Queue using Arrays

Write a Java program to implement the Queue using arrays. Write insert(), delete(), and display() methods to demonstrate its working.

Input Format

3 1 53 1 68 1 20 2 2 2 3 4

Constraints

Size of Queue should be always positive

Output Format

Inserted Element is 53 Inserted Element is 68 Inserted Element is 20 Dequeued Element is 53 Dequeued Element is 68 Dequeued Element is 20 Queue is Empty

Sample Input 0

```
3
1
53
1
68
1
20
2
2
2
3
4
```

Sample Output 0

```
Inserted Element is 53
Inserted Element is 68
Inserted Element is 20
Dequeued Element is 53
Dequeued Element is 68
Dequeued Element is 20
Queue is Empty
```

Difficulty: Medium

Rate This Challenge:

More

Java 7

```java
//224G1A0553
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;
public class Solution {
    static int front = -1, rear = -1;
    static int arr[] = new int[50];
    public void insert(int num, int ele) {
        if (rear == num - 1) {
            System.out.println("Queue is overflow");
        } else {
            rear++;
            arr[rear] = ele;
            System.out.println("Inserted Element is "+ele);
        }
        if (front == -1) {
            front++;
        } }
    public void delete() {
        if (front == -1) {
            System.out.println("Queue is underflow");
        } else {
            System.out.println("Dequeued Element is "+ arr[front]);
            if (front == rear) {
                front = rear = -1;
            } else {
                front++;
            }} }
    public void display(int[] arr, int num) {
        if (rear == -1 && front == -1) {
            System.out.print("Queue is Empty");
        } else {
            System.out.print("ELEMENTS : ");
            for (int i = front; i <= rear; i++) {
                System.out.print(arr[i] + " ");
            } }
        System.out.println();
    }
    public static void main(String[] args) {
        Solution qe = new Solution();
        Scanner sc = new Scanner(System.in);
        int num, opt;
        int ele;
        //System.out.print("Enter size of Queue : ");
        num = sc.nextInt();
        Boolean kl = true;
        //System.out.println("Options\n1.INSERT\t2.DELETE\t3.DISPLAY\t4.EXIT");
        while (kl) {  //System.out.print("Enter option :");
            opt = sc.nextInt();
            switch (opt) {
            case 1: //System.out.print("Enter element : ");
                ele = sc.nextInt();
                qe.insert(num, ele);
                break;
            case 2:
```

```
58                  qe.delete();
59                  break;
60              case 3:
61                  qe.display(arr, num);
62                  break;
63              case 4:
64                  kl = false;   //System.out.println("EXIT");
65                  break;
66              default : //System.out.println("Enter a valid option between 1-4");
67                  break;
68              }}}}
```

Line: 20 Col: 10

Upload Code as File Test against custom input Run Code Submit Code

Testcase 0 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
3
1
53
1
68
1
20
2
2
2
3
4
```

Your Output (stdout)

```
Inserted Element is 53
Inserted Element is 68
Inserted Element is 20
Dequeued Element is 53
Dequeued Element is 68
Dequeued Element is 20
Queue is Empty
```

Expected Output

```
Inserted Element is 53
Inserted Element is 68
Inserted Element is 20
Dequeued Element is 53
Dequeued Element is 68
Dequeued Element is 20
Queue is Empty
```