

## [Create] Queue

```
public class Queue {

    Node head;
    Node tail;
    int size = 0;

    public Queue() {
        head = null;
        tail = null;
        size = 0;
    }

    public void enqueue(String data) {
        Node newNode = new Node(data);

        if (head == null) {
            head = newNode;
            tail = newNode;
        } else {
            tail.nextNode = newNode;
            tail = newNode;
        }
        size++;
    }

    public Node dequeue() {
        if (head == null) return null;
        Node currNode = head;
        head = head.nextNode;
        currNode.nextNode = null;
        if (head == null) tail = null;

        return currNode;
    }

    @Override
    public String toString() {
```

```

        String result = "";
        Node tempNode = head;
        while (tempNode != null) {
            result += " " + tempNode.data;
            tempNode = tempNode.nextNode;
        }
        return result;
    }

    public static void main(String[] array) {
        Queue q = new Queue();
        q.enqueue("A");
        q.enqueue("B");
        System.out.println("Queue : " + q.toString());
        q.dequeue();
        q.dequeue();
        System.out.println("Queue : " + q.toString());
        q.enqueue("C");
        q.enqueue("D");
        q.enqueue("E");
        System.out.println("Queue : " + q.toString());
        q.enqueue("E");
        q.dequeue();
        q.dequeue();
        System.out.println("Queue : " + q.toString());
    }
}

```