**LinkedList in C#**

A LinkedList in C# is a collection of nodes where each node contains data and a reference to the next (and optionally previous) node. It is part of the System.Collections.Generic namespace and is ideal for scenarios where frequent insertions and deletions are required.

**Key Features:**

1. **Dynamic Size**: Grows and shrinks dynamically.
2. **Doubly Linked**: Nodes have references to both the next and previous nodes.
3. **Efficient Insertions/Deletions**: No need for shifting elements like arrays.
4. **Traversal**: Can be traversed from the first to the last node (and vice versa).

**Common Use Cases:**

1. **Queue and Stack Implementation**: Useful when insertion and removal operations are frequent.
2. **Undo Functionality**: Maintaining a history of operations.
3. **Graphs**: Representing adjacency lists in graph algorithms.
4. **Music/Video Playlists**: Maintaining the order of play with quick navigation.

**Commonly Used Methods:**

* **AddFirst(T value)**: Adds a new node at the beginning.
* **AddLast(T value)**: Adds a new node at the end.
* **AddBefore(LinkedListNode<T> node, T value)**: Adds a new node before the specified node.
* **AddAfter(LinkedListNode<T> node, T value)**: Adds a new node after the specified node.
* **Remove(T value)**: Removes the first occurrence of the specified value.
* **RemoveFirst() / RemoveLast()**: Removes the first or last node.
* **Find(T value)**: Finds the first node containing the specified value.
* **Clear()**: Removes all nodes.