# LinkedList

## Single LinkedList

Like arrays, Linked List is a linear data structure. Unlike arrays, linked list elements are not stored at a contiguous location; the elements are linked using pointers.

Diagram, box and whisker chart

Description automatically generated

A linked list is represented by a pointer to the first node of the linked list. The first node is called the head. If the linked list is empty, then the value of the head points to NULL.   
Each node in a list consists of at least two parts:   
1) data (we can store integer, strings or any type of data).  
2) Pointer (Or Reference) to the next node (connects one node to another)

In Java or C#, LinkedList can be represented as a class and a Node as a separate class. The LinkedList class contains a reference of Node class type.

The limitation with Arrays:

* The size of the arrays is fixed
* Inserting a new element in an array of elements is expensive because the room has to be created for the new elements and to create the room existing elements have to be shifted

Advantages to LinkedList over Arrays

* Dynamic size
* Ease of insertion/deletion

Drawback:

* Random access is not allowed
* Extra memory space for a pointer is required with each element of the list.

/\* Start with the empty list. \*/

* LinkedList llist = **new** LinkedList();
* llist.head = **new** Node(1);
* Node second = **new** Node(2);
* Node third = **new** Node(3);
* /\* Three nodes have been allocated dynamically.
* We have references to these three blocks as head,
* second and third
* llist.head        second              third
* |                |                  |
* |                |                  |
* +----+------+     +----+------+     +----+------+
* | 1  | null |     | 2  | null |     |  3 | null |
* +----+------+     +----+------+     +----+------+ \*/
* llist.head.next = second; // Link first node with the second node
* /\*  Now next of the first Node refers to the second.  So they
* both are linked.
* llist.head        second              third
* |                |                  |
* |                |                  |
* +----+------+     +----+------+     +----+------+
* | 1  |  o-------->| 2  | null |     |  3 | null |
* +----+------+     +----+------+     +----+------+ \*/
* second.next = third; // Link second node with the third node
* /\*  Now next of the second Node refers to third.  So all three
* nodes are linked.
* llist.head        second              third
* |                |                  |
* |                |                  |
* +----+------+     +----+------+     +----+------+
* | 1  |  o-------->| 2  |  o-------->|  3 | null |
* +----+------+     +----+------+     +----+------+ \*/
* }