I have chosen the Queue data structure on an array to fit for my application for the hospital as per the outline states “doctors may get the patients to the operation theatre based on the order in which they have been inserted to the system”, and a queue data structure is a first in, first out data structure, so the outline’s specification is perfectly met.

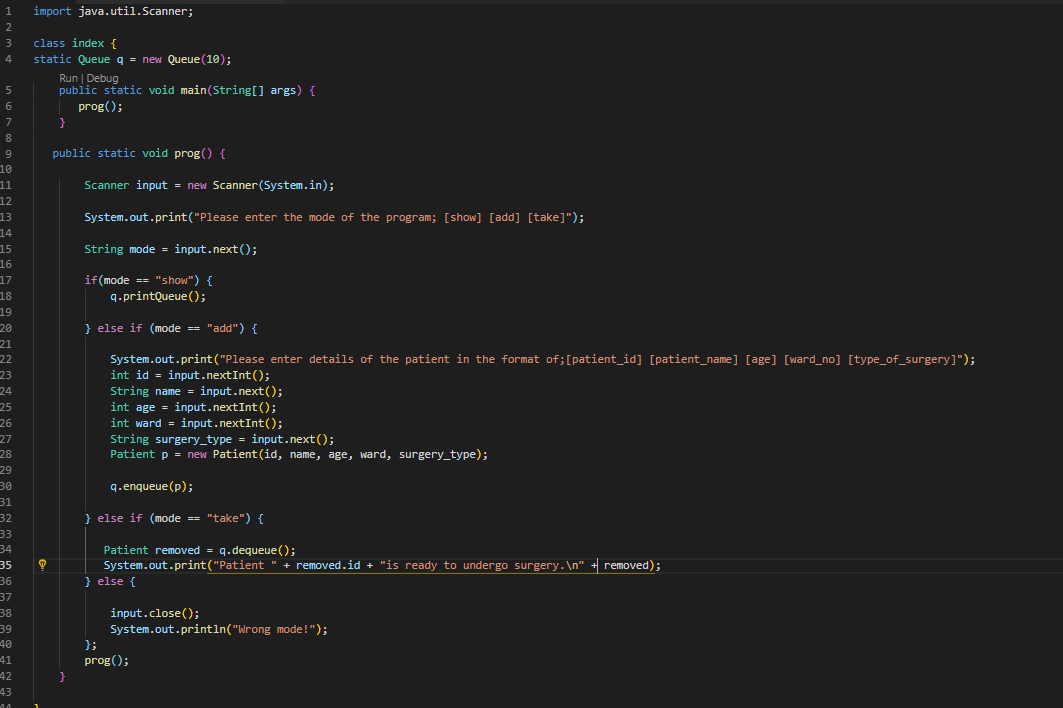
Persistent storage has not been incorporated to the application as the outline does not say that the application should be coupled with a database.

The queue data structure is simple to implement, and follows a decent time complexity. The queue is capable of only holding 10 elements, I believe 10 will be more than just sufficient for maintaining surgery information.

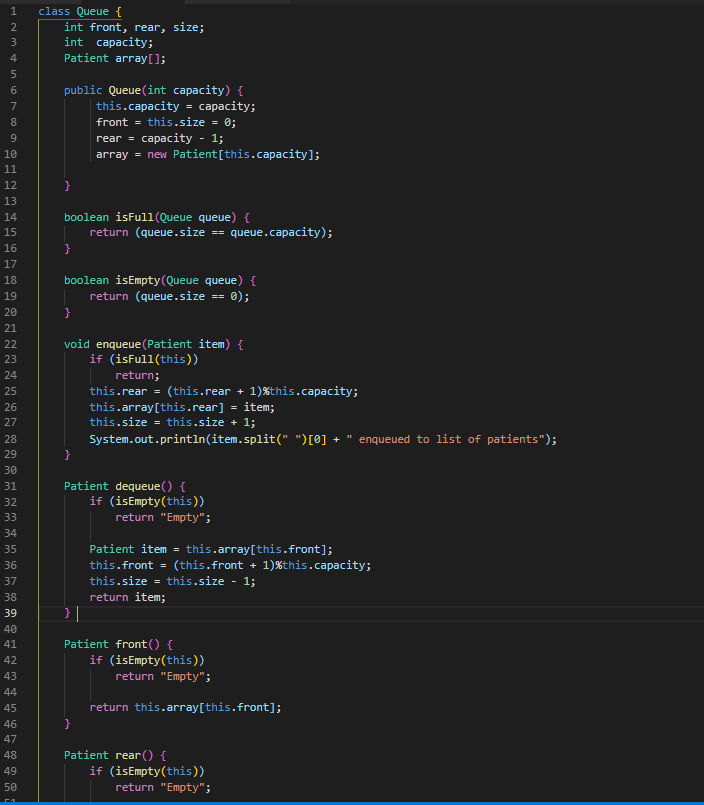
My proposed improvement to this application is to instead implement a circular queue on a circular linked list, this would be more suitable for an environment that has a very high frequency of surgeries. I’ve chosen not to improve my own application as the outline does not state the frequency of surgery.

The source code has not been included in this document as readability will be hindered. So I suggest opening index.java, queue.java and patient.java in a suitable IDE or a text editor. Regardless, below I will include a few screenshots of the code and how it looks when it’s running. Please check all the pages.

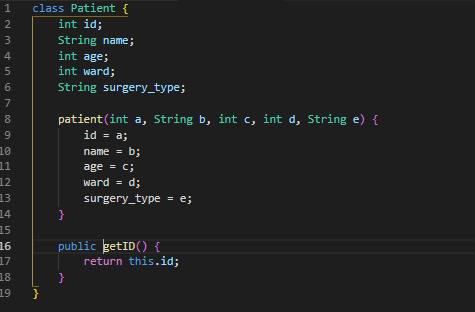
**Index.java**



**Queue.java**



**Patient.java**

****

**Runtime view**

