```java
//1. Implement a simple task queue using LinkedList as a Queue.
package Queue_Interface;
import java.util.LinkedList;
import java.util.Queue;
public class Challenge1 {
	public static void main(String[] args) {
		Queue<String> taskQueue = new LinkedList<>();
	taskQueue.add("Task1: Write a basic java program");
	taskQueue.add("Task2: Write an operators java program");
	taskQueue.add("Task3: Explain about Collection Framework");
	System.out.println("Tasks:");
	for (String task : taskQueue) {
	   System.out.println(task);
	}
	}
}
```

```java
//2. Demonstrate how to add and remove elements using offer() and poll().
package Queue_Interface;
import java.util.LinkedList;
import java.util.Queue;
public class Challenge2 {
	public static void main(String[] args) {
		Queue<String> queue = new LinkedList<>();
	queue.offer("Alice");
	queue.offer("Bob");
	queue.offer("John");
	System.out.println("Queue after adding elements: " + queue);
	queue.poll();
	System.out.println("Queue after removing an element: " + queue);
	}
}
```

```java
//3. Use a PriorityQueue to order tasks by priority (integers).
package Queue_Interface;
import java.util.PriorityQueue;
public class Challenge3 {
	public static void main(String[] args) {
		PriorityQueue<Integer> taskQueue = new PriorityQueue<>();
		taskQueue.add(2);
		taskQueue.add(6);
		taskQueue.add(1);
		taskQueue.add(3);
		if(taskQueue.isEmpty()) {
			System.out.println("The Queue is empty.");
```

```
                return;
            }
            System.out.println("Order tasks by priority:");
            while (!taskQueue.isEmpty()) {
               System.out.println(taskQueue.poll());
            }
        }
}
```

```
//4. Simulate a print queue system where print jobs are processed in order
package Queue_Interface;
import java.util.LinkedList;
import java.util.Queue;
public class Challenge4 {
        public static void main(String[] args) {
                Queue<String> jobs = new LinkedList<>();
                jobs.add("Java Developer");
                jobs.add("Full-Stack Developer");
                jobs.add("Software Engineer");
    System.out.println("Jobs:");
    for (String job : jobs) {
       System.out.println(job);
    }
        }
}
```

```
//5. Create a ticket booking system where customer names are added to a queue and served in
order.
package Queue_Interface;
import java.util.LinkedList;
import java.util.Queue;
import java.util.Scanner;
public class Challenge5 {
        public static void main(String[] args) {
                Scanner sc = new Scanner(System.in);

                Queue<String> jobs = new LinkedList<>();

                System.out.print("Enter the number of customers: ");
                int n = sc.nextInt();

                for(int i=1; i<=n; i++) {
                        System.out.print("Enter customer "+i+" name: ");
                        String customer = sc.next();
                        jobs.add(customer);
```

```
                }

                System.out.println("Customers to be served in order are: ");
        for (String job : jobs) {
            System.out.println(job);
        }
        sc.close();
            }
}
```