

Credit
3110
Assignment
E4QueueList

How has your program changed from planning to coding to now? Please explain?

```
13 public class QueueList
14 {
15     private Node front;
16     private Node rear;
17     private int size;
18
19     public QueueList()
20     {
21         front = null;
22         rear = null;
23         size = 0;
24     }
25
26     // adds item to rear of queue
27     public void enqueue(String item)
28     {
29         Node newNode = new Node(item);
30
31         if (isEmpty())
32         {
33             front = newNode;
34             rear = newNode;
35         }
36         else
37         {
38             rear.next = newNode;
39             rear = newNode;
40         }
41
42         size++;
43     }
44
45     // adds item to front of queue
46     public void enqueueFront(String item)
47     {
48         Node newNode = new Node(item);
49
50         if (isEmpty())
51         {
52             front = newNode;
53             rear = newNode;
54         }
55         else
56         {
57             newNode.next = front;
58             front = newNode;
59         }
60
61         size++;
62     }
63 }
```

```

49
50     if (isEmpty())
51     {
52         front = newNode;
53         rear = newNode;
54     }
55     else
56     {
57         newNode.next = front;
58         front = newNode;
59     }
60
61     size++;
62 }
63
64 // removes item from front
65 public String dequeue()
66 {
67     String item = null;
68
69     if (!isEmpty())
70     {
71         item = front.data;
72         front = front.next;
73         size--;
74
75         if (front == null)
76         {
77             rear = null;
78         }
79     }
80
81     return item;
82 }
83
84 // checks if queue empty
85 public boolean isEmpty()
86 {
87     return size == 0;
88 }
89
90 // returns queue size
91 public int size()
92 {
93     return size;
94 }
95
96 // node class
97 private class Node

```

```

89
90 // returns queue size
91 public int size()
92 {
93     return size;
94 }
95
96 // node class
97 private class Node
98 {
99     private String data;
100     private Node next;
101
102     public Node(String d)
103     {
104         data = d;
105         next = null;
106     }
107 }
108 }
109

```

```

5 public class QueueListTest
6 {
7     public static void main(String[] args)
8     {
9         Scanner input = new Scanner(System.in);
10        QueueList queue = new QueueList();
11
12        int choice = 0;
13
14        while (choice != 4)
15        {
16            System.out.println("\nQUEUE MENU");
17            System.out.println("1. Add item");
18            System.out.println("2. Remove item (front)");
19            System.out.println("3. Display size");
20            System.out.println("4. Exit");
21            System.out.print("Enter choice: ");
22
23            choice = input.nextInt();
24            input.nextLine(); // clear buffer
25
26            if (choice == 1)
27            {
28                System.out.println("Add to:");
29                System.out.println("1. Front");
30                System.out.println("2. Rear");
31                System.out.print("Enter choice: ");
32
33                int addChoice = input.nextInt();
34                input.nextLine();
35
36                System.out.print("Enter string: ");
37                String item = input.nextLine();
38
39                if (addChoice == 1)
40                {
41                    queue.enqueueFront(item);
42                }
43                else if (addChoice == 2)
44                {
45                    queue.enqueue(item);
46                }
47            }
48            else if (choice == 2)
49            {
50                if (!queue.isEmpty())
51                {
52                    System.out.println("Removed: " + queue.dequeue());
53                }
54            }
55            else if (choice == 3)
56            {
57                if (!queue.isEmpty())
58                {
59                    System.out.println("Removed: " + queue.dequeue());
60                }
61                else
62                {
63                    System.out.println("Queue is empty.");
64                }
65            }
66            else if (choice == 4)
67            {
68                System.out.println("Queue size: " + queue.size());
69            }
70        }
71        input.close();
72    }
73 }

```