**Java Assignment 6 : Part - 3**

Demonstrating How to create packages and using interface to create stacks

By:

Vaibhav Jaiswal

PRN:17070122071

Batch:CS-4

Branch: CSE

SY CSE (Batch of 2017-2021)

Code:

Stack Interface File in package

package pkg\_stack;

public interface IntStack{

void push(int item); // Stores an item into Stack

int pop(); // Retrieves an item from stack

}

The Main Driver File :

import java.util.\*;

import java.lang.\*;

import pkg\_stack.IntStack;

// An implementation of IntStack that uses fixed storage.

class FixedStack implements IntStack{

private int stck[];

private int tos;

public FixedStack (int size) {

stck = new int[size];

tos = -1;

}

public void printStack(){

for(int i=0;i<stck.length;i++)

System.out.print(stck[i]+"\t");

}

public void push(int item) {

if (tos == stck.length-1)

System.out.println("Stack is full.");

else

stck[++tos] =item;

}

public int pop (){

if (tos < 0) {

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

return 0;

}

else

return stck[tos--];

}

}

class DynStack implements IntStack {

private int stck[];

private int tos;

DynStack (int size) {

stck= new int[size];

tos = -1;

}

public void printStack(){

for(int i=0;i<stck.length;i++)

System.out.print(stck[i]+"\t");

}

public void push(int item){

if (tos==stck.length-1){

int temp[] = new int[stck.length\*2]; // double size

for (int i=0; i<stck.length; i++){

temp[i] = stck[i];

}

stck = temp;

stck[++tos]= item;

}

else

stck[++tos]=item;

}

public int pop(){

if(tos<0){

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

return 0;

}

else

return stck[tos--];

}

}

class Driver{

public static void main(String args[]){

FixedStack fs = new FixedStack(5);

System.out.println("Fixed Stack :");

fs.push(3);

fs.push(4);

fs.printStack();

System.out.println(fs.pop());

System.out.println(fs.pop());

fs.pop();

System.out.println("Fixed Stack ends");

System.out.println("Growable Stack :");

DynStack ds = new DynStack(2);

ds.push(1);

ds.push(2);

ds.push(3);

ds.push(4);

ds.push(5);

ds.printStack();

System.out.println("Growable Stack Ends");

}

}

**Output** :

