//fa19-bse-070

//Muhammad Hamza

package queue.to.linkedlist;

public class QueueToLinkedList {

Node front;

Node rear;

public QueueToLinkedList ()

{

front = null;

rear = null;

}

private class Node{

int n;

Node next;

Node(int n){

this.n = n;

}

public void displayData(){

System.out.println("n= " + n);

}

}

public void insertLast(int n){

Node newNode = new Node(n);

if(isEmpty()){

front = newNode;

}else{

rear.next = newNode;

}

rear = newNode;

}

public int removeFirst(){

int temp = front.n;

if(front.next == null){

rear = null;

}

front = front.next;

return temp;

}

public void displayList(){

Node current = front;

while(current != null){

current.displayData();

current = current.next;

}

}

public int nodeData(){

return front.n;

}

public boolean isEmpty(){

return front == null;

}

public void insert(int item){

insertLast(item);

}

public int remove(){

if(isEmpty()){

throw new RuntimeException("Queue is empty..");

}

return removeFirst();

}

public int peek(){

if(isEmpty()){

throw new RuntimeException("Queue is empty..");

}

return nodeData();

}

public static void main(String[] args) {

System.out.println("\*\*\*DS lab Assignment\*\*\* Muhammad Hamza (FA19-BSE-070)");

QueueToLinkedList queue = new QueueToLinkedList ();

queue.insert(4);

queue.insert(8);

System.out.println("\*\*\* Displaying Queue data\*\*\*");

queue.displayList();

System.out.println("Item peeked = " + queue.peek());

System.out.println("\*\*\* Removing Queue elements \*\*\*");

System.out.println("Item removed= " + queue.remove());

System.out.println("Item removed= " + queue.remove());

}

}