

Write a Java program to implement Queue Using Array And Class

```
public class QueueUsingArrayMain {

    private int capacity;

    int queueArr[];

    int front;

    int rear;

    int currentSize = 0;

    public QueueUsingArrayMain(int sizeOfQueue) {

        this.capacity = sizeOfQueue;

        front = 0;

        rear = -1;

        queueArr = new int[this.capacity];

    }

    public void enqueue(int data) {

        if (isFull()) {

            System.out.println("Queue is full!! Can not add more elements");

        } else {

            rear++;

            if (rear == capacity - 1) {

                rear = 0;

            }

        }

    }

}
```

```
        queueArr[rear] = data;

        currentSize++;

        System.out.println(data + " added to the queue");
    }
}
```

```
public void dequeue() {
    if (isEmpty()) {
        System.out.println("Queue is empty!! Can not dequeue element");
    } else {
        front++;
        if (front == capacity - 1) {
            System.out.println(queueArr[front - 1] + " removed from the queue");
            front = 0;
        } else {
            System.out.println(queueArr[front - 1] + " removed from the queue");
        }
        currentSize--;
    }
}
```

```
public boolean isFull() {
    if (currentSize == capacity) {
```

```
        return true;
    }
    return false;
}
```

```
public boolean isEmpty() {

    if (currentSize == 0) {
        return true;
    }
    return false;
}
```

```
public static void main(String a[]) {

    QueueUsingArrayMain queue = new QueueUsingArrayMain(5);
    queue.enqueue(6);
    queue.dequeue();
    queue.enqueue(3);
    queue.enqueue(99);
    queue.enqueue(56);
    queue.dequeue();
    queue.enqueue(43);
    queue.dequeue();
}
```

```

        queue.enqueue(89);

        queue.enqueue(77);

        queue.dequeue();

        queue.enqueue(32);

        queue.enqueue(232);

    }

}

```

The screenshot shows a web browser window with the URL `tutorialspoint.com/compile_java_online.php`. The page title is "Compile and Execute Java Online (JDK 1.8.0)". The interface includes a "Execute" button and tabs for "Share", "Source File", and "STDIN". The main area displays Java code for a queue implementation. The "Result" tab on the right shows the output of the program, which consists of a series of "added to the queue" and "removed from the queue" messages corresponding to the operations in the code.

```

50         return true;
51     }
52     return false;
53 }
54
55
56 public boolean isEmpty() {
57
58     if (currentSize == 0) {
59         return true;
60     }
61     return false;
62 }
63
64 public static void main(String a[]) {
65
66     QueueUsingArrayMain queue = new QueueUsingArrayMain(5);
67     queue.enqueue(6);
68     queue.dequeue();
69     queue.enqueue(3);
70     queue.enqueue(99);
71     queue.enqueue(56);
72     queue.dequeue();
73     queue.enqueue(43);
74     queue.dequeue();
75     queue.enqueue(89);
76     queue.enqueue(77);
77     queue.dequeue();
78     queue.enqueue(32);
79     queue.enqueue(232);
80 }
81 }
82

```

Result

```

$javac QueueUsingArrayMain.java

$java -Xmx128M -Xms16M QueueUsingArrayMain

6 added to the queue
6 removed from the queue
3 added to the queue
99 added to the queue
56 added to the queue
3 removed from the queue
43 added to the queue
99 removed from the queue
89 added to the queue
77 added to the queue
56 removed from the queue
32 added to the queue
232 added to the queue

```