

Program: It's Your Queue

Create a generic `Queue` class that supports the following operations (and only these) as defined in our lectures:

- Enqueue: insert an item at the end of the queue
- Dequeue: delete an item from the beginning of the queue (and return it)
- Peek: return (but do not delete) the item at the beginning of the queue
- Size: return the size of the queue
- IsEmpty: return if the queue is empty
- IsFull: return if the queue is full
- Equals: compare two queues
- Add: concatenate two queues

Also include a constructor, a copy constructor, and the output capability – i.e.:

- `Queue()`
- `Queue(Queue q)`
- `String toString()`

Again, I am providing the `main` function and will test the functionality and implementation of your queue. You are to use it (unmodified) and build your generic queue on top of your generic list. As such, the only member variable your queue will *absolutely* require is a generic list. Implement your queue in a separate file (e.g., `Queue.java`) and make sure that `List.class` and `Node.class` are in the same directory.