F# - MUTABLE DICTIONARY

The **Dictionary<'TKey, 'TValue>** class is the mutable analog of the F# map data structure and contains many of the same functions.

Recapitulating from the Map chapter in F#, a map is a special kind of set that associates the values with key.

Creating of a Mutable Dictionary

Mutable dictionaries are created using the **new** keyword and calling the list's constructor. The following example demonstrates this −

```
open System.Collections.Generic
let dict = new Dictionary<string, string>()
dict.Add("1501", "Zara Ali")
dict.Add("1502","Rishita Gupta")
dict.Add("1503","Robin Sahoo")
dict.Add("1504","Gillian Megan")
printfn "Dictionary - students: %A" dict
```

When you compile and execute the program, it yields the following output −

```
Dictionary - students: seq
[[1501, Zara Ali]; [1502, Rishita Gupta]; [1503, Robin Sahoo];
[1504, Gillian Megan]]
```

The Dictionary*TKey, TValue* Class

The Dictionary*TKey, TValue* Class represents a collection of keys and values.

The following tables provide the properties, constructors and the methods of the List*T* class −

Properties

Property	Description
Comparer	Gets the IEqualityComparer*T* that is used to determine equality of keys for the dictionary.
Count	Gets the number of key/value pairs contained in the Dictionary*TKey, TValue*.
Item	Gets or sets the value associated with the specified key.
Keys	Gets a collection containing the keys in the Dictionary*TKey, TValue*.
Values	Gets a collection containing the values in the Dictionary*TKey, TValue*.

Constructors

Constructors	Description
Dictionary*TKey, TValue*	Initializes a new instance of the **Dictionary*TKey, TValue*** class that is empty, has the default initial capacity, and uses the default equality comparer for the key type.
Dictionary*TKey, TValue* *IDictionary*(*TKey, TValue*)	Initializes a new instance of the **Dictionary*TKey, TValue*** class that contains elements copied from the specified **IDictionary**

	TKey, TValue and uses the default equality comparer for the key type.
Dictionary*TKey*, *TValue* *IEqualityComparer*(*TKey*)	Initializes a new instance of the **Dictionary*TKey, TValue*** class that is empty, has the default initial capacity, and uses the specified **IEqualityComparer*T*.**
Dictionary*TKey*, *TValue*Int32	Initializes a new instance of the **Dictionary*TKey, TValue*** class that is empty, has the specified initial capacity, and uses the default equality comparer for the key type.
Dictionary*TKey*, *TValue* *IDictionary*(*TKey*, *TValue*, IEqualityComparer*TKey*)	Initializes a new instance of the **Dictionary*TKey, TValue*** class that contains elements copied from the specified **IDictionary *TKey, TValue*** and uses the specified **IEqualityComparer*T*.**
Dictionary*TKey*, *TValue* *Int*32, *IEqualityComparer*(*TKey*)	Initializes a new instance of the **Dictionary*TKey, TValue*** class that is empty, has the specified initial capacity, and uses the specified **IEqualityComparer*T*.**
Dictionary*TKey*, *TValue* *SerializationInfo*, *StreamingContext*	Initializes a new instance of the **ictionary*TKey, TValue*** class with serialized data.

Methods

Method	Description
Add	Adds the specified key and value to the dictionary.
Clear	Removes all keys and values from the Dictionary*TKey*, *TValue*.
ContainsKey	Determines whether the Dictionary*TKey*, *TValue* contains the specified key.
ContainsValue	Determines whether the Dictionary*TKey*, *TValue* contains a specific value.
Equals*Object*	Determines whether the specified object is equal to the current object. *InheritedfromObject*.
Finalize	Allows an object to try to free resources and perform other cleanup operations before it is reclaimed by garbage collection. *InheritedfromObject*.
GetEnumerator	Returns an enumerator that iterates through the Dictionary*TKey*, *TValue*.
GetHashCode	Serves as the default hash function. *InheritedfromObject*.
GetObjectData	Implements the System.Runtime.Serialization.ISerializable interface and returns the data needed to serialize the Dictionary*TKey*, *TValue*instance.
GetType	Gets the Type of the current instance. *InheritedfromObject*.
MemberwiseClone	Creates a shallow copy of the current Object. *InheritedfromObject*.
OnDeserialization	Implements the System.Runtime.Serialization.ISerializable interface and raises the deserialization event when the deserialization is complete.
Remove	Removes the value with the specified key from the Dictionary*TKey*, *TValue*.
ToString	Returns a string that represents the current object. *InheritedfromObject*.
TryGetValue	Gets the value associated with the specified key.

Example

```
open System.Collections.Generic
```

```
let dict = new Dictionary<string, string>()

dict.Add("1501", "Zara Ali")
dict.Add("1502","Rishita Gupta")
dict.Add("1503","Robin Sahoo")
dict.Add("1504","Gillian Megan")

printfn "Dictionary - students: %A" dict
printfn "Total Number of Students: %d" dict.Count
printfn "The keys: %A" dict.Keys
printf"The Values: %A" dict.Values
```

When you compile and execute the program, it yields the following output –

```
Dictionary - students: seq
[[1501, Zara Ali]; [1502, Rishita Gupta]; [1503, Robin Sahoo];
[1504, Gillian Megan]]
Total Number of Students: 4
The keys: seq ["1501"; "1502"; "1503"; "1504"]
The Values: seq ["Zara Ali"; "Rishita Gupta"; "Robin Sahoo"; "Gillian Megan"]
```