PRAJICAL :-5

Aim :- Create debug and run Java programe based on wrapper class OR vector

Theory :-

Wrapper classes :- A wrapper class is a class whose object wraps OR contains primitive data type. when we create on object to a wropper class. it contains a field and in this field, we can strore primitive data types, In other wrapper we can wraps a Primitive value in a wrapper class object

Primitive Data Types	Wrapper class
char	character
bytes	Bute
short	short
int	Integen
long	long
Float	Float
double	Double

The eight classes of Java package are known as wrapper classes in Java

Autoboxing :- Autumic conversion of purimitive data types to the object of their Corrsponding wrapper classes is known as autoboxing Ex:- conversion of int into integer long to long, doble to double ec.

Unboxing :- It Just the reverse process of autoboxing Automatically converting an object of wrapper class to its Corresponding primitive types of known as unboxing Ex:- Conversion of Integen to Int Long to Long; Double to Double.

Unboxing in vector in Java :-
Vector is like the dynamic
array which can grow OR shrink to it's
size unlike array we can store number
of elements in it as there is no
size limit it is a part of Java
Collection forward since Java 1.2. It is
found in the savanth package and
Implement the therefore so we can
use all the methods of interface
Here.

DATE : __/__/__

PAGE No : ____

Java vector class decleration.

Public class vector <E>
extency object <E>
implements List <F> elements, Semizizate

Java vector constructory.

vector class supports 4 types of constru-
-tor

1] vector () :- It constructs on empty secto
with the deafult size of

2] vector [int intial capicity] :- If constortour on empty with the speified intial capacity.

3] Vector [int. intial capicity, int capacity inncement]

4] Vector [collection <? extends E >e]

Scanned with CamScanner

Program :-

(1) program :-

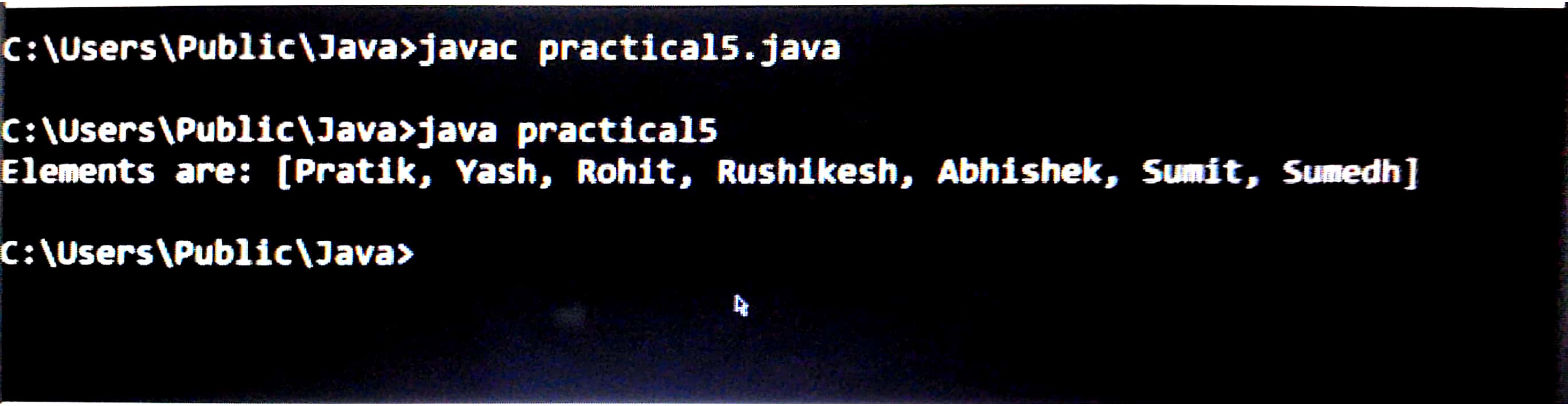
```
import   java.util.*;
class    practical5 {
   public  static  void  main  (String [] args ){
     Vector <String> vec  =  new  Vector <String> ();
      vec.add  ("Ashwin");
      vec.add  ("Pratik");
      vec.add  ("Yash");
      vec.add  ("Rohit");
      vec.add  ("Rushikesh");

      vec.addElement  ("Abhishek");
      vec.addElement  ("Sumit");
      vec.addElement  ("Sumedh");
     System.out.println ("Elements are : " + vec);
   }
}
```

(2) program :-

```
class  practical5_2 {
   public  static  void  main  (String [] args ){
    Double  d1 = new Double (2.71828);
    Double  d2 = new Double ("2.71828E-5");
    System.out.println (d1 + " = " + d2 + " --> " +
                                 d1.equals (d2));
   }
}
```

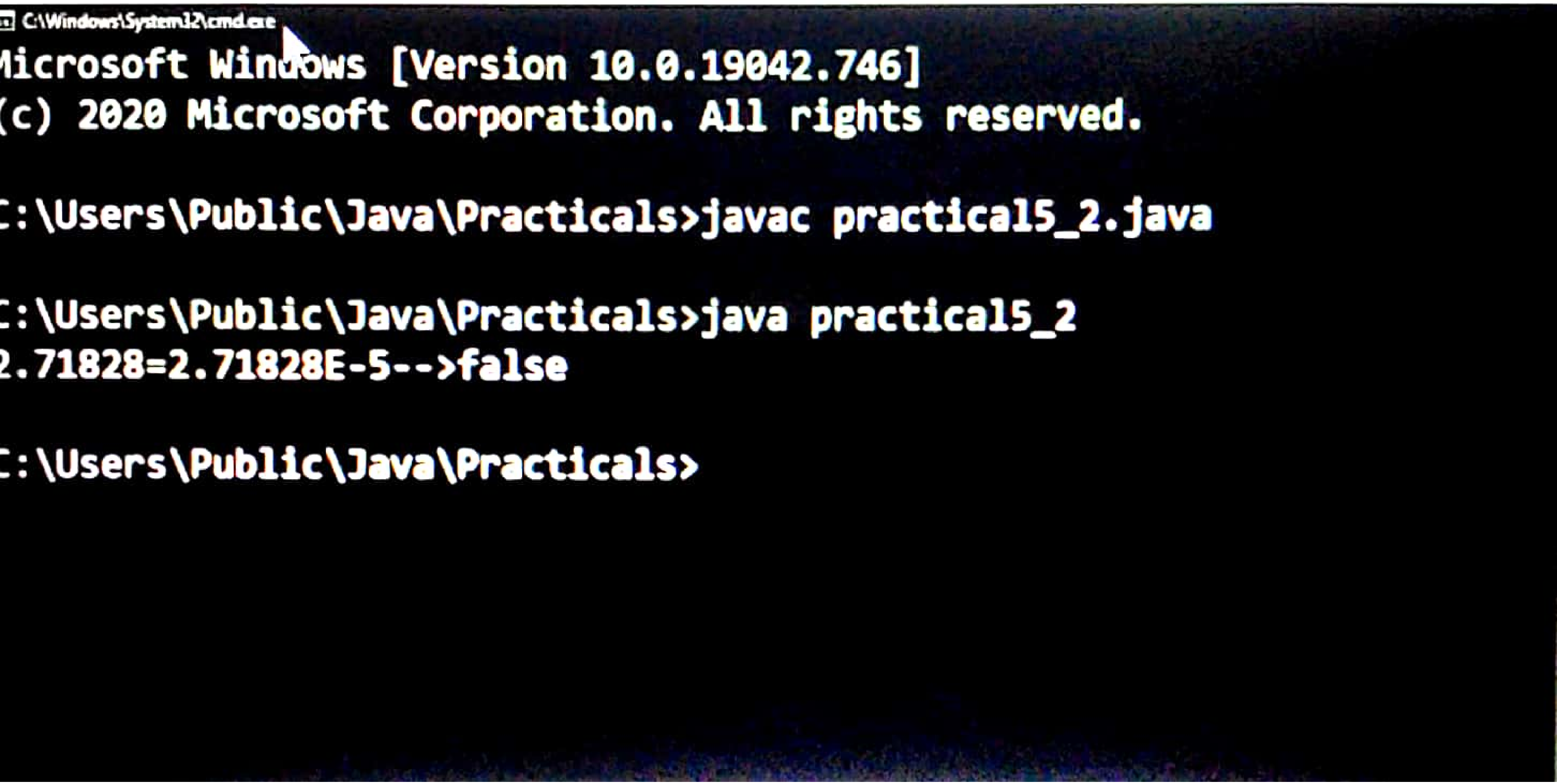
Conclusion:- Hence, we successfully create and run program on wrapper class and Vector.



```
C:\Users\Public\Java>javac practical5.java

C:\Users\Public\Java>java practical5
Elements are: [Pratik, Yash, Rohit, Rushikesh, Abhishek, Sumit, Sumedh]

C:\Users\Public\Java>
```



```
Microsoft Windows [Version 10.0.19042.746]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Public\Java\Practicals>javac practical5_2.java

C:\Users\Public\Java\Practicals>java practical5_2
2.71828=2.71828E-5-->false

C:\Users\Public\Java\Practicals>
```