



How-Tos

Version 5.0.5-1

This is a collection of HOW-TOs designed to make it easier to get started using Wing on certain platforms with special requirements and with specific tools and libraries for GUI, web, and other types of development.

Wingware, the feather logo, Wing IDE, Wing IDE 101, Wing IDE Personal, Wing IDE Professional, and "The Intelligent Development Environment" are trademarks or registered trademarks of Wingware in the United States and other countries.

Disclaimers: The information contained in this document is subject to change without notice. Wingware shall not be liable for technical or editorial errors or omissions contained in this document; nor for incidental or consequential damages resulting from furnishing, performance, or use of this material.

Hardware and software products mentioned herein are named for identification purposes only and may be trademarks of their respective owners.

Copyright (c) 1999-2014 by Wingware. All rights reserved.

Wingware
P.O. Box 400527
Cambridge, MA 02140-0006
United States of America

Contents

How-Tos	1
Wing IDE Quick Start Guide	1
Install Python and Wing IDE	1
Configure Python	1
Key Features	1
Related Documents	2

Wing IDE Quick Start Guide

This is a minimalist guide for those wanting to get started with Wing IDE as quickly as possible. For a more in-depth introduction, try the [Tutorial](#).

Install Python and Wing IDE

Both Python and Wing IDE must be installed. The Wing IDE executable is called **wing-101-5.0**. See [Installing](#), [Running the IDE](#), and [Installing your License](#) for details.

Configure Python

Wing finds the latest installed Python and uses that with your code. If you want to change which Python is used or need to define some environment or set the **PYTHONPATH**, use the **Configure Python** item in the **Edit** menu.

Key Features

You are now ready to start working with code. Most of Wing's features are readily evident from the user interface. The **Wing Tips** tool provides brief usage hints.

These are the features you should be sure to try while evaluating Wing IDE:

- *Configurable Key Bindings* -- Wing can emulate VI/Vim, Visual Studio, Emacs, Eclipse, and Brief key bindings, selected with the editor **Personality** preference.
- *Configurable Tab Key* -- The default tab key action depends on file type, context, and whether or not there is a selection. This can be changed from the **Tab Key Action** preference.
- *Goto-definition* -- Available from the toolbar, **Source** menu, and by right-clicking on symbols in the editor. Use the browser-like forward/back history buttons at the top left of the editor to return from visiting a point of definition.
- *Source Navigation* -- The index menus at the top of the editor provide quick access to other parts of a source file from the menus at the top of the source editor.
- *Search Tool* -- Provides wild card, and regular expression search and replace.
- *Python Shell* -- Wing's **Python Shell** lets you try out code in a sandbox process kept isolated from Wing IDE and your debug process.
- *Basic Debugging* -- Set a breakpoint and start debugging. You can use the **Stack Data** tools to inspect or change program data. Wing distinguishes between fatal and non-fatal exceptions at the time they are raised, allowing you to inspect live program state more often. Debug process I/O is shown in the **Debug I/O** tool (or optionally in an external console).

Wing IDE Quick Start Guide

- *Customizable User Interface* -- Many options are available from **Preferences**, and you can split tools panels and move around the tools within them. Right click on the tabs for options, or drag tool or editor tabs to move them around or create new splits. Right click on the toolbar to configure which tools are visible or to add your own.

Related Documents

For more information see:

- [Wing IDE Tutorial](#), a detailed guided tour for Wing IDE.
- [Wing IDE Reference Manual](#), which describes Wing IDE in detail.