//Stack by composition

import java.util.EmptyStackException;

import java.util.LinkedList;

public class Plates {

private LinkedList list;

public Plates() {

list = new LinkedList();

}

public boolean empty() {

return list.isEmpty();

}

public void push(String string) {

list.addFirst(string);

}

public int size (){

return list.size();

}

public int search (String string){

return list.indexOf(string);

}

public Object set (int index, Object element ){

return list.set(index, element);

}

public void pushlast (String string){

list.addLast(string);

}

public Object pop() throws EmptyStackException {

if (empty())

throw new EmptyStackException();

else

return list.pollFirst();

}

public String toString() {

return list.toString();

}

///////////////////////////////////////////////////////////////////////////////////////////////

public static void main(String[] args) {

Plates stack = new Plates();

//Catch EmptyListException's

try {

System.out.println("Stack empty? " + stack.empty());

stack.push("Mario");

stack.push("Link");

stack.push("Adonis");

stack.push("Luigi");

stack.push("Rose");

stack.push("Zelda");

stack.push("Ash");

stack.push("Luna");

stack.push("Maria");

stack.push("Juan");

System.out.println("stack empty? " + stack.empty());

System.out.println(stack);

int i = stack.search("Rose");

System.out.println("Rose is in location "+i);

// System.out.println("Object to be replaced:" + stack.set(9, "Coffee"));

int l = stack.size()-1;

if (i == l ){

stack.pushlast("Mary");

stack.pop();

stack.set(8, "Rose");

stack.set(7, "Ms");

}

else if (i == 0){

stack.set(0, "Ms");

stack.set(1, "Rose");

stack.set(2, "Mary");

}

else{

stack.set(i-1, "Ms");

stack.set(i+1, "Mary");

}

System.out.println(stack);

}

catch (EmptyStackException e) {

System.err.println(e);

}

//See what happens if we pop empty stack without

//catching exception

stack.pop();

}

}