|  |  |
| --- | --- |
| *A close up of a logo  Description automatically generated* | *DEPARTMENT OF COMPUTER ENGINEERING* |

|  |  |
| --- | --- |
| Semester | S.E. Semester III – Computer Engineering |
| Subject | Object Oriented Programming Using Java (Skill Based Lab) |
| Subject Professor In-charge | Prof. Indu Anoop |
| Laboratory | Online Lab |

|  |  |  |
| --- | --- | --- |
| Student Name | Trisha Shah | |
| Roll Number | 20102A0004 | |
| Grade and Subject Teacher’s Signature |  |  |

|  |  |  |
| --- | --- | --- |
| Experiment | 7C | |
| Problem Statement | WAP to demonstrate 2DArray, ArrayList and Vectors. | |
| Resources / Apparatus Required | Hardware: Computer System | Software: jdk 1.8, Eclipse / Notepad++/IntelliJ IDEA |
| Details | **Vector** is like the *dynamic array* which can grow or shrink its size. Unlike array, we can store n-number of elements in it as there is no size limit. It is a part of Java Collection framework since Java 1.2. It is found in the java.util package and implements the *List* interface, so we can use all the methods of List interface here.  It is recommended to use the Vector class in the thread-safe implementation only. If you don't need to use the thread-safe implementation, you should use the ArrayList, the ArrayList will perform better in such case. The Iterators returned by the Vector class are *fail-fast*. In case of concurrent modification, it fails and throws the Concurrent Modification Exception. It is similar to the ArrayList, but with two differences-   1. Vector is synchronized. 2. Java Vector contains many legacy methods that are not the part of a collections framework. | |
| Code | import java.util.Scanner;  import java.util.Vector;  public class VectorDemo {  public static void main(String[] args) {  int n,i,c;  System.out.println("Enter number of students=");  Scanner t=new Scanner(System.in);  n=t.nextInt();    //Create vector to add names from user    Vector v=new Vector();    for(i=0;i<n;i++) {  System.out.println("Enter student name=");  String s=t.next();  v.addElement(s);  }  while(true) {  System.out.println("MENU\n1.Insert new student\n2.Delete student name\n3.Display list\n4.Exit\nEnter choice=");  c=t.nextInt();  if(c==4)  break;  switch (c) {  case 1:System.out.println("Enter name and position");  String s1=t.next();  int p=t.nextInt();  v.insertElementAt(s1, p);  System.out.println("Name is inserted");  break;    case 2:System.out.println("Enter name of student to be deleted");  String s2=t.next();  v.removeElement(s2);  System.out.println("Name is deleted");  break;    case 3:for (i = 0; i < v.size(); i++) {  System.out.println(v.elementAt(i).toString());  }  break;    default:System.out.println("Invalid choice");  break;  }  }    }  } | |
| Output |  | |
| Conclusion | Thus, we successfully learnt how to implement vectors in our program. | |