of the vector to the specified object. |
|  | setSize() | It is used to set the size of the given vector. |
|  | size() | It is used to get the number of components in the given vector. |
|  | sort() | It is used to sort the list according to the order induced by the specified Comparator. |
|  | spliterator() | It is used to create a late-binding and fail-fast Spliterator over the elements in the list. |
|  | subList() | It is used to get a view of the portion of the list between fromIndex, inclusive, and toIndex, exclusive. |
|  | toArray() | It is used to get an array containing all of the elements in this vector in correct order. |
|  | toString() | It is used to get a string representation of the vector. |
|  | trimToSize() | It is used to trim the capacity of the vector to the vector's current size. |

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

class Q01\_Vector\_add

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**        vr**.**add**(**20**);**

vr**.**add**(**30**);**        vr**.**add**(**40**);**

vr**.**add**(**50**);**        vr**.**add**(**"ABCD"**);**

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

**}**

**}**

[10, 20, 30, 40, 50, ABCD]

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

class Q02\_Vector\_add

**{**

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

**{**

        Vector**<**Integer**>** vr **=** **new** Vector**<>();**

        vr**.**add**(**10**);**        vr**.**add**(**20**);**

        vr**.**add**(**30**);**        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

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

**}**

**}**

[10, 20, 30, 40, 50]

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

class Q03\_Vector\_add

**{**

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

**{**

        Vector**<**Integer**>** vr **=** **new** Vector**<>();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

        vr**.**add**(**"Vishal Soner"**);**

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

**}**

**}**

 error: no suitable method found for add(String)

                vr.add("Vishal Soner");

                  ^

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

class Q04\_Vector\_remove

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

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

        System**.**out**.**println**(** vr**.**remove**()** **);**

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

**}**

**}**

 error: no suitable method found for remove(no arguments)

                System.out.println( vr.remove() );

                                      ^

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

class Q05\_Vector\_remove

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

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

        System**.**out**.**println**(** vr**.**remove**(**2**)** **);**

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

**}**

**}**

[10, 20, 30, 40, 50]

30

[10, 20, 40, 50]

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

class Q06\_Vector\_set

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

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

        System**.**out**.**println**(** vr**.**set**(**2**,** 200**)** **);**

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

**}**

**}**

[10, 20, 30, 40, 50]

30

[10, 20, 200, 40, 50]

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

class Q07\_Vector\_get

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

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

        System**.**out**.**println**(** vr**.**get**(**4**)** **);**

**}**

**}**

[10, 20, 30, 40, 50]

50

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

class Q08\_Vector\_addAll

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

        Vector vr2 **=** **new** Vector**();**

        vr2**.**addAll**(**vr**);**

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

        System**.**out**.**println**(** vr2 **);**

**}**

**}**

[10, 20, 30, 40, 50]

[10, 20, 30, 40, 50]

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

class Q09\_Vector\_removeAll

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

        vr**.**add**(**20**);**

        vr**.**add**(**30**);**

        vr**.**add**(**40**);**

        vr**.**add**(**50**);**

        Vector vr2 **=** **new** Vector**();**

        vr2**.**add**(**10**);**

        vr2**.**add**(**20**);**

        vr2**.**add**(**300**);**

        vr2**.**add**(**400**);**

        vr2**.**add**(**500**);**

        vr2**.**removeAll**(**vr**);**

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

        System**.**out**.**println**(** vr2 **);**

**}**

**}**

[10, 20, 30, 40, 50]

[300, 400, 500]

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

class Q10\_Vector\_subList

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**     vr**.**add**(**20**);**     vr**.**add**(**30**);**

        vr**.**add**(**40**);**     vr**.**add**(**50**);**

        List vr2 **=** **new** Vector**();**

        vr2 **=** vr**.**subList**(**2**,** 5**);**

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

        System**.**out**.**println**(** vr2 **);**

**}**

**}**

[10, 20, 30, 40, 50]

[30, 40, 50]

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

class Q11\_Vector\_toArray

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**     vr**.**add**(**20**);**     vr**.**add**(**30**);**

        vr**.**add**(**40**);**     vr**.**add**(**50**);**

        Object o**[]** **=** **new** Object**[**vr**.**size**()];**

        o **=** vr**.**toArray**();**

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

**for(** Object io **:** o**)**

**{**

            System**.**out**.**print**(** io **+** ", " **);**

**}**

**}**

**}**

[Ljava.lang.Object;@15db9742

10, 20, 30, 40, 50,

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

class Q12\_Vector\_isEmpty

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        System**.**out**.**println**(** vr**.**isEmpty**()** **);**

        vr**.**add**(**10**);**     vr**.**add**(**20**);**     vr**.**add**(**30**);**

        vr**.**add**(**40**);**     vr**.**add**(**50**);**

        System**.**out**.**println**(** vr**.**isEmpty**()** **);**

**}**

**}**

true

false

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

class Q13\_Vector

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**for(**int i **=** 1**;** i **<=** 10**;** i**++)**

**{**

            vr**.**add**(**i**\***10**);**

**}**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        vr**.**add**(**500**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**}**

**}**

0

10

10

10

11

20

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

class Q14\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**(**4**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**for(**int i **=** 1**;** i **<=** 10**;** i**++)**

**{**

            vr**.**add**(**i**\***10**);**

**}**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        vr**.**add**(**500**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**}**

**}**

0

4

10

16

11

16

class Q15\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**(**6**,** 4**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**for(**int i **=** 1**;** i **<=** 10**;** i**++)**

**{**

            vr**.**add**(**i**\***10**);**

**}**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        vr**.**add**(**500**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**}**

**}**

Vector vr = new Vector(6, 4);

0

6

10

10

11

14

if Case :

Vector vr = new Vector(-6, 4);

Exception in thread "main" java.lang.IllegalArgumentException: Illegal Capacity: -6

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

class Q16\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**(**6**,** **-**4**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**for(**int i **=** 1**;** i **<=** 20**;** i**++)**

**{**

            vr**.**add**(**i**\***10**);**

**}**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        vr**.**add**(**500**);**

        vr**.**add**(**501**);**

        vr**.**add**(**502**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**}**

**}**

0

6

20

24

23

24

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

class Q17\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**(**0**,** 0**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**for(**int i **=** 1**;** i **<=** 20**;** i**++)**

**{**

            vr**.**add**(**i**\***10**);**

**}**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        vr**.**add**(**500**);**

        System**.**out**.**println**(** vr**.**size**());**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

**}**

**}**

0

0

20

32

21

32

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

class Q18\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**

    vr**.**add**(**20**);**

vr**.**add**(**30**);**

vr**.**add**(**40**);**

vr**.**add**(**50**);**

vr**.**add**(**60**);**

        Vector vr2 **=** **new** Vector**(**vr**);**

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

        System**.**out**.**println**(** vr2 **);**

**}**

**}**

[10, 20, 30, 40, 50, 60]

[10, 20, 30, 40, 50, 60]

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

class Q19\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**     vr**.**add**(**20**);**     vr**.**add**(**30**);**

        vr**.**add**(**40**);**     vr**.**add**(**50**);**     vr**.**add**(**60**);**

        System**.**out**.**println**(** vr**.**size**()** **);**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        Vector vr2 **=** **new** Vector**(**vr**);**

        System**.**out**.**println**(** vr2**.**size**()** **);**

        System**.**out**.**println**(** vr2**.**capacity**()** **);**

**}**

**}**

6

10

6

6

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

class Q20\_Vector\_Constructor

**{**

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

**{**

        Vector vr **=** **new** Vector**();**

        vr**.**add**(**10**);**     vr**.**add**(**20**);**     vr**.**add**(**30**);**

        vr**.**add**(**40**);**     vr**.**add**(**50**);**     vr**.**add**(**60**);**

        System**.**out**.**println**(** vr**.**size**()** **);**

        System**.**out**.**println**(** vr**.**capacity**()** **);**

        Vector vr2 **=** **new** Vector**(**vr**);**

        System**.**out**.**println**(** vr2**.**size**()** **);**

        System**.**out**.**println**(** vr2**.**capacity**()** **);**

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

        System**.**out**.**println**(** vr2 **);**

        vr2**.**add**(**70**);**

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

        System**.**out**.**println**(** vr2 **);**

        System**.**out**.**println**(** vr2**.**size**()** **);**

        System**.**out**.**println**(** vr2**.**capacity**()** **);**

**}**

**}**

6

10

6

6

[10, 20, 30, 40, 50, 60]

[10, 20, 30, 40, 50, 60]

[10, 20, 30, 40, 50, 60]

[10, 20, 30, 40, 50, 60, 70]

7

12