1. **Write a java program to create an interface for stack of integers with abstract methods push, pop and display. Write an implementation of the above mentioned abstract methods for a fixed size stack and a dynamic size stack.**

interface IntStack {

void push(int value);

int pop();

void display();

}

class FixedSizeIntStack implements IntStack {

private final int[] stack;

private int top;

public FixedSizeIntStack(int size) {

stack = new int[size];

top = -1;

}

public void push(int value) {

if (top == stack.length - 1) {

System.out.println("Stack overflow!");

return;

}

stack[++top] = value;

}

public int pop() {

if (top == -1) {

System.out.println("Stack underflow!");

return -1;

}

return stack[top--];

}

public void display() {

if (top == -1) {

System.out.println("Stack is empty!");

return;

}

for (int i = top; i >= 0; i--) {

System.out.println(stack[i]);

}

}

}

class DynamicSizeIntStack implements IntStack {

private int[] stack;

private int top;

private final int initialSize;

public DynamicSizeIntStack(int size) {

stack = new int[size];

top = -1;

initialSize = size;

}

public void push(int value) {

if (top == stack.length - 1) {

int[] newStack = new int[stack.length \* 2];

System.arraycopy(stack, 0, newStack, 0, stack.length);

stack = newStack;

}

stack[++top] = value;

}

public int pop() {

if (top == -1) {

System.out.println("Stack underflow!");

return -1;

}

return stack[top--];

}

public void display() {

if (top == -1) {

System.out.println("Stack is empty!");

return;

}

for (int i = top; i >= 0; i--) {

System.out.println(stack[i]);

}

}

}

public class StackImplementation {

public static void main(String[] args) {

IntStack stack1 = new FixedSizeIntStack(5);

IntStack stack2 = new DynamicSizeIntStack(5);

stack1.push(10);

stack1.push(20);

stack1.push(30);

stack1.push(40);

stack1.push(50);

stack1.push(60);

stack2.push(10);

stack2.push(20);

stack2.push(30);

stack2.push(40);

stack2.push(50);

stack2.push(60);

stack1.display();

stack2.display();

}

}