A singly linked list is a linear data structure where each node contains two parts:

 **Data:**The actual data stored in the node.

 **Next:** A pointer or reference to the next node in the sequence

A doubly linked list is similar to a singly linked list, but each node contains three parts:

 **Data:** The actual data stored in the node.

 **Next:** A pointer or reference to the next node in the sequence.

 **Previous:** A pointer or reference to the previous node in the sequence.

* **Dynamic Size:** Linked lists can grow or shrink dynamically as needed, while arrays have a fixed size determined at creation.

This makes linked lists more flexible for handling varying data sizes.

* **Efficient Insertion and Deletion:** Inserting or deleting elements in a linked list is generally more efficient than in an array, especially at arbitrary positions. Arrays often require shifting elements to accommodate changes, which can be time-consuming.
* **Memory Utilization:** Linked lists only allocate memory for the nodes that are currently in use, while arrays allocate a fixed amount of contiguous memory upfront, even if some elements are not used. This can be more memory-efficient for linked lists.
* **Flexibility:** Linked lists can be used to implement various data structures like stacks, queues, and graphs, offering more flexibility than arrays.