C# provides several **built-in data structure classes** in the **System.Collections**, **System.Collections.Generic**, and **System.Collections.Concurrent** namespaces.

Here’s a categorized list of **common built-in data structures in C#** with examples:

**🧱 1. Arrays (Fixed Size)**

**🗂 2. Lists & Collections**

| **Data Structure** | **Namespace** | **Description** |
| --- | --- | --- |
| List<T> | System.Collections.Generic | Resizable array (most commonly used) |
| LinkedList<T> | System.Collections.Generic | Doubly linked list |
| HashSet<T> | System.Collections.Generic | Set of unique elements |
| SortedSet<T> | System.Collections.Generic | Automatically sorted unique elements |
| Collection<T> | System.Collections.ObjectModel | Base class for custom collections |

**3. Dictionaries / Maps**

| **Type** | **Description** |
| --- | --- |
| Dictionary<TKey,TValue> | Fast key-value store (hash table) |
| SortedDictionary<TKey,TValue> | Sorted by keys |
| SortedList<TKey,TValue> | Sorted keys, optimized for lookups |
| ConcurrentDictionary<TKey,TValue> | Thread-safe dictionary |

**4. Stacks and Queues**

| **Data Structure** | **Description** |
| --- | --- |
| Stack<T> | LIFO (last in, first out) |
| Queue<T> | FIFO (first in, first out) |
| ConcurrentQueue<T> | Thread-safe queue |
| ConcurrentStack<T> | Thread-safe stack |
| PriorityQueue<TElement, TPriority> | Priority-based queue (C# 10/.NET 6+) |

**Summary Table:**

| **Category** | **Types** |
| --- | --- |
| Array | Array, ArrayList |
| Linear | List<T>, LinkedList<T>, Queue<T>, Stack<T> |
| Set | HashSet<T>, SortedSet<T> |
| Map | Dictionary<TKey, TValue>, SortedDictionary<TKey, TValue> |
| Priority | PriorityQueue<TElement, TPriority> |
| Concurrent | ConcurrentQueue<T>, ConcurrentStack<T>, ConcurrentDictionary<T> |
| Immutable | ImmutableList<T>, ImmutableDictionary<TKey, TValue>, etc. |
| Others | BitArray, ObservableCollection<T>, BlockingCollection<T> |

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

C# does **not** include built-in classes like Node, BinaryTree, or Graph directly in the base class libraries.

But does not give ready-made classes like TreeNode or BinaryTree

**⚙️ Common Custom Structures You Might Write in C#**

| **Data Structure** | **How to Do It** |
| --- | --- |
| Node | Simple class with Value and Next or Left/Right |
| LinkedList | Manual or use LinkedList<T> |
| Binary Tree | Custom class with Left and Right nodes |
| Graph | Dictionary of adjacency lists |
| Trie | Nested classes with child node dictionaries |
| Heap | Manual array-based or PriorityQueue<T> |