

#develop 1.0



This work is licensed under the Creative Commons Attribution License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/2.0/at/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Introduction

The #develop project started on September 11th, 2000. The project was initiated by Mike Krüger, who is the main developer and architect of #develop to date. In the course of the project, Mike was joined by several contributors, who have helped a great deal to make the 1.0 release a successful one – though it took us four years and a few architectural changes along the way.

With version 1.0, we have built a solid foundation for the growth of this Integrated Development Environment. Though the features found inside the IDE should not be considered “basic”, we have plans for the future and we are sure that you will like those too.

The #develop team

<http://www.icsharpcode.net/pub/relations/team.aspx>

<http://wiki.sharpdevelop.net/default.aspx/SharpDevelop.Contributors>

Overview

#develop (short for SharpDevelop) is a free Integrated Development Environment (IDE) for C#, VB.NET, Managed C++ and ILAsm projects on Microsoft's .NET platform. It is written entirely in C#, and comes with features you would expect in an IDE plus a few more – the next section details those.

Features

Code Completion

No IDE would be complete without this feature, which is supported for both C# and VB.NET. Write a statement followed by the . (dot) and code completion will kick in. Too lazy to type that long variable or class name? Press Ctrl+Space and #develop will provide you with potential candidates for what you began to type.

Forms Designer

A Windows Forms Designer ships inside #develop for both C# and VB.NET. It comes with full design time support, roundtripping and generally saves you a lot of time when developing Windows-based client software. Note: there is no designer for ASP.NET Web forms.

Code AutoInsert

Tired of implementing interface definitions, overrideables, writing get and set accessors for various member variables? Alt+Ins is your ticket to time saving. Just select the action you want to perform and the tedious typing work is taken care of for you.

Code Converter

You are a VB.NET developer. You find a sample that exactly solves your problem. It is C#. Sound familiar? No problem with #develop – it comes with conversion from C# to VB.NET and vice versa. And it does its magic on the project level too, so converting entire projects is a piece of cake. Combined with the next feature, this really becomes a time saver.

VS.NET Solution Import / Export

You can import Visual Studio .NET solutions (.sln as well as the project files only, which is especially helpful for ASP.NET projects) into #develop. Export is also supported, but please

note that because #develop is capable of deeper project nesting than VS.NET, this hierarchy will be flattened if necessary.

Folding

Considered a standard feature in modern development editors, folding code regions is supported by #develop based on the parsing information of your code files. Folding state is remembered for you across development sessions.

Integrated NUnit Unit Testing

View / Tools / Unit Tests is an integrated NUnit front end for executing unit tests in your assembly. No need to leave the development environment, do test-first-design inside your IDE!

Assembly Analyzer

When you have built your project, go to View / Assembly Analyzer. It will point out errors and make suggestions to improve your code's compliance to .NET coding standards. This is comparable to the separately available FxCop tool from Microsoft, however integrated into your development experience.

XML Documentation Preview

You have written an XML documentation comment – want to know how it will look like in the help file? Simply place the cursor in the comment and press Ctrl+Q. And when you are satisfied, use Project / Generate Documentation to let NDoc create the help file for you.

And more...

To pick a few of the other features that are supported by #develop, starting with the obvious: syntax highlighting, intelligent braces, bookmarks, code templates, regular expressions toolkit, New Class wizard, HTML export, Class browser, NDoc integration, NProf integration.

Documentation

The Help system inside #develop has not been updated for version 1.0. Though most of the information still should be applicable, it is neither complete nor guaranteed to work as advertised. Plans are to move the entire help system to a Web based content management system to provide you with the most up to date information and give our help authors an easy way to build the help experience.

Support

Our primary means of support is via our Web-based forum (please do not email team members directly unless they advise you to do so in the forum):

<http://www.icssharpcode.net/opensource/sd/forum/>

Before posting, we would like to encourage you to visit the following Wiki pages:

<http://wiki.sharpdevelop.net/default.aspx/SharpDevelop.FrequentlyAskedQuestions1x>
<http://wiki.sharpdevelop.net/default.aspx/SharpDevelop.BugList1x>

When reporting bugs, please use the Bug Reporting forum and be sure to revisit the sticky topics on how to make good bug reports. Please provide us with steps to reproduce the error.

License

#develop's source code is GPL licensed (please see license.txt for the GPL in its entire length). Should you have questions about the GPL, please visit the following URLs:

Frequently Asked Questions about the GNU GPL

<http://www.gnu.org/licenses/gpl-faq.html>

IC#Code FAQ: Open Source Licenses (GPL, LGPL)

http://www.icsharpcode.net/opensource/sd/forum/forum.asp?FORUM_ID=12