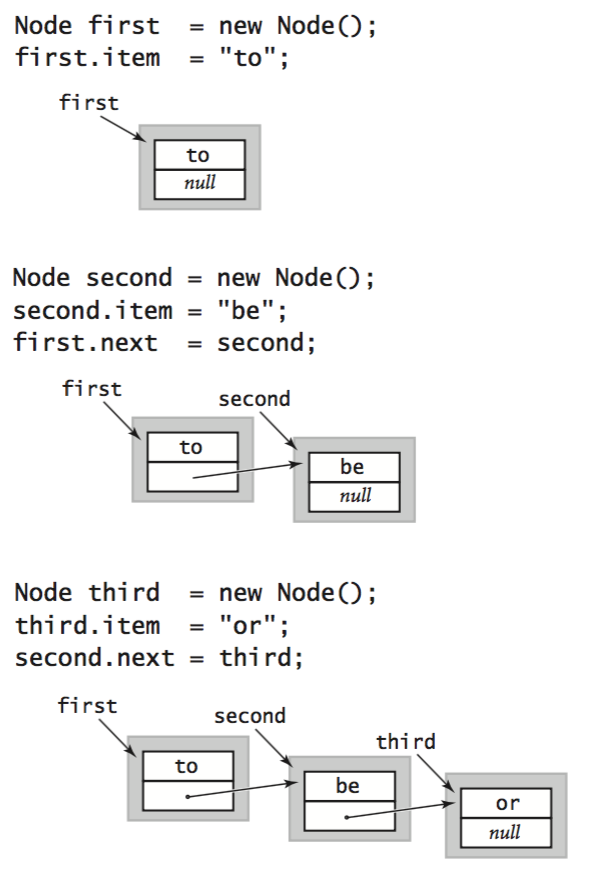
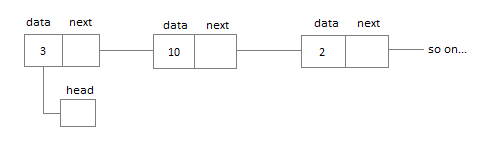
**Bullet Points:**

1. A linked list is a dynamic data structure consisting of nodes and links:
2. Linear data structure which consists of group of **nodes** in a sequence.
3. Each node holds its own **data** and the **address of the next node** hence forming a chain like structure.
4. Linked Lists are used to create trees and graphs.
5. LinkedList is faster in add and remove, but slower in get.
6. One drawback of linked list is that data access is sequential
7. LinkedList should be preferred if:
   1. there are no large number of random access of element
   2. there are a large number of add/remove operations
8. Each Node contains:
   1. Data
   2. A pointer to the next node
9. Create Node for each item.
   1. Node Object will have data and is waiting for reference of next node
10. Link the node
    1. When you create next node object, you will have object reference for that node.
    2. Assign this reference to the Node that was created before.

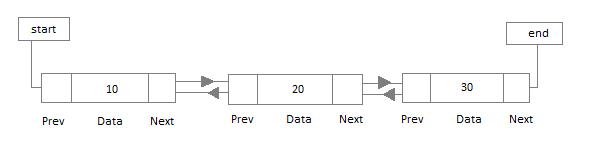
Example: Let’s say we are storing String data. “To”, “be” and “or”



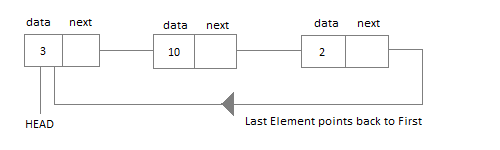
1. Singly Linked List:
   * 1. A *singly linked list* comprises a sequence of *nodes*, with each node containing a reference (or *link*) to its successor.
     2. Singly linked lists contain nodes which have a **data** part as well as an **address part** i.e. next, which points to the next node in the sequence of nodes.
     3. The operations we can perform on singly linked lists are **insertion**, **deletion** and **traversal**.



1. Doubly Linked List
   1. Since in Singly linked list going back to previous node is difficult.
   2. In a doubly linked list, each node contains a **data** part and two addresses, one for the **previous** node and one for the **next** node.



1. Circular Linked List:
   1. In circular linked list the last node of the list holds the address of the first node hence forming a circular chain.



Disadvantage of Linked List:

* The memory is wasted as pointers require extra memory for storage.
* No element can be accessed randomly; it must be accessed one node at a time, sequentially.
* Reverse Traversing is difficult in linked list.