C#

Dictionary

- Dictionary collection contains a group of elements of key/value pairs.
- Full Path: System.Collections.Generic.Dictionary
- The "Dictionary" class is a generic class; so you need to specify data type of the key and data type of the value while creating object.
- You can set / get the value based on the key.
- The key can't be null or duplicate.

Dictionary Collection

[key 0]	value0
[key 1]	value1
[key 2]	value2
[key 3]	value3
[key 4]	value4
[key 5]	value5
[key 6]	value6

'Dictionary' collection

```
Dictionary<TKey, TValue>  referenceVariable = new  Dictionary<TKey, TValue>( );
```

- It is dynamically sized. You can add, remove elements (key/value pairs) at any time.
- Key can't be null or duplicate; but value can be null or duplicate.
- It is not index-based. You need to access elements by using key.
- It is not sorted by default. The elements are stored in the same order, how they are initialized.

Properties

- **Count** : Returns count of elements.
- **[TKey]** : Returns value based on specified key.
- **Keys** : Returns a collection of key (without values).
- **Values** : Returns a collection of values (without keys).

Methods

- `void` **Add(TKey, TValue)** : Adds an element (key/value pair).
- `bool` **Remove(TKey)** : Removes an element based on specified key.
- `bool` **ContainsKey(TKey)** : Determines whether the specified key exists.
- `bool` **ContainsValue(TValue)** : Determines whether the specified value exists.
- `void` **Clear()** : Removes all elements.