Array and Linked List:

Array:

An array is a data structure that stores elements of the same data type in contiguous memory locations.
Arrays are index-based data structures where each data element is associated with an index.
An array is a big block of memory divided into smaller chunks, where each chunk stores value.

Linked list:

A Linked List is made up of a sequence of elements called Nodes.
Each node consists of two fields : one field stores data, and the second field (Pointer), stores an address that keeps a reference to the next node.
Pointer (or Reference) to the next node, connects one node to an address of another node

-> The main difference between an array and a linked list is in their data storage and manipulation methods. Arrays allocate a fixed-size,,
providing efficient indexing for direct access to elements. Linked lists, on the other hand, allocate memory dynamically at runtime, allowing for flexible size adjustments. While arrays excel in fast element access and are ideal for scenarios where size remains constant, linked lists shine in dynamic data scenarios, enabling efficient insertion and deletion operations. However, linked lists consume more memory due to node overhead, and their elements are interdependent, making them less suitable for scenarios where direct element access is crucial.

Array	Linked List
An array is a grouping of data elements of equivalent data type.	A linked list is a group of entities called a node. The node includes two segments: data and address.
It stores the data elements in a contiguous memory zone.	It stores elements randomly, or we can say anywhere in the memory zone.
In the case of an array, memory size is fixed, and it is not possible to change it during the run time.	In the linked list, the placement of elements is allocated during the run time.

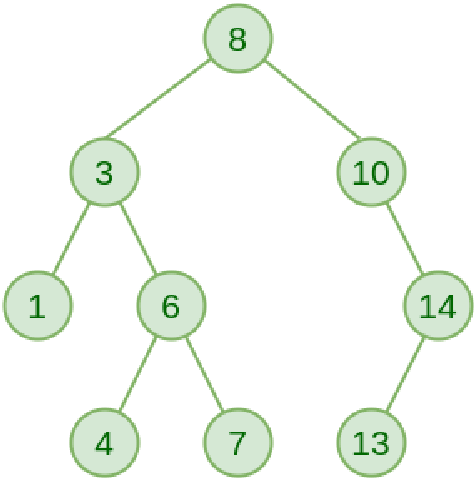
The elements are not dependent on each other.	The data elements are dependent on each other.
It is easier and faster to access the element in an array.	In a linked list, the process of accessing elements takes more time.
When it comes to executing any operation like insertion, deletion, array takes more time.	When it comes to executing any operation like insertion, deletion, the linked list takes less time.

Singly Linked List	Doubly Linked List
Node contains 2 field -data field and next link field.	Node contains 3 fields -data field, a previous link field and a next link field.
The traversal can be done using the next node link only. Thus traversal is possible in one direction only.	The traversal can be done using the previous node link or the next node link. Thus traversal is possible in both directions (forward and backward).
Less memory	More memory
Inserting and Deleting are harder	Inserting and Deleting are easier

Binary Search Tree:

Binary Search Tree is a node-based binary tree data structure which has the following properties:

The left subtree of a node contains only nodes with keys lesser than the node’s key.
The right subtree of a node contains only nodes with keys greater than the node’s key.
The left and right subtree each must also be a binary search tree.



Binary Search Tree