Practical No. 14: Develop program for implementation of Vectors in Java.

Practical Significance:

Vector implementation a dynamic array. Vector hold different number of objects. Students will able to use Vectors in the program efficiently.

Relevant Course Outcome:

Develop programs using Object Oriented methodology in Java.

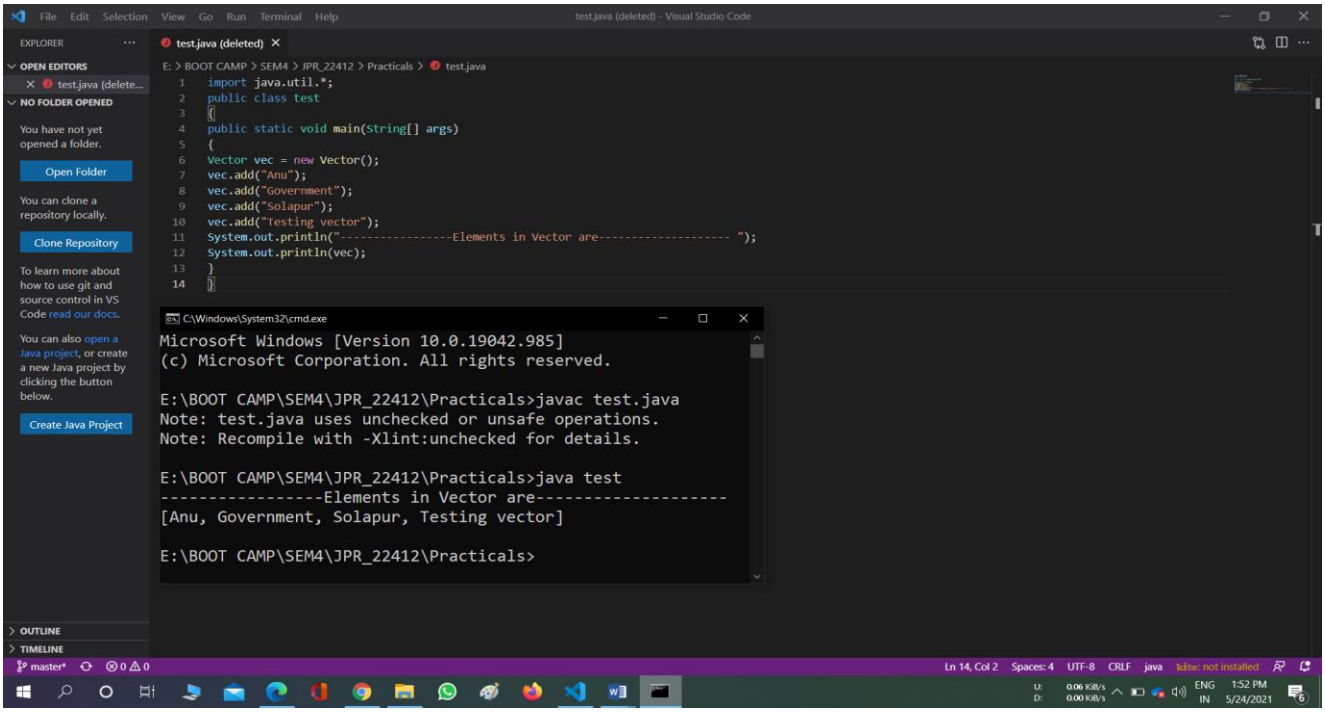
Practical Outcome:

Develop program for implementation of Vectors in Java.

Sr. No.	Syntax	Task Performed
1.	void add Element(Object ob)	Adds the specified component to the end of this vector increasing its size by one
2.	int capacity()	Returns the current capacity of this vector
3.	boolean contains(Object elem)	Tests if the specified object is a component in this vector.
4.	void clear()	Removes all the element form this vector
5.	Object elementAt(int index)	Returns the component at specified index
6.	Enumeration elements()	Returns enumeration of components of this vector
7.	Object firstElement()	Returns first component of this vector
8.	Object lastElement()	Returns last component of this vector
9.	int indexOf(Object elem)	Searches for the first occurrence of given argument
10.	void insetElementAt(Object obj, int index)	Inserts specified object as a component at the specified index position
11.	void removeElementAt(int index)	Removes the element at specified position in the vector
12.	boolean removeElement(Object obj)	Removes first occurrence of the argument form the vector
13.	Int size()	Returns number of component in the vector
14.	void copyInto(Object[] array)	Copies the components of vector into specified array.

Program Code

```
import java.util.*;

public class Vector_Example

{

public static void main(String[] args)

{

Vector vec = new Vector(); vec.add(“Government ");

vec.add("Polytechnic");

vec.add("Solapur");

System.out.println("----------------------Elements in Vector are as follows----------------------");

System.out.println(vec);
```



Practical Related Questions:

1. **State difference between size() and capacity() method of Vector class.**
   The difference between capacity() and size() in java. util. Vector is that the size() is the number of elements which is currently hold and capacity() is the number of element which can maximum hold. A Vector is a dynamically grow able data structure, and it would reallocate its backing array as necessary.

2. **Differentiate between addElement() and insertElementAt() methods of Vector class.**
   addElement() method is used to add an object at the end of vector and increase the size of vector by one.
   insetElementAt() method is used to insert a particular element at specified index of the vector.

3. **Difference Between Array & Vector**

1. Array cant grow and shrink dynamically.
   Vector can grow and shrink dynamically.

2. Array is a static list of primitive data types.
   Vector can hold dynamic list of objects or primitive data types.

3. Array class is found in java.lang (default) package.
   Vector class is found in java.util package.

4. Array can store elements of same data type.
Vector can store elements of different data types.

5. For accessing elements of an array no special methods are available as it is nor a class, but a derived type.
   Vector class provides different methods for accessing and managing vector

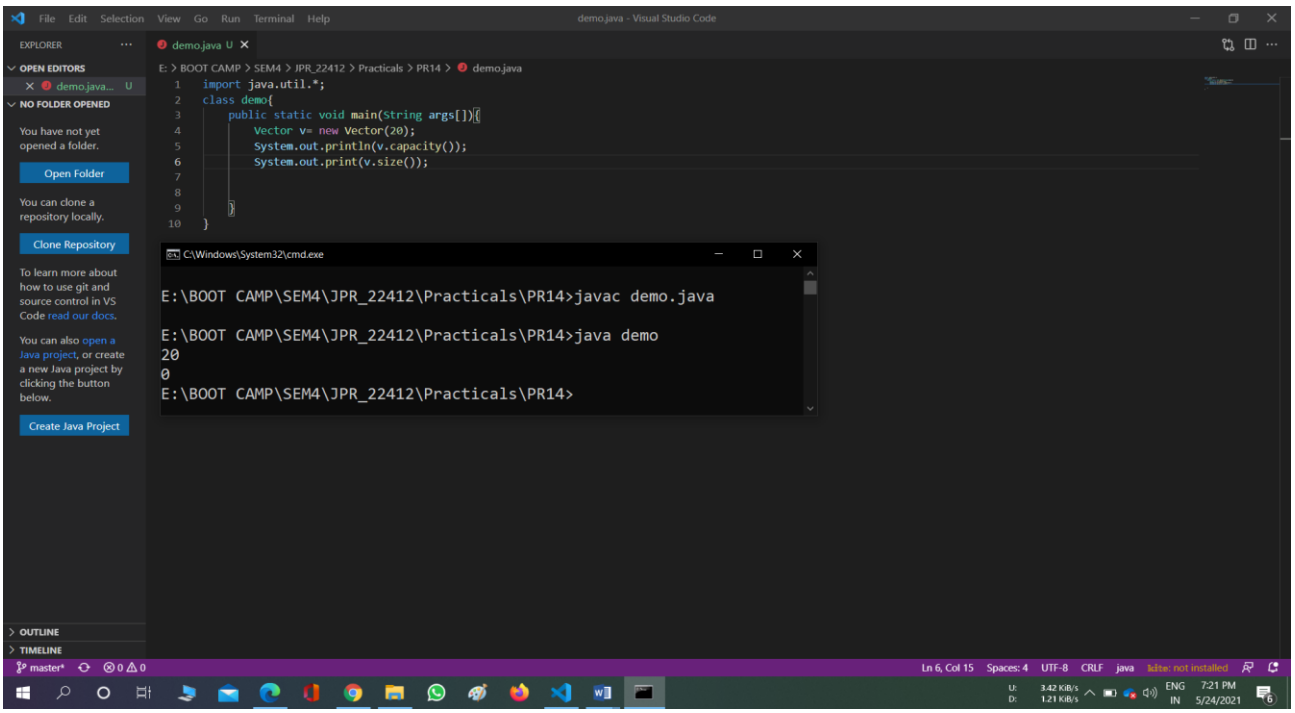
elements.

6. Syntax:

Datatype array_name[] = new datatype[size] Syntax:

Vector object_name = new Vector()

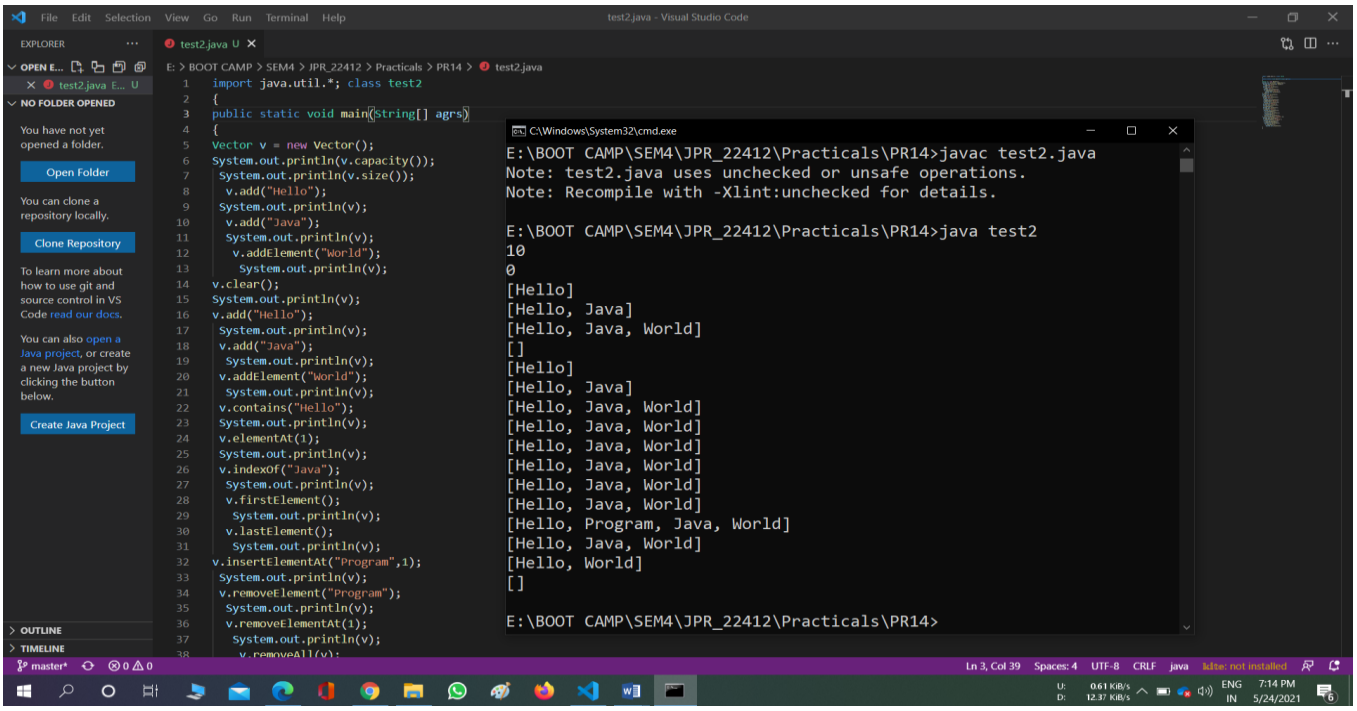
1)



2)

```
import java.util.*; class test2
{
public static void main(String[] agrs)
{
Vector v = new Vector();
System.out.println(v.capacity());
 System.out.println(v.size());
  v.add("Hello");
 System.out.println(v);
  v.add("Java");
  System.out.println(v);
   v.addElement("World");
    System.out.println(v);
v.clear();
System.out.println(v);
```

```
v.add("Hello");
 System.out.println(v);
 v.add("Java");
  System.out.println(v);
 v.addElement("World");
  System.out.println(v);
 v.contains("Hello");
 System.out.println(v);
 v.elementAt(1);
 System.out.println(v);
 v.indexOf("Java");
  System.out.println(v);
  v.firstElement();
   System.out.println(v);
  v.lastElement();
   System.out.println(v);
v.insertElementAt("Program",1);
 System.out.println(v);
 v.removeElement("Program");
  System.out.println(v);
  v.removeElementAt(1);
   System.out.println(v);
    v.removeAll(v);
     System.out.println(v);
}
}
```


File Edit Selection View Go Run Terminal Help
test2.java - Visual Studio Code
EXPLORER
test2.java U
E: > BOOT CAMP > SEM4 > JPR_22412 > Practicals > PR14 > test2.java
OPEN E...
test2.java E... U
NO FOLDER OPENED
You have not yet opened a folder.
Open Folder
You can clone a repository locally.
Clone Repository
To learn more about how to use git and source control in VS Code read our docs.
You can also open a Java project, or create a new Java project by clicking the button below.
Create Java Project
OUTLINE
TIMELINE
1 import java.util.*; class test2
2 {
3 public static void main(String[] agrs)
4 {
5 Vector v = new Vector();
6 System.out.println(v.capacity());
7 System.out.println(v.size());
8 v.add("Hello");
9 System.out.println(v);
10 v.add("Java");
11 System.out.println(v);
12 v.addElement("World");
13 System.out.println(v);
14 v.clear();
15 System.out.println(v);
16 v.add("Hello");
17 System.out.println(v);
18 v.add("Java");
19 System.out.println(v);
20 v.addElement("World");
21 System.out.println(v);
22 v.contains("Hello");
23 System.out.println(v);
24 v.elementAt(1);
25 System.out.println(v);
26 v.indexOf("Java");
27 System.out.println(v);
28 v.firstElement();
29 System.out.println(v);
30 v.lastElement();
31 System.out.println(v);
32 v.insertElementAt("Program",1);
33 System.out.println(v);
34 v.removeElement("Program");
35 System.out.println(v);
36 v.removeElementAt(1);
37 System.out.println(v);
C:\Windows\System32\cmd.exe
E:\BOOT CAMP\SEM4\JPR_22412\Practicals\PR14>javac test2.java
Note: test2.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
E:\BOOT CAMP\SEM4\JPR_22412\Practicals\PR14>java test2
10
0
[Hello]
[Hello, Java]
[Hello, Java, World]
[]
[Hello]
[Hello, Java]
[Hello, Java, World]
[Hello, Java, World]
[Hello, Java, World]
[Hello, Java, World]
[Hello, Java, World]
[Hello, Java, World]
[Hello, Program, Java, World]
[Hello, Java, World]
[Hello, World]
[]
E:\BOOT CAMP\SEM4\JPR_22412\Practicals\PR14>
master*
Ln 3, Col 39 Spaces: 4 UTF-8 CRLF java
ENG IN 7:14 PM 5/24/2021