### Get minimum element from stack(medium)

You are given **N** elements and your task is to Implement a Stack in which you can get a minimum element in O(1) time.

**Example 1:**

**Input:**

push(2)

push(3)

pop()

getMin()

push(1)

getMin()

**Output:** 2 1

**Explanation:** In the first test case for

query

push(2)  Insert 2 into the stack.

  The stack will be {2}

push(3)  Insert 3 into the stack.

  The stack will be {2 3}

pop() Remove top element from stack

  Poped element will be 3 the

  stack will be {2}

getMin() Return the minimum element

  min element will be 2

push(1) Insert 1 into the stack.

The stack will be {2 1}

getMin() Return the minimum element

  min element will be 1

**Expected Time Complexity**: O(1) for all the 3 methods.  
**Expected Auxiliary Space**: O(1) for all the 3 methods.

**Constraints:**  
1 <= Number of queries <= 100  
1 <= values of the stack <= 100

**Company Tags**

[**Flipkart**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Flipkart) [**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**Snapdeal**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Snapdeal) [**D-E-Shaw**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=D-E-Shaw) [**FactSet**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=FactSet) [**Walmart**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Walmart) [**Goldman Sachs**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Goldman%20Sachs) [**SAP Labs**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=SAP%20Labs) [**Sapient**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Sapient) [**One9 7**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=One97)[**Grey Orange**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=GreyOrange) [**Kuliza**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Kuliza)

**Topic Tags**

[**Stack**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Stack) [**Data Structures**](https://practice.geeksforgeeks.org/explore/?category%5b%5d=Data%20Structures)

Java code

//{ Driver Code Starts

import java.util.\*;

class Get\_Min\_From\_Stack

{

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

int T = sc.nextInt();

while(T>0)

{

int q = sc.nextInt();

CodingMaxima g = new CodingMaxima ();

while(q>0)

{

int qt = sc.nextInt();

if(qt == 1)

{

int att = sc.nextInt();

g.push(att);

//System.out.println(att);

}

else if(qt == 2)

{

System.out.print(g.pop()+" ");

}

else if(qt == 3)

{

System.out.print(g.getMin()+" ");

}

q--;

}

System.out.println();

T--;

}

}

}

// } Driver Code Ends

class CodingMaxima

{

int minEle;

Stack<Integer> s=new Stack<>();

// Constructor

CodingMaxima ()

{

}

/\*returns min element from stack\*/

int getMin()

{

if(s.isEmpty())return -1;

return Collections.min(s);

}

/\*returns poped element from stack\*/

int pop()

{

if(s.isEmpty())return -1;

return s.pop();

}

/\*push element x into the stack\*/

void push(int x)

{

s.push(x);

}

}