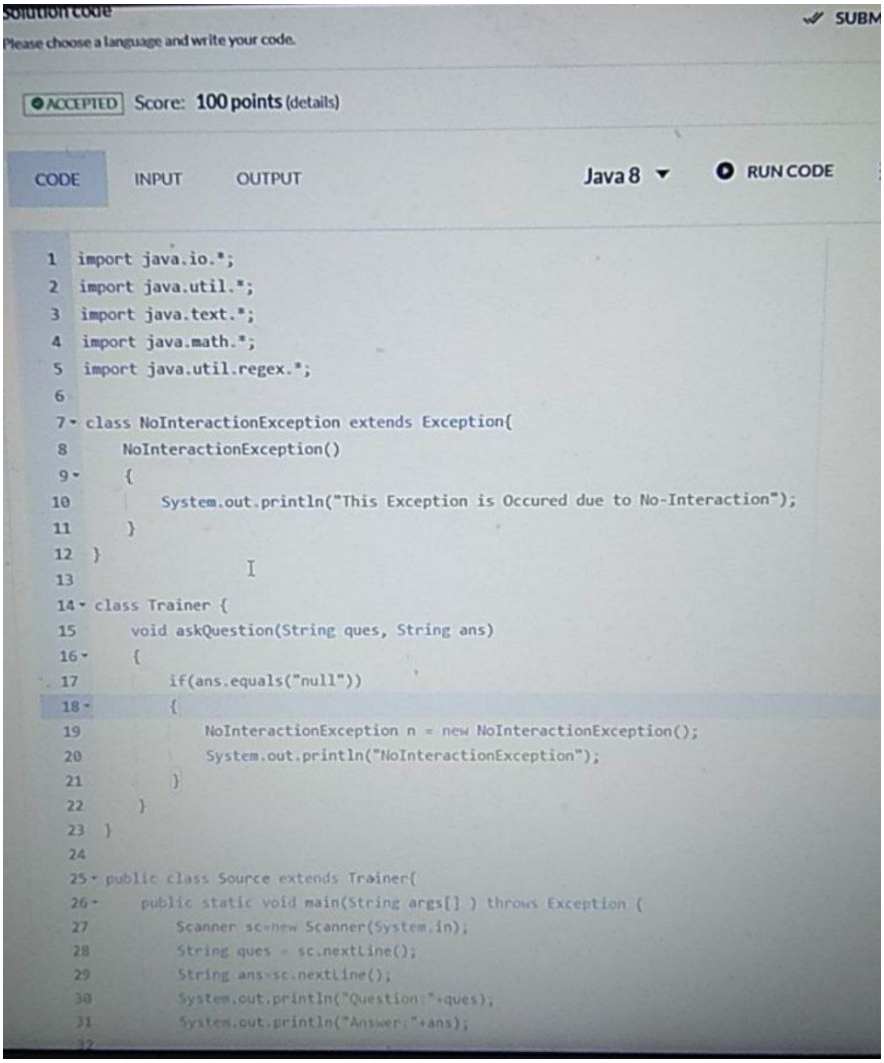
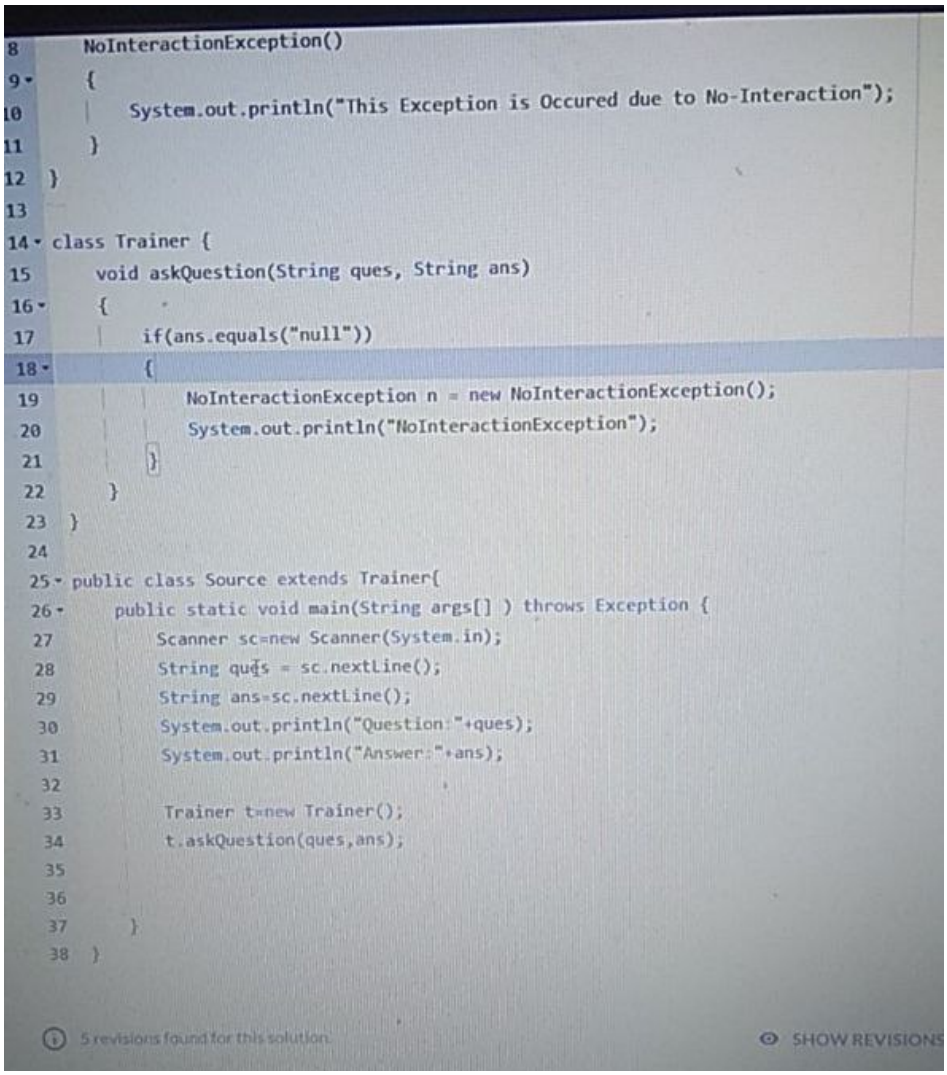
NoInteraction exception:



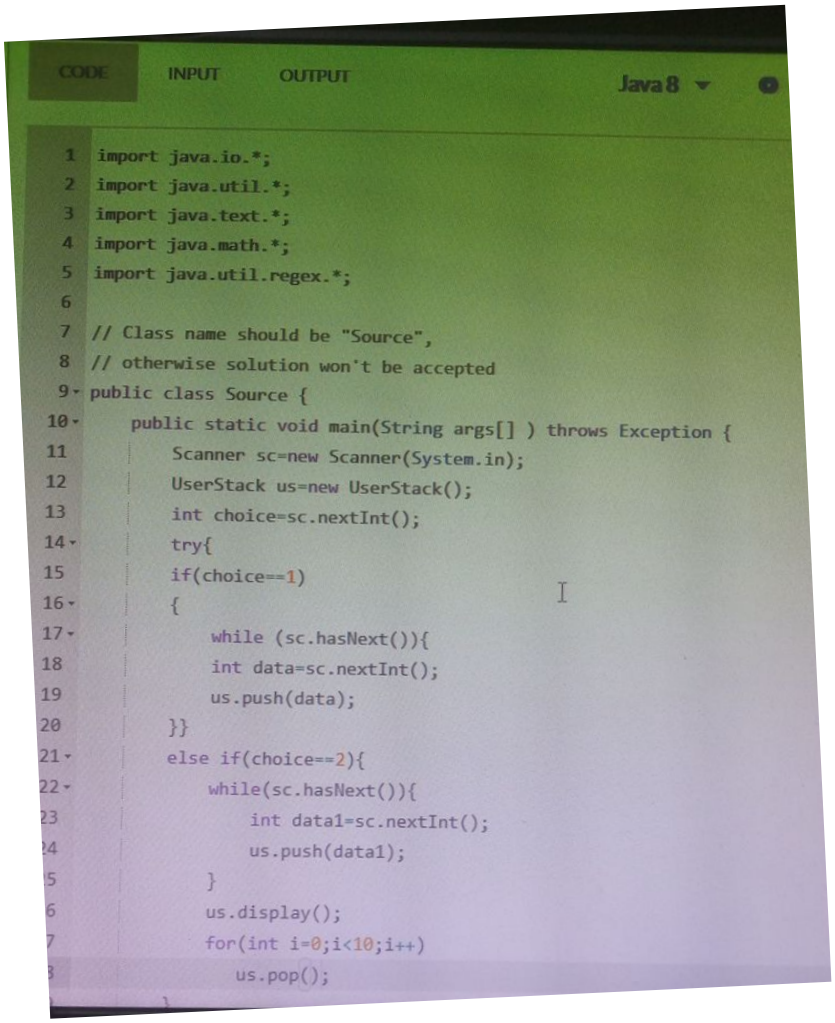


```
8       NoInteractionException()
9       {
10          System.out.println("This Exception is Occured due to No-Interaction");
11      }
12  }
13
14  class Trainer {
15      void askQuestion(String ques, String ans)
16      {
17          if(ans.equals("null"))
18          {
19              NoInteractionException n = new NoInteractionException();
20              System.out.println("NoInteractionException");
21          }
22      }
23  }
24
25  public class Source extends Trainer{
26      public static void main(String args[] ) throws Exception {
27          Scanner sc=new Scanner(System.in);
28          String ques = sc.nextLine();
29          String ans=sc.nextLine();
30          System.out.println("Question:"+ques);
31          System.out.println("Answer:"+ans);
32
33          Trainer t=new Trainer();
34          t.askQuestion(ques,ans);
35
36
37      }
38  }
```

5 revisions found for this solution

SHOW REVISIONS

Stack:



CODE INPUT OUTPUT Java 8

```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

// Class name should be "Source",
// otherwise solution won't be accepted
public class Source {
    public static void main(String args[] ) throws Exception {
        Scanner sc=new Scanner(System.in);
        UserStack us=new UserStack();
        int choice=sc.nextInt();
        try{
        if(choice==1)
        {
            while (sc.hasNext()){
            int data=sc.nextInt();
            us.push(data);
        }}
        else if(choice==2){
            while(sc.hasNext()){
                int data1=sc.nextInt();
                us.push(data1);
            }
            us.display();
            for(int i=0;i<10;i++)
               us.pop();
```

```
29              us.pop();
30          }
31          }catch(FullStackException e){
32              System.out.print(e.getMessage());
33          }
34          catch(EmptyStackException e){
35              System.out.print(e.getMessage());
36          }
37      }
38  }
39  class UserStack{
40      public UserStack(){};
41      int array1[]=new int[10];
42      public static int index=0;
43      public void push(int data) throws FullStackException{
44          if(index==9)
45              throw new FullStackException("Stack overflow");
46          array1[index]=data;
47          index++;
48      }
49      public int  pop() throws EmptyStackException{
50          if(index==0)
51              throw new EmptyStackException("Stack empty");
52          return array1[--index];
53
54      }
55      public void display(){
56          for(int i=0;i<index;i++)
57              System.out.println(array1[i]);
58      }
```

```
54      public void display(){
55          for(int i=0;i<index;i++)
56              System.out.println(array1[i]);
57      }
58
59  }
60  class EmptyStackException extends Exception{
61      public EmptyStackException(String message){
62          super(message);
63      }
64  }
65  class FullStackException extends Exception{
66      public FullStackException(String msg){
67          super(msg);
68      }
69  }
```