

```
1  import java.util.*;
2
3  class MyPriorityQueue <E>
4  {
5      private ArrayList <E> items;
6
7      public MyPriorityQueue ()
8      {
9          items = new ArrayList <E> ();
10     }
11
12     public boolean isEmpty ()
13     {
14         return true;
15     }
16
17     public boolean add(E x)
18     {
19         return false;
20     }
21
22     public E remove ()
23     {
24         return null;
25     }
26
27     public E peek ()
28     {
29         return null;
30     }
31 }
32
33 class Patient implements Comparable
34 {
35     private String name;
36     private int health; // a number from 1 to 100 with 1 meaning poor health
37
38     public Patient (String n, int h)
39     {
40         name = n;
41         health = h;
42     }
43
44     public String getName ()
45     {
46         return name;
47     }
48
49     public int getHealth ()
50     {
51         return health;
52     }
53
54     public String toString ()
55     {
56         return getName ();
57     }
58
59     public int compareTo (Object other)
60     {
61         return getHealth () - ((Patient) other).getHealth ();
62     }
63
64 }
65
66
67 public class PriorityQueueImplementationProject
```

```
68     {
69         public static void main (String [] args)
70         {
71             MyPriorityQueue <Patient > patients = new MyPriorityQueue <Patient > ();
72             patients.add (new Patient ("bill", 80));
73             patients.add (new Patient ("mary", 44));
74             patients.add (new Patient ("jane", 6));
75             patients.add (new Patient ("sam", 13));
76             System.out.println ("The sickest patient is " + patients.peek ());
77
78             System.out.println ("The patients in order from sickest" +
79                 " to healthiest are: " );
80
81             while (patients.peek () != null)
82             {
83                 System.out.println (patients.remove () + " ");
84             }
85         }
86     }
```