**Name: VYSHNAV SURESH**

**Roll No:56**

**Batch:MCA-B**

**Date:31-05-2022**

**OBJECT ORIENTED PROGRAMMING LAB**

**Experiment No.: 12**

**Aim**

Program to create a generic stack and do the Push and Pop operations.

**Procedure**

import java.util.Scanner;

class CO46 {

private int arr[];

private int top;

private int capacity;

CO46(int size) {

arr = new int[size];

capacity = size;

top = -1;

}

public void push(int x) {

if (isFull()) {

System.out.println("Stack OverFlow");

System.exit(1);

}

System.out.println("Inserting " + x);

arr[++top] = x;

}

public int pop() {

if (isEmpty()) {

System.out.println("STACK EMPTY");

System.exit(1);

}

return arr[top--];

}

public int getSize() {

return top + 1;

}

public Boolean isEmpty() {

return top == -1;

}

public Boolean isFull() {

return top == capacity - 1;

}

public void printStack() {

for (int i = 0; i <= top; i++) {

System.out.print(arr[i] + ", ");

}

}

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

System.out.println("Enter the no elements in the stack:");

int n = s.nextInt();

CO46 stack = new CO46(n);

int x;

int f = 1;

while (f == 1) {

System.out.println(

"\n---Menu -----\n1.PUSH\n2.POP\n3.DISPLAY\n4.Exit");

int ch = s.nextInt();

switch (ch) {

case 1:

System.out.println("Enter the element to be inserted:");

x = s.nextInt();

stack.push(x);

break;

case 2:

stack.pop();

break;

case 3:

System.out.print("Stack: ");

stack.printStack();

break;

case 4:

f = 0;

break;

default:

System.out.println("Invalid output");

break;

}

}

}}

**Output Screenshot**



