

Forum:[Hack Forums](#)

Thread: [\[C#\]\[TUT\]How to use Dictionaries!](#)

Ethereal - January 1, 2011, 9:40 am

A dictionary is basically a database that you can easily access within your code. It has two variables - The key, and the value. You can access the value via a lookup utilizing the key. The following code is the syntax for declaring a new dictionary:

```
Code Dictionary<varType1,varType2> DicName = new Dictionary<varType1,varType2>();  
For example:
```

```
Code private Dictionary<String, String> _keySettings = new Dictionary<String,String>();  
NOTE* It uses the class, rather than the type.
```

We can add values to our dictionary very easily:

```
Code DicName.Add(var1, var2)  
For example:
```

```
Code _keySettings.Add(kname, kval);  
This will be our first entry in our Dictionary. We can check how many values we have in our  
Dictionary with the code:
```

```
Code DicName.Count();  
When we wish to access the data contained inside our Dictionary, we can call a lookup with the key  
as a parameter. This will return the value that the key corresponds to:
```

```
Code DicName[key];  
And an example for outputting the values stored in the Dictionary into a listbox is:
```

```
Code foreach(string s in GetDictionaryKeys()) {  
listBox1.Items.Add(s); }
```

```
public string[] GetDictionaryKeys() { return _keySettings.Values.ToArray(); }
```

Now we are past the boring bit. You may ask where you would use this over an array, or other variable type? I personally first used this to hold the variable data for a program that allows you to manipulate a CS:S config.

I read text from a file, parsed it and then added it into a Dictionary. The Dictionary data can then be passed to another form, for you to access as you please.

Then, in a second form, I allowed new values to be entered changing the keybinding, and re-wrote the dictionary. Lookups were then used to re-write the config, with the new configuration settings.

I have also used this as a lookup to convert characters/numeric values for my private encryption methods.

This is just one example of implementing a Dictionary. Your use of it is limited only by your imagination.

