20MCA132 OBJECT ORIENTED PROGRAMMING LAB

ASSIGNMENT-3

SUBMITTED BY

VIVIN V. ABRAHAM
R MCA-2020-S2
ROLL NO : 42

SUBMITTED TO ,

Sis. Elsin

Course Outcome 4(CO4)

1. Program to create a generic stack and do push and pop operations.

PROGRAM

```
import java.io.*;
import java.util.*;

public class Main{

static final int MAX=100;
int top=-1;

int[] stack=new int[MAX];

public static void main(String args[])
{
   Main s1=new Main();
      int opt, val,k=0;

      System.out.println("1. PUSH ");
      System.out.println("2. POP ");
      System.out.println("3. PEEP ");
      System.out.println("4. DISPLAY STACK ");
      System.out.println("5. EXIT ");
      do{
      System.out.println("\n Enter Your Option: ");
      Scanner s=new Scanner(System.in);
      opt=s.nextInt();

      switch(opt)
      {
         case 1: System.out.println("Enter the value to be added to the stack: ");
```

```
                val=s.nextInt();
                s1.push(val);
                break;

        case 2: s1.pop();
                break;

        case 3: s1.peep();
                break;

        case 4: s1.display();
                break;

        case 5:
            k++;
                break;
        default:
                System.out.println("enter valid option");
                break;

      }
   }while(k<=0);

}

public void push(int val)
{
   if(top==MAX-1)
   {
      System.out.println("Stack is FULL!");
   }
   else
   {
      top++;
```

```
        stack[top]=val;
        System.out.println("Element added to the stack is: "+val);
        display();
    }
}

public void pop()
{
    int x;
    if(top==-1)
    {
        System.out.println("Stack is EMPTY!");
    }
    else
    {
        x=stack[top];
        System.out.println("The element deleted from the stack is: "+x);
        top--;
        display();
    }
}

public void peep()
{
    int n;
    n=stack[top];
    System.out.println("The value at the top of the stack is: "+n);
}

public void display()
{
    int i;
    if(top==-1)
    System.out.println("STACK IS EMPTY!");
```

```
    else
    {
        for(i=0; i<=top; i++)
        System.out.println("The elements in the stack are: "+stack[i]);
    }


}


}
```

OUTPUT

```
1. PUSH
2. POP
3. PEEP
4. DISPLAY STACK
5. EXIT

 Enter Your Option:
1
Enter the value to be added to the stack:
2
Element added to the stack is: 2
The elements in the stack are: 2

 Enter Your Option:
1
Enter the value to be added to the stack:
3
Element added to the stack is: 3
The elements in the stack are: 2
The elements in the stack are: 3

 Enter Your Option:
3
The value at the top of the stack is: 3

 Enter Your Option:
2
The element deleted from the stack is: 3
The elements in the stack are: 2

 Enter Your Option:
4
The elements in the stack are: 2

 Enter Your Option:
```

2. Using generic method Bubble sort.

PROGRAM

```
import java.util.*;

class Main {

  static void bubbleSort(int array[]) {
    int size = array.length;

    for (int i = 0; i < size - 1; i++)

      for (int j = 0; j < size - i - 1; j++)

        if (array[j] > array[j + 1]) {


          int temp = array[j];
          array[j] = array[j + 1];
          array[j + 1] = temp;
        }
  }

  public static void main(String args[]) {
      int n;
      Scanner sc=new Scanner(System.in);
System.out.print("Enter the number of elements: ");

n=sc.nextInt();

int[] data = new int[n];
System.out.println("Enter the elements of the array: ");
for(int i=0; i<n; i++)
```

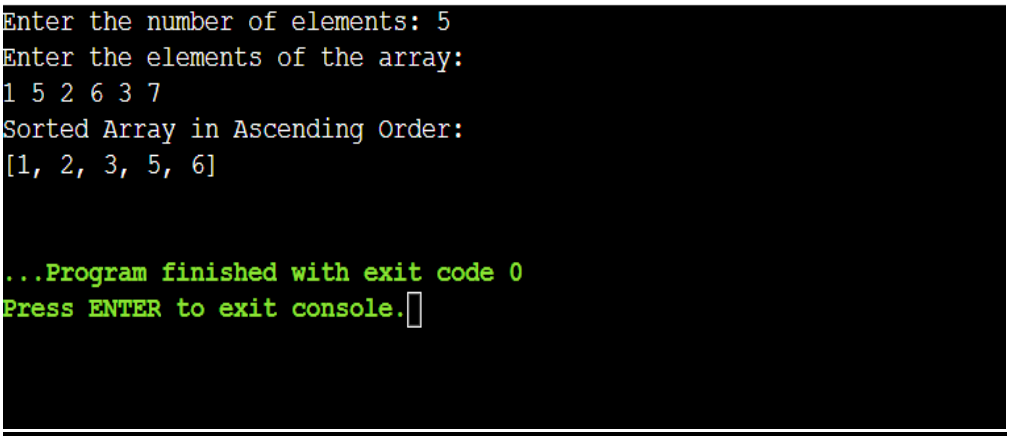
```
{

data[i]=sc.nextInt();
}

   Main.bubbleSort(data);

   System.out.println("Sorted Array in Ascending Order:");
   System.out.println(Arrays.toString(data));
  }
}
```

OUTPUT



3. Maintain a list of Strings Using ArrayList from Collection framework,perform builtin operations.

PROGRAM

```
import java.util.*;


public class Main {
  public static void main(String args[]) {


    ArrayList<String> obj = new ArrayList<String>();



    obj.add("jobin");
    obj.add("joice");
    obj.add("julia");
    obj.add("nimisha");
    obj.add("karthika");



    System.out.println("Original ArrayList:");
    for(String str:obj)
      System.out.println(str);
System.out.println("\n");


    obj.add(0, "sonu");
    obj.add(1, "pradeep");



    System.out.println("ArrayList after add operation:");
    for(String str:obj)
      System.out.println(str);


   System.out.println("\n");
    obj.remove("joice");
```

```
        obj.remove("karthika");



        System.out.println("ArrayList after remove operation:");
        for(String str:obj)
           System.out.println(str);
        obj.remove(1);
        System.out.println("\n");

         System.out.println("set index 2 as tom:");
        obj.set(2, "Tom");

        System.out.println("\n");
        System.out.println("index of nimisha");
        int pos = obj.indexOf("nimisha");
        System.out.println(pos);
        System.out.println("\n");

         System.out.println("check for jobin in arraylist");
        System.out.println(obj.contains("jobin"));
        System.out.println("\n");

      System.out.println("Number of items:");
       int numberofitems = obj.size();
       System.out.println(numberofitems);
      System.out.println("\n");

        System.out.println("Final ArrayList:");
        for(String str:obj)
           System.out.println(str);
    }
}
```

OUTPUT

```
Original ArrayList:
jobin
joice
julia
nimisha
karthika


ArrayList after add operation:
sonu
pradeep
jobin
joice
julia
nimisha
karthika


ArrayList after remove operation:
sonu
pradeep
jobin
julia
nimisha


set index 2 as tom:


index of nimisha
3


check for jobin in arraylist
true
```

```


Number of items:
4


Final ArrayList:
sonu
jobin
Tom
nimisha
```