Use our LinkedList code and extend it to be a doubly linked list.

See <https://en.wikipedia.org/wiki/Doubly_linked_list>. As it describes in the article “***a doubly linked list is a linked data structure that consists of a set of sequentially linked records called nodes. Each node contains three fields: two link fields (references to the previous and to the next node in the sequence of nodes) and one data field.”*** We already have the data field and the reference to the next node. Add a reference to the previous.

As the article continues, “***the beginning and ending nodes' previous and next links, respectively, point to some kind of terminator”***. This in our case is the Head and the Tail.

The article continues, “***the two node links*** (next and previous) ***allow traversal of the list in either direction. While adding or removing a node in a doubly linked list requires changing more links than the same operations on a singly linked list, the operations are simpler and potentially more efficient”***. Fix the methods we already have, to account for the double linking. You can also implement some of the methods we left empty like FindLast(). AddLast(), AddBefore()