| **Name:** | Mahadev Balla |
| --- | --- |
| **UID:** | 2023300010 |
| **Experiment No.** | 8B |

| **AIM:** | Implement a program to demonstrate multiple inheritance using interfaces. |
| --- | --- |
| **Program 1** | |
| **PROBLEM STATEMENT :** | Write a Java program to create an interface Sortable with a method sort() that sorts an array of integers in ascending order. Create two classes BubbleSort and SelectionSort that implement the Sortable interface and provide their own implementations of the sort() method. |
| **PROGRAM:** | import java.util.\*;  interface Sortable{  public void sort(int arr[]);  }  class BubbleSort implements Sortable{  public void sort(int arr[]){  for(int i=0; i<arr.length; i++){  for(int j=0; j<arr.length-i-1; j++){  if(arr[j]>arr[j+1]){  int temp = arr[j];  arr[j] = arr[j+1];  arr[j+1] = temp;  }  }  System.out.print("Sorted array after pass " + (i+1) + " : ");  for(int z=0; z<arr.length; z++){  System.out.print(arr[z] + " ");  }  System.out.println();  }  }  }  class SelectionSort implements Sortable{  public void sort(int arr[]){  for(int i=0; i<arr.length-1; i++){  int temp=i;  for(int j=i+1; j<arr.length; j++){  if(arr[j]<arr[temp]){  temp=j;  }  }  int smaller = arr[temp];  arr[temp] = arr[i];  arr[i] = smaller;  System.out.print("Sorted array after pass " + (i+1) + " : ");  for(int z=0; z<arr.length; z++){  System.out.print(arr[z] + " ");  }  System.out.println();  }  }  }  class sort{  public static void main(String args[]){  Scanner sc = new Scanner(System.in);  System.out.print("Enter the number of elements in the array : ");  int x = sc.nextInt();  int arr1[] = new int[x];  int arr2[] = new int[x];  for(int i=0; i<x; i++){  System.out.print("Enter element " + (i+1) + " : ");  arr1[i] = sc.nextInt();  arr2[i]=arr1[i];  }    BubbleSort s1 = new BubbleSort();  System.out.println("Bubble Sort -");  s1.sort(arr1);  System.out.print("Sorted Array : ");  print(arr1);  System.out.println();    SelectionSort s2 = new SelectionSort();  System.out.println("Selection Sort -");  s2.sort(arr2);  System.out.print("Sorted Array : ");  print(arr2);  System.out.println();  }  static void print(int arr[]){  for(int i=0; i<arr.length; i++){  System.out.print(arr[i] + " ");  }  }  } |
| **RESULT:** | |
| **Program 2** | |
| **PROBLEM STATEMENT :** | Write a Java program to create an interface Searchable with a method search() that searches for a given word in a text document. Create two classes Document and WebPage that implement the Searchable interface and provide their own implementations of the search() method, one searches a String and other a number. |
| **PROGRAM:** | import java.util.\*;  interface Searchable{  public void search(String s, String toBeSearched);  }  class Document implements Searchable{  public void search(String s, String search){  int count = 0;  int index = s.indexOf(search);  int firstocc =index;  while (index != -1) {  count++;  index = s.indexOf(search, index + 1);  }  if(count>0){  System.out.println("Index of first occurence of \'" + search + "\' is : " + firstocc);  System.out.println("Count of occurrences of \'" + search + "\'' is : " + count);  }  else{  System.out.println("Substring to be searched isn't present in the original string.");  }  }  }  class WebPage implements Searchable{  String web;  WebPage(String s){  this.web=s;  }  public void search(String s, String search){  int count = 0;  int index = s.indexOf(search);  int firstocc =index;  while (index != -1) {  count++;  index = s.indexOf(search, index + 1);  }  if(count>0){  System.out.println("Index of first occurence of \'" + search + "\' is : " + firstocc);  System.out.println("Count of occurrences of \'" + search + "\' is : " + count);  }  else{  System.out.println("Number to be searched isn't present in the original string.");  }  }  }  class search{  public static void main(String args[]){  Scanner sc = new Scanner(System.in);  int x;  System.out.print("Enter a documnetary string : ");  String doc = sc.nextLine();  System.out.print("Enter a string to be searched : ");  String search1 = sc.nextLine();  Document d1 = new Document();  d1.search(doc,search1);  do{  System.out.println("Do you want to check for the occurrence of any other string ?\n1. Yes\n2. No");  x = sc.nextInt();  switch (x) {  case 1: System.out.print("Enter a number to be searched : ");  int s3 = sc.nextInt();  String search23 = "" + s3;  d1.search(doc,search23);  break;    case 2: System.out.println("Thank you!!");    default: System.out.println("Invalid input!!");  }  }  while(x!=2);    System.out.print("Enter a web page string : ");  String web = sc.nextLine();  System.out.print("Enter a number to be searched : ");  int s = sc.nextInt();  String search2 = "" + s;  WebPage d2 = new WebPage(web);  d2.search(web,search2);  do{  System.out.println("Do you want to check for the occurrence of any other number ?\n1. Yes\n2. No");  x = sc.nextInt();  switch (x) {  case 1: System.out.print("Enter a number to be searched : ");  int s1 = sc.nextInt();  String search21 = "" + s1;  d2.search(web,search21);  break;    case 2: System.out.println("Thank you!!");    default: System.out.println("Invalid input!!");  }  }  while(x!=2);  }  } |
| **RESULT:** | |
| **CONCLUSION:** | Studied the implementation of multiple inheritance using interfaces to solve the given problems. |