

Define a queue class called `MyQueue`, in such a way that it can store any type of data. All data in a `MyQueue` must be stored in a `DoublyLinkedList` instance. Implement the following member methods:

- `enqueue`: Adds a new element into the queue.
- `dequeue`: Removes and returns the element in the front of queue.
- `IsEmpty`: Returns true if the queue is empty or not.
- `Size`: returns the number of element in the queue

Implement a `Stack` class with `push`, `pop` and `print` functions by using the queue class implemented in the previous step. You have to use the queue class as the data structure in the stack class.

Define a stack class called `MyStack`, in such a way that it can store any type of data. All data in a `MyStack` must be stored in a `MyQueue` instance. Implement the following member methods:

- `push`: Adds a new element into the stack.
- `dequeue`: deletes and returns the element in the end of stack.
- `IsEmpty`: Returns true if the stack is empty or not.
- `Size`: returns the number of element in the queue
- `Print`: prints the content of the stack