**Q1.**

interface QueueImpl {

    void insert(int value) throws Exception;

    void delete() throws Exception;

    void display();

}

class QueueDemo implements QueueImpl {

    private int[] queue;

    private int front, rear, size;

    private static final int MAX\_SIZE = 10;

    public QueueDemo() {

        queue = new int[MAX\_SIZE];

        front = -1;

        rear = -1;

        size = 0;

    }

    @Override

    public void insert(int value) throws Exception {

        if (size == MAX\_SIZE) {

            throw new Exception("Queue Overflow! Cannot insert more elements.");

        }

        if (rear == MAX\_SIZE - 1) {

            rear = -1;

        }

        queue[++rear] = value;

        size++;

        System.out.println(value + " has been inserted into the queue.");

    }

    @Override

    public void delete() throws Exception {

        if (size == 0) {

            throw new Exception("Queue Underflow! No elements to delete.");

        }

        int deletedValue = queue[front + 1];

        if (front == MAX\_SIZE - 1) {

            front = -1;

        }

        front++;

        size--;

        System.out.println(deletedValue + " has been deleted from the queue.");

    }

    @Override

    public void display() {

        if (size == 0) {

            System.out.println("Queue is empty.");

            return;

        }

        System.out.print("Current queue: ");

        for (int i = front + 1; i <= rear; i++) {

            System.out.print(queue[i] + " ");

        }

        System.out.println();

    }

}

public class QueueDemoTest {

    public static void main(String[] args) {

        QueueDemo queue = new QueueDemo();

        try {

            queue.insert(5);

            queue.insert(10);

            queue.insert(15);

            queue.insert(20);

            queue.insert(25);

            queue.display();

            queue.delete();

            queue.delete();

            queue.display();

            queue.insert(30);

            queue.insert(35);

            queue.display();

            queue.delete();

            queue.delete();

            queue.delete();

            queue.delete();

            queue.delete();

            queue.delete();

        } catch (Exception e) {

            System.out.println("Exception: " + e.getMessage());

        }

    }

}