Queue Using Linked List

```
import java.util.Scanner;

class Node {

    int info;
    Node link;
}

public class QueueLinkedList
{
      static Node front=null;
      static Node rear=null;

      //Inserting an element to the Queue
      public static void enqueue(int x)
    {

        Node p = new Node();
        p.info = x;
        p.link = null;

        if(rear==null)
            front=rear=p;
        else
        {
        rear.link = p;
        rear=p;
        }
    }

      //Removing an element from the Queue
      public static void dequeue()
    {
        if(front==null)
        {
            System.out.println("Queue Underflow ");
            return;
        }
        System.out.println("Deleted value is:"+front.info);
        front=front.link;
        if(front==null)
            rear=null;
    }

      //Displaying the Queue elements
      public static void display()
    {
        if(front==null)
        {
            System.out.println("Queue Underflow ");
            return;
        }
        Node temp=front;
        while(temp!=null)
        {
        System.out.print(temp.info+"-->");
        temp=temp.link;
        }
```

```
            System.out.println("null");
        }


          public static void main(String[] args) {
                Scanner sc=new Scanner (System.in);

                while(true)
                {
                System.out.println("****MENU*****");
                System.out.println("0:Exit");
                System.out.println("1:Enqueue");
                System.out.println("2:Dequeue");
                System.out.println("3:Display");

                System.out.println("*************");
                System.out.println("Enter the choice");
                int choice=sc.nextInt();
                switch(choice)
                {
                case 0:
                System.exit(0);
                case 1:
                      System.out.println("Enter the element:");
                      int e=sc.nextInt();
                      enqueue(e);
                      break;
                case 2:
                      dequeue();
                      break;
                case 3:
                      display();
                      break;
                default:
                System.out.println("Wrong choice");
                }
                }
          }

}
```

```
import java.util.Scanner;

public class QueueUsingArray {
     int arr[];
     int front,rear;
     QueueUsingArray(int max)
     {
          arr=new int[max];
          front=rear=-1;

     }

     void enqueue(int ele)
     {
          if(is_full())
          {
               System.out.println("Queue Overflow");
               return;
          }
          else if(front==-1&&rear==-1)
                    front=rear=0;
               else
                    rear=rear+1;

               arr[rear]=ele;
     }

     void dequeue()
     {
          if(is_empty())
          {
               System.out.println("Queue Underflow");
               return;
          }
          else
          {
              System.out.println("Deleted element:"+arr[front]);
               if(front==rear)
                    front=rear=-1;
               else
                    front=front+1;
          }
     }
     void display()
     {
```

```
        if(is_empty())
        {
            System.out.println("Queue Underflow");
            return ;
        }
        else
        {
            for(int i=front;i<=rear;i++)
                System.out.print(arr[i]+" ");
System.out.println();
        }
}

boolean is_empty()
{
        if(front==-1&&rear==-1)
            return true;
        else
            return false;
}

boolean is_full()
{
        if(rear==arr.length-1)
            return true;
        else
            return false;

}


public static void main(String[] args) {
        Scanner sc=new Scanner (System.in);
        System.out.println("Enter the size of Queue");
        int s=sc.nextInt();
        QueueUsingArray ob=new QueueUsingArray(s);
        while(true)
        {
        System.out.println("****MENU*****");
        System.out.println("0:Exit");
        System.out.println("1:Enqueue");
        System.out.println("2:Dequeue");
        System.out.println("3:Display");
        System.out.println("*************");
        System.out.println("Enter the choice");
```

```
            int choice=sc.nextInt();
            switch(choice)
            {
            case 0:
            System.exit(0);
            case 1:
            System.out.println("Enter the element:");
            int e=sc.nextInt();
            ob.enqueue(e);
            break;
            case 2:
            ob.dequeue();
            break;
            case 3:
            ob.display();
            break;
            default:
            System.out.println("Wrong choice");
            }
        }
    }
}
```