

C# 6 New Features

Jon Limjap

Microsoft MVP for C#



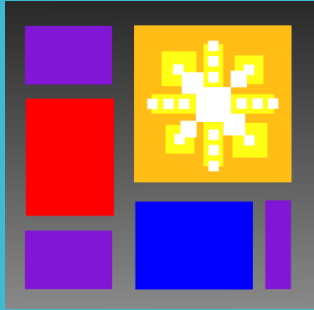
Visual Studio



About Me



- Microsoft MVP for C#
- Technical Specialist, Provoke Solutions
- President, Philippine .NET Users Group
- <http://dotnet.kapenilattex.com>
- @lattex | jonlimjap@gmail.com



Philippine .NET Users Group (PHINUG)

- facebook.com/groups/phinug
- @phinug
- phinug.doorkeeper.jp

Agenda

- A history of the continuous evolution of the C# language
- C# 6.0 New features

History

Continuous evolution of the C# language



C# 2.0

November 2005

- Generics
- Partial Classes
- Anonymous Methods
- Iterators
- Nullable Types
- Static Classes
- More...

C# 3.0

November 2007

- Implicitly typed variables aka var keyword
- Object and collection initializers
- Auto-properties
- Anonymous Types
- Extension Methods
- Language Integrated Query (LINQ)
- Lambda Expressions
- More...

C# 4.0
April 2010

- Dynamic Binding
- Named and Optional Parameters
- Generic Covariance and Contravariance
- More...

C# 5.0

August 2012

- Asynchronous methods (async and await keywords)
- Caller info attributes

C# 6.0

Not released yet ;)

New Features

- Using statements for static members
- Auto-property Initializers
- Getter-only automatic properties
- Dictionary Initializers
- Expression-bodied members
- Await inside a finally block
- Exception Filters
- Null Propagation
- And a few more...

Using statement for Static Members

```
using System;
using System.Console;

namespace Csharp6Demo
{
    class Program
    {
        static void Main(string[] args)
        {
            WriteLine("WriteLine stands on its own :)");
        }
    }
}
```

Auto-property initializers

```
string FirstName { get; set; } = "Juan";  
string LastName { get; set; } = "Dela Cruz";  
DateTime BirthDate { get; set; } = new DateTime(1898, 6, 12);
```

Getter only Auto- properties

```
string OtherInfo { get; } = "Other information";
```

Dictionary Initializers

```
var madlangPeople = new Dictionary<string, Person>
{
    ["Jon"] = new Person("Jon", "Limjap", new DateTime(1990, 11, 23)),
    ["Toto"] = new Person("Toto", "Gamboa", new DateTime(1996, 7, 11)),
    ["Teril"] = new Person("Teril", "Bilog", new DateTime(1998, 7, 10))
};
```

Expression- bodied members

```
public int Age =>  
    Convert.ToInt32(DateTime.Now.Subtract(BirthDate).Days/365.25);
```


Await inside Catch and Finally blocks

```
public async Task SomeAsyncMethod()
{
    try
    {
        await OperationThatMayThrowAsync();
    }
    catch (Exception ex)
    {
        await LogAsync(ex);
    }
    finally
    {
        await Cleanup();
    }
}
```

Exception Filters

```
try
{
    throw new Exception(message);
}
catch (Exception ex) if (ex.Message == "Filter me!")
{
    WriteLine("Filtered!");
}
catch
{
    WriteLine("Not filtered");
}
```

Null Propagation

```
public class Address
{
    public string AddressLine1 { get; set; }
    public string AddressLine2 { get; set; }
}
```

```
public class Person
{
    //other code
    public Address Address { get; set; }
}
```

```
var jon1 = new Person();
var jon2 = new Person() { Address = new Address {
    AddressLine1 = "6750 Ayala Ave" } };
```

```
WriteLine(jon1?.Address?.AddressLine1 ?? "No address");
WriteLine(jon2?.Address?.AddressLine1 ?? "No address too");
WriteLine(jon2?.Address?.AddressLine2 ?? "No address
either");
```

Thank you!

Q & A

References

- <https://roslyn.codeplex.com/wikipage?title=Language%20feature%20status&referringTitle=Home>
- http://www.youtube.com/watch?v=BA3sL783_Co