

# Microsoft Visual Basic Default Keybindings

### Introduction

This document contains the keyboard shortcuts that are set by default when you use the Visual Basic Development Settings. To set the Visual Basic Development Settings, on the **Tools** menu, click **Options**. In the Options dialog box, select **Show all settings**, and then, under **Environment**, click **Keyboard**. Click the Visual Basic 6 keyboard mapping scheme in the drop-down list and then click **OK**.

You can customize individual command shortcuts yourself. To do so, on the **Tools** menu, click **Options**. In the Options dialog box, select **Show all settings**, and then, under **Environment**, click **Keyboard**. To edit a shortcut, locate the command name in the dialog box and then assign it a shortcut key. In this document, names for the commands described are provided in parentheses.

For a complete list of commands, along with their keyboard shortcuts and detailed descriptions, see <http://go.microsoft.com/fwlink/?LinkId=74687>.

### Editing

Toggle All Outlining	<b>CTRL + M, CTRL + L</b>	Toggles all existing regions between collapsed and expanded states. <i>(Edit.ToggleAllOutlining)</i>
Toggle Outlining for Current Region	<b>CTRL + M, CTRL + M</b>	Toggles the current region between collapsed and expanded states. <i>(Edit.ToggleOutliningExpansion)</i>
Comment and Uncomment	<b>CTRL + K, CTRL + C</b> and <b>CTRL + K, CTRL + U</b>	Inserts and removes, respectively, the apostrophe (') at the beginning of the current line or every selected line. <i>(Edit.CommentSelection and Edit.UncommentSelection)</i>
Undo	<b>CTRL + Z</b>	Undoes the last action. <i>(Edit.Undo)</i>
Redo	<b>CTRL + SHIFT + Z</b>	Redoes the last action. <i>(Edit.Redo)</i>
Cut Line	<b>CTRL + Y</b>	Cuts the current line of code. <i>(Edit.LineCut)</i>
Insert Blank Line	<b>CTRL + ENTER</b> and <b>CTRL + SHIFT + ENTER</b>	Inserts a blank line above and below the cursor position, respectively. <i>(Edit.LineOpenAbove and Edit.LineOpenBelow)</i>
Select Word	<b>CTRL + SHIFT + W</b>	Selects the word containing, or to the right of, the insertion point. <i>(Edit.SelectCurrentWord)</i>
Delete Word	<b>CTRL + BACKSPACE</b> and <b>CTRL + DELETE</b>	Deletes to the beginning and end of the word, respectively. <i>(Edit.WordDeleteToStart and Edit.WordDeleteToEnd)</i>
Change Casing	<b>CTRL + U</b> and <b>CTRL + SHIFT + U</b>	Changes the selected text to lowercase and uppercase characters, respectively. <i>(Edit.MakeLowercase and Edit.MakeUppercase)</i>
Replace	<b>CTRL + H</b> and <b>CTRL + SHIFT + H</b>	Displays the <b>Quick Replace</b> tab and the <b>Replace In Files</b> tab, respectively, of the <b>Find and Replace</b> dialog box. <i>(Edit.Replace and Edit.ReplaceInFiles)</i>
Extend Selection	<b>SHIFT + ALT + UP ARROW</b> and <b>SHIFT + ALT + DOWN ARROW</b>	Moves the cursor one line up or down respectively, extending the line selection. <i>(Edit.LineUpExtendColumn and Edit.LineDownExtendColumn)</i>
Format Code	<b>CTRL + K, CTRL + D</b> and <b>CTRL + K, CTRL + F</b>	Formats the current document or selection, respectively. <i>(Edit.FormatDocument and Edit.FormatSelection)</i>
Display Smart Tag	<b>CTRL + DOT (.)</b> or <b>SHIFT + ALT + F10</b>	Displays the available options on the Smart Tag menu. <i>(View.ShowSmartTag)</i>
Help	<b>F1</b>	Displays a topic from Help that corresponds to the current user interface element or to the code item or error messages selected. <i>(Help.F1Help)</i>

### Build

Build Solution	<b>CTRL + SHIFT + B</b>	Builds all the projects in the solution. <i>(Build.BuildSolution)</i>
----------------	-------------------------	---

### Windows

Navigation Bar	<b>CTRL + F2</b>	Moves the cursor to the drop-down bar located at the top of the Code Editor. <i>(Window.MoveToNavigationBar)</i>
Object Browser	<b>F2</b>	Displays the <b>Object Browser</b> . <i>(View.ObjectBrowser)</i>
Properties Window	<b>F4</b>	Displays the <b>Properties</b> window for the currently selected item. <i>(View.PropertiesWindow)</i>
Solution Explorer	<b>CTRL + R</b>	Displays <b>Solution Explorer</b> . <i>(View.SolutionExplorer)</i>
View Server Explorer	<b>CTRL + ALT + S</b>	Displays <b>Server Explorer</b> . <i>(View.ServerExplorer)</i>
Show Data Sources	<b>SHIFT + ALT + D</b>	Displays the <b>Data Sources</b> window. <i>(Data.ShowDataSources)</i>
Toolbox	<b>CTRL + ALT + X</b>	Displays the <b>Toolbox</b> . <i>(View.Toolbox)</i>
Error List	<b>CTRL + \, E</b>	Displays the <b>Error List</b> . <i>(View.ErrorList)</i>
Close Tool Window	<b>SHIFT + ESC</b>	Closes the current tool window. <i>(Window.CloseToolWindow)</i>
Close Document Window	<b>CTRL + F4</b>	Closes the current tab. <i>(Window.CloseDocumentWindow)</i>

### Debugging

Start Debugging	<b>F5</b>	Starts the application in the debugger. When in Break mode, invoking this command runs the application until the next breakpoint. <i>(Debug.Start)</i>
Start Without Debugging	<b>CTRL + F5</b>	Starts the application without invoking the debugger. Use this exclusively rather than F5 for Web site debugging. <i>(Debug.StartWithoutDebugging)</i>
Step Into	<b>F8</b> or <b>F11</b>	Executes code one statement at a time, following execution into method calls. <i>(Debug.StepInto)</i>
Step Out	<b>CTRL + SHIFT + F8</b> or <b>SHIFT + F11</b>	Executes the remaining lines of the method in which the current execution point is located. <i>(Debug.StepOut)</i>
Step Over	<b>SHIFT + F8</b> or <b>F10</b>	Executes the next line of code, but does not follow execution into any method calls. <i>(Debug.StepOver)</i>
Stop Debugging	<b>CTRL + ALT + BREAK</b>	Stops running the current application in the debugger. <i>(Debug.StopDebugging)</i>
Toggle Breakpoint	<b>F9</b>	Sets or removes a breakpoint at the current line. <i>(Debug.ToggleBreakpoint)</i>
Set Next Statement	<b>CTRL + F9</b>	Sets the execution point to the line of code you choose. <i>(Debug.SetNextStatement)</i>
Break at a Function	<b>CTRL + B</b>	Displays the <b>New Breakpoint</b> window. <i>(Debug.BreakatFunction)</i>
Attach to Process	<b>CTRL + ALT + P</b>	Displays the <b>Attach to Process</b> dialog box. <i>(Tools.AttachToProcess)</i>
Make Data Tip Transparent	<b>CTRL</b>	Hides the current data tip so that you can see the code beneath. Must be invoked while a data tip is active.
Immediate Window	<b>CTRL + G</b>	Displays the <b>Immediate</b> window. <i>(Debug.Immediate)</i>
Call Stack Window	<b>CTRL + L</b>	Displays the <b>Call Stack</b> window. <i>(Debug.CallStack)</i>
QuickWatch Window	<b>SHIFT + F9</b>	Displays the <b>QuickWatch</b> dialog box. <i>(Debug.QuickWatch)</i>

### Refactoring

Refactor!	<b>CTRL + TILDE (~)</b>	Refactor! for Visual Basic 2005 is a free plug-in from Developer Express Inc., in partnership with Microsoft. Refactor! supports more than 15 individual refactoring features including Extract Method, Encapsulate Field, Reorder Parameters, Create Overload, and Surrounds With. See <a href="http://go.microsoft.com/fwlink/?LinkId=74990">http://go.microsoft.com/fwlink/?LinkId=74990</a> for more information.
	<b>CTRL + TILDE (~)</b>	Invokes available Developer Express Inc. refactorings.

### Navigation

Go to Definition	<b>F12</b> or <b>SHIFT + F2</b>	Moves to the declaration for the selected symbol. <i>(Edit.GoToDefinition)</i>
Navigate Backward	<b>CTRL + SHIFT + F2</b> or <b>CTRL + MINUS SIGN (-)</b>	Moves to the previously browsed line of code. <i>(View.NavigateBackward)</i>
Navigate Forward	<b>CTRL + SHIFT + MINUS SIGN (-)</b>	Moves to the next browsed line of code. <i>(View.NavigateForward)</i>
Previous Method or Type	<b>CTRL + UP ARROW</b> or <b>CTRL + PAGE UP</b>	Moves to the previous method or type in the document. <i>(Edit.PreviousMethod)</i>
Next Method or Type	<b>CTRL + DOWN ARROW</b> or <b>CTRL + PAGE DOWN</b>	Moves to the next method or type in the document. <i>(Edit.NextMethod)</i>
Previous and Next Word	<b>CTRL + LEFT ARROW</b> and <b>CTRL + RIGHT ARROW</b>	Moves the insertion point left or right one word, respectively. <i>(Edit.WordPrevious and Edit.WordNext)</i>
IDE Navigator	<b>CTRL + TAB</b>	Displays the IDE Navigator, with the first document window selected. The IDE Navigator functions similarly to the Windows Navigator (ALT + SHIFT + TAB), only it is for files and tool windows within Visual Studio. <i>(Window.NextDocumentWindowNav)</i>
View All Open Documents	<b>CTRL + ALT + DOWN ARROW</b>	Displays a pop-up listing of all open documents. <i>(Window.ShowEzMDIFileList)</i>
View Code	<b>F7</b>	Displays the selected item in Code view of the Editor. <i>(View.ViewCode)</i>
View Designer	<b>SHIFT + F7</b>	Displays the selected item in Design view of the Editor. <i>(View.ViewDesigner)</i>
HTML Designer	<b>SHIFT + F7</b>	Switches between <b>Source</b> view and <b>Design</b> view for the current HTML. <i>(View.ViewMarkup and View.ViewDesigner)</i>
View Document Outline	<b>CTRL + ALT + T</b>	Displays the <b>Document Outline</b> window. Particularly useful in the Form Designer and HTML Designer. <i>(View.DocumentOutline)</i>
Add or Remove Bookmark	<b>CTRL + K, CTRL + K</b>	Sets or removes a bookmark at the current line. <i>(Edit.ToggleBookmark)</i>
Navigate Bookmarks	<b>CTRL + K, CTRL + N</b> or <b>CTRL + K, CTRL + P</b>	Moves to the next or previous bookmark, respectively. <i>(Edit.NextBookmark and Edit.PreviousBookmark)</i>
Delete All Bookmarks	<b>CTRL + K, CTRL + L</b>	Deletes all bookmarks. <i>(Edit.ClearBookmarks)</i>

### Search

Find Symbol	<b>ALT + F12</b>	Displays the <b>Find Symbol</b> dialog box. <i>(Edit.FindSymbol)</i>
Find All References	<b>ALT + SHIFT + F12</b>	Displays a list of all references for the symbol selected. <i>(Edit.FindAllReferences)</i>
Find Text	<b>CTRL + F</b> and <b>CTRL + SHIFT + F</b>	Displays the <b>Find and Replace</b> dialog box for a single-file and multiple-file search, respectively. <i>(Edit.Find and Edit.FindInFiles)</i>
Next and Previous Search Result	<b>F3</b> and <b>SHIFT + F3</b>	Finds the next and previous occurrence, respectively, of the text from the most recent search. <i>(Edit.FindNext and Edit.FindPrevious)</i>
Next and Previous Selected	<b>CTRL + F3</b> and <b>CTRL + SHIFT + F3</b>	Finds the next and previous occurrence, respectively, of the currently selected text or the word at the insertion point. <i>(Edit.FindNextSelected and Edit.FindPreviousSelected)</i>
Incremental Search	<b>ALT + I</b> and <b>ALT + SHIFT + I</b>	Activates incremental search (forward and reverse). If no input is typed, the previous search query is used. <i>(Edit.IncrementalSearch and Edit.ReverseIncrementalSearch)</i>
Stop Search	<b>ALT + F3, S</b>	Halts the current <b>Find In Files</b> operation. <i>(Edit.StopSearch)</i>

### File

New Project	<b>CTRL + N</b> or <b>CTRL + SHIFT + N</b>	Displays the <b>New Project</b> dialog box. <i>(File.NewProject)</i>
Open Project	<b>CTRL + O</b> or <b>CTRL + SHIFT + O</b>	Displays the <b>Open Project</b> dialog box. <i>(File.OpenProject)</i>
Add New Item	<b>CTRL + SHIFT + A</b>	Displays the <b>Add New Item</b> dialog box. <i>(Project.AddNewItem)</i>
Add Existing Item	<b>CTRL + D</b>	Displays the <b>Add Existing Item</b> dialog box. <i>(Project.AddExistingItem)</i>

### Snippets

You can use Code Snippets Manager on the Tools menu to find the snippet shortcuts for the "Insert Snippet from Shortcut" commands listed below. You can also customize snippet shortcuts by editing the Shortcut tag of the snippet XML files. The Code Snippet Editor for Visual Basic 2005 is a free application that you can download to help do this. See <http://go.microsoft.com/fwlink/?LinkId=75013> for more information.

Insert Snippet	<b>Type "?" and press TAB</b>	Displays the Code Snippet Picker in the Code Editor. The selected code snippet is then inserted at the cursor position. <i>(Edit.InsertSnippet)</i>
Insert Snippet from Shortcut	<b>Type the snippet shortcut and press TAB</b>	Inserts the expanded code snippet. <i>(Edit.InvokeSnippetFromShortcut)</i>
Insert Property Snippet	<b>Type "property" and press TAB</b>	Inserts a Property snippet. <i>(Example of Edit.InvokeSnippetFromShortcut)</i>
List Snippet Shortcuts	<b>Type a snippet shortcut prefix, type "?", and press TAB</b>	Displays the Code Snippet Shortcut Picker. The shortcut in the list, which most closely matches the prefix, is selected.
List Snippet Replacements	<b>CTRL + SPACE</b>	Invokes an IntelliSense completion list for the currently selected snippet replacement.
Escape Replacement Selection	<b>ESC</b>	Deselects the current text. A second ESC deselects the replacement. Can be useful when you want to type at the end of a replacement without extending its bounds.

### IntelliSense

Make IntelliSense Transparent	<b>CTRL</b>	Hides the current IntelliSense window so that you can see the code beneath. Must be invoked while IntelliSense is active.
Display a Filtered List	<b>CTRL + J</b>	Displays the IntelliSense completion list for the current cursor position. <i>(Edit.ListMembers)</i>
Display the Global List or Complete a Word	<b>CTRL + SPACE</b>	If invoked while no list is active, displays the IntelliSense completion list for the current cursor position. If a substring has already been typed and there is an exact match in the list, completes the word without invoking the list. If invoked while a filtered list is active, switches to the global list. <i>(Edit.CompleteWord)</i>
Common Tab	<b>ALT + COMMA (,)</b>	Decreases the filter level of the active IntelliSense list to the <b>Common</b> tab.
All Tab	<b>ALT + PERIOD (.)</b>	Increases the filter level of the active IntelliSense list to the <b>All</b> tab.
Navigate Up	<b>CTRL + PAGE UP</b>	Navigates to the first item in the IntelliSense completion list.
Navigate Down	<b>CTRL + PAGE DOWN</b>	Navigates to the last item in the IntelliSense completion list.
Commit an Item	<b>TAB</b> or <b>SPACE</b> or <b>ENTER</b>	Inserts the currently selected item in the list. The following characters can also be used to commit: <code>{ } ( ) . ; + - * / ^ ! = &lt; &gt; \</code>
Escape the List	<b>ESC</b>	Closes the IntelliSense completion list. This can be useful if you want to prevent the currently selected item from being inserted.

## Introduction

This document contains the keyboard shortcuts that are set by default when you use the Visual Basic Development Settings. To set the Visual Basic Development Settings, on the **Tools** menu, click **Options**. In the Options dialog box, select **Show all settings**, and then, under **Environment**, click **Keyboard**. Click the Visual Basic 6 keyboard mapping scheme in the drop-down list and then click **OK**.

You can customize individual command shortcuts yourself. To do so, on the **Tools** menu, click **Options**. In the Options dialog box, select **Show all settings**, and then, under **Environment**, click **Keyboard**. To edit a shortcut, locate the command name in the dialog box and then assign it a shortcut key. In this document, names for the commands described are provided in parentheses.

For a complete list of commands, along with their keyboard shortcuts and detailed descriptions, see <http://go.microsoft.com/fwlink/?LinkId=74687>.

## Editing

Toggle All Outlining	<b>CTRL + M, CTRL + L</b>	Toggles all existing regions between collapsed and expanded states. <i>(Edit.ToggleAllOutlining)</i>
Toggle Outlining for Current Region	<b>CTRL + M, CTRL + M</b>	Toggles the current region between collapsed and expanded states. <i>(Edit.ToggleOutliningExpansion)</i>
Comment and Uncomment	<b>CTRL + K, CTRL + C</b> and <b>CTRL + K, CTRL + U</b>	Inserts and removes, respectively, the apostrophe (') at the beginning of the current line or every selected line. <i>(Edit.CommentSelection and Edit.UncommentSelection)</i>
Undo	<b>CTRL + Z</b>	Undoes the last action. <i>(Edit.Undo)</i>
Redo	<b>CTRL + SHIFT + Z</b>	Redoes the last action. <i>(Edit.Redo)</i>
Cut Line	<b>CTRL + Y</b>	Cuts the current line of code. <i>(Edit.LineCut)</i>
Insert Blank Line	<b>CTRL + ENTER</b> and <b>CTRL + SHIFT + ENTER</b>	Inserts a blank line above and below the cursor position, respectively. <i>(Edit.LineOpenAbove and Edit.LineOpenBelow)</i>
Select Word	<b>CTRL + SHIFT + W</b>	Selects the word containing, or to the right of, the insertion point. <i>(Edit.SelectCurrentWord)</i>
Delete Word	<b>CTRL + BACKSPACE</b> and <b>CTRL + DELETE</b>	Deletes to the beginning and end of the word, respectively. <i>(Edit.WordDeleteToStart and Edit.WordDeleteToEnd)</i>
Change Casing	<b>CTRL + U</b> and <b>CTRL + SHIFT + U</b>	Changes the selected text to lowercase and uppercase characters, respectively. <i>(Edit.MakeLowercase and Edit.MakeUppercase)</i>
Replace	<b>CTRL + H</b> and <b>CTRL + SHIFT + H</b>	Displays the <b>Quick Replace</b> tab and the <b>Replace In Files</b> tab, respectively, of the <b>Find and Replace</b> dialog box. <i>(Edit.Replace and Edit.ReplaceInFiles)</i>
Extend Selection	<b>SHIFT + ALT + UP ARROW</b> and <b>SHIFT + ALT + DOWN ARROW</b>	Moves the cursor one line up or down respectively, extending the line selection. <i>(Edit.LineUpExtendColumn and Edit.LineDownExtendColumn)</i>
Format Code	<b>CTRL + K, CTRL + D</b> and <b>CTRL + K, CTRL + F</b>	Formats the current document or selection, respectively. <i>(Edit.FormatDocument and Edit.FormatSelection)</i>
Display Smart Tag	<b>CTRL + DOT (.)</b> or <b>SHIFT + ALT + F10</b>	Displays the available options on the Smart Tag menu. <i>(View.ShowSmartTag)</i>
Help	<b>F1</b>	Displays a topic from Help that corresponds to the current user interface element or to the code item or error messages selected. <i>(Help.F1Help)</i>

## Build

Build Solution	<b>CTRL + SHIFT + B</b>	Builds all the projects in the solution. <i>(Build.BuildSolution)</i>
----------------	-------------------------	---

## Windows

Navigation Bar	<b>CTRL + F2</b>	Moves the cursor to the drop-down bar located at the top of the Code Editor. <i>(Window.MoveToNavigationBar)</i>
Object Browser	<b>F2</b>	Displays the <b>Object Browser</b> . <i>(View.ObjectBrowser)</i>
Properties Window	<b>F4</b>	Displays the <b>Properties</b> window for the currently selected item. <i>(View.PropertiesWindow)</i>
Solution Explorer	<b>CTRL + R</b>	Displays <b>Solution Explorer</b> . <i>(View.SolutionExplorer)</i>
View Server Explorer	<b>CTRL + ALT + S</b>	Displays <b>Server Explorer</b> . <i>(View.ServerExplorer)</i>
Show Data Sources	<b>SHIFT + ALT + D</b>	Displays the <b>Data Sources</b> window. <i>(Data.ShowDataSources)</i>
Toolbox	<b>CTRL + ALT + X</b>	Displays the <b>Toolbox</b> . <i>(View.Toolbox)</i>
Error List	<b>CTRL + \, E</b>	Displays the <b>Error List</b> . <i>(View.ErrorList)</i>
Close Tool Window	<b>SHIFT + ESC</b>	Closes the current tool window. <i>(Window.CloseToolWindow)</i>
Close Document Window	<b>CTRL + F4</b>	Closes the current tab. <i>(Window.CloseDocumentWindow)</i>

## Debugging

Start Debugging	<b>F5</b>	Starts the application in the debugger. When in Break mode, invoking this command runs the application until the next breakpoint. <i>(Debug.Start)</i>
Start Without Debugging	<b>CTRL + F5</b>	Starts the application without invoking the debugger. Use this exclusively rather than F5 for Web site debugging. <i>(Debug.StartWithoutDebugging)</i>
Step Into	<b>F8</b> or <b>F11</b>	Executes code one statement at a time, following execution into method calls. <i>(Debug.StepInto)</i>
Step Out	<b>CTRL + SHIFT + F8</b> or <b>SHIFT + F11</b>	Executes the remaining lines of the method in which the current execution point is located. <i>(Debug.StepOut)</i>
Step Over	<b>SHIFT + F8</b> or <b>F10</b>	Executes the next line of code, but does not follow execution into any method calls. <i>(Debug.StepOver)</i>
Stop Debugging	<b>CTRL + ALT + BREAK</b>	Stops running the current application in the debugger. <i>(Debug.StopDebugging)</i>
Toggle Breakpoint	<b>F9</b>	Sets or removes a breakpoint at the current line. <i>(Debug.ToggleBreakpoint)</i>
Set Next Statement	<b>CTRL + F9</b>	Sets the execution point to the line of code you choose. <i>(Debug.SetNextStatement)</i>
Break at a Function	<b>CTRL + B</b>	Displays the <b>New Breakpoint</b> window. <i>(Debug.BreakatFunction)</i>
Attach to Process	<b>CTRL + ALT + P</b>	Displays the <b>Attach to Process</b> dialog box. <i>(Tools.AttachToProcess)</i>
Make Data Tip Transparent	<b>CTRL</b>	Hides the current data tip so that you can see the code beneath. Must be invoked while a data tip is active.
Immediate Window	<b>CTRL + G</b>	Displays the <b>Immediate</b> window. <i>(Debug.Immediate)</i>
Call Stack Window	<b>CTRL + L</b>	Displays the <b>Call Stack</b> window. <i>(Debug.CallStack)</i>
QuickWatch Window	<b>SHIFT + F9</b>	Displays the <b>QuickWatch</b> dialog box. <i>(Debug.QuickWatch)</i>

## Refactoring

Refactor!	<b>CTRL + TILDE (~)</b>	Refactor! for Visual Basic 2005 is a free plug-in from Developer Express Inc., in partnership with Microsoft. Refactor! supports more than 15 individual refactoring features including Extract Method, Encapsulate Field, Reorder Parameters, Create Overload, and Surrounds With. See <a href="http://go.microsoft.com/fwlink/?LinkId=74990">http://go.microsoft.com/fwlink/?LinkId=74990</a> for more information.
		Invokes available Developer Express Inc. refactorings.

## Navigation

Go to Definition	<b>F12</b> or <b>SHIFT + F2</b>	Moves to the declaration for the selected symbol. <i>(Edit.GoToDefinition)</i>
Navigate Backward	<b>CTRL + SHIFT + F2</b> or <b>CTRL + MINUS SIGN (-)</b>	Moves to the previously browsed line of code. <i>(View.NavigateBackward)</i>
Navigate Forward	<b>CTRL + SHIFT + MINUS SIGN (-)</b>	Moves to the next browsed line of code. <i>(View.NavigateForward)</i>
Previous Method or Type	<b>CTRL + UP ARROW</b> or <b>CTRL + PAGE UP</b>	Moves to the previous method or type in the document. <i>(Edit.PreviousMethod)</i>
Next Method or Type	<b>CTRL + DOWN ARROW</b> or <b>CTRL + PAGE DOWN</b>	Moves to the next method or type in the document. <i>(Edit.NextMethod)</i>
Previous and Next Word	<b>CTRL + LEFT ARROW</b> and <b>CTRL + RIGHT ARROW</b>	Moves the insertion point left or right one word, respectively. <i>(Edit.WordPrevious and Edit.WordNext)</i>
IDE Navigator	<b>CTRL + TAB</b>	Displays the IDE Navigator, with the first document window selected. The IDE Navigator functions similarly to the Windows Navigator (ALT + SHIFT + TAB), only it is for files and tool windows within Visual Studio. <i>(Window.NextDocumentWindowNav)</i>
View All Open Documents	<b>CTRL + ALT + DOWN ARROW</b>	Displays a pop-up listing of all open documents. <i>(Window.ShowEzMDIFileList)</i>
View Code	<b>F7</b>	Displays the selected item in <b>Code</b> view of the Editor. <i>(View.ViewCode)</i>
View Designer	<b>SHIFT + F7</b>	Displays the selected item in <b>Design</b> view of the Editor. <i>(View.ViewDesigner)</i>
HTML Designer	<b>SHIFT + F7</b>	Switches between <b>Source</b> view and <b>Design</b> view for the current HTML. <i>(View.ViewMarkup and View.ViewDesigner)</i>
View Document Outline	<b>CTRL + ALT + T</b>	Displays the <b>Document Outline</b> window. Particularly useful in the Form Designer and HTML Designer. <i>(View.DocumentOutline)</i>
Add or Remove Bookmark	<b>CTRL + K, CTRL + K</b>	Sets or removes a bookmark at the current line. <i>(Edit.ToggleBookmark)</i>
Navigate Bookmarks	<b>CTRL + K, CTRL + N</b> or <b>CTRL + K, CTRL + P</b>	Moves to the next or previous bookmark, respectively. <i>(Edit.NextBookmark and Edit.PreviousBookmark)</i>
Delete All Bookmarks	<b>CTRL + K, CTRL + L</b>	Deletes all bookmarks. <i>(Edit.ClearBookmarks)</i>

## Search

Find Symbol	<b>ALT + F12</b>	Displays the <b>Find Symbol</b> dialog box. <i>(Edit.FindSymbol)</i>
Find All References	<b>ALT + SHIFT + F12</b>	Displays a list of all references for the symbol selected. <i>(Edit.FindAllReferences)</i>
Find Text	<b>CTRL + F</b> and <b>CTRL + SHIFT + F</b>	Displays the <b>Find and Replace</b> dialog box for a single-file and multiple-file search, respectively. <i>(Edit.Find and Edit.FindInFiles)</i>
Next and Previous Search Result	<b>F3</b> and <b>SHIFT + F3</b>	Finds the next and previous occurrence, respectively, of the text from the most recent search. <i>(Edit.FindNext and Edit.FindPrevious)</i>
Next and Previous Selected	<b>CTRL + F3</b> and <b>CTRL + SHIFT + F3</b>	Finds the next and previous occurrence, respectively, of the currently selected text or the word at the insertion point. <i>(Edit.FindNextSelected and Edit.FindPreviousSelected)</i>
Incremental Search	<b>ALT + I</b> and <b>ALT + SHIFT + I</b>	Activates incremental search (forward and reverse). If no input is typed, the previous search query is used. <i>(Edit.IncrementalSearch and Edit.ReverseIncrementalSearch)</i>
Stop Search	<b>ALT + F3, S</b>	Halts the current <b>Find In Files</b> operation. <i>(Edit.StopSearch)</i>

## File

New Project	<b>CTRL + N</b> or <b>CTRL + SHIFT + N</b>	Displays the <b>New Project</b> dialog box. <i>(File.NewProject)</i>
Open Project	<b>CTRL + O</b> or <b>CTRL + SHIFT + O</b>	Displays the <b>Open Project</b> dialog box. <i>(File.OpenProject)</i>
Add New Item	<b>CTRL + SHIFT + A</b>	Displays the <b>Add New Item</b> dialog box. <i>(Project.AddNewItem)</i>
Add Existing Item	<b>CTRL + D</b>	Displays the <b>Add Existing Item</b> dialog box. <i>(Project.AddExistingItem)</i>

## Snippets

You can use Code Snippets Manager on the Tools menu to find the snippet shortcuts for the "Insert Snippet from Shortcut" commands listed below. You can also customize snippet shortcuts by editing the Shortcut tag of the snippet XML files. The Code Snippet Editor for Visual Basic 2005 is a free application that you can download to help do this. See <a href="http://go.microsoft.com/fwlink/?LinkId=75013">http://go.microsoft.com/fwlink/?LinkId=75013</a> for more information.		
Insert Snippet	<b>Type "?" and press TAB</b>	Displays the Code Snippet Picker in the Code Editor. The selected code snippet is then inserted at the cursor position. <i>(Edit.InsertSnippet)</i>
Insert Snippet from Shortcut	<b>Type the snippet shortcut and press TAB</b>	Inserts the expanded code snippet. <i>(Edit.InvokeSnippetFromShortcut)</i>
Insert Property Snippet	<b>Type "property" and press TAB</b>	Inserts a Property snippet. <i>(Example of Edit.InvokeSnippetFromShortcut)</i>
List Snippet Shortcuts	<b>Type a snippet shortcut prefix, type "?", and press TAB</b>	Displays the Code Snippet Shortcut Picker. The shortcut in the list, which most closely matches the prefix, is selected.
List Snippet Replacements	<b>CTRL + SPACE</b>	Invokes an IntelliSense completion list for the currently selected snippet replacement.
Escape Replacement Selection	<b>ESC</b>	Deselects the current text. A second ESC deselects the replacement. Can be useful when you want to type at the end of a replacement without extending its bounds.

## IntelliSense

Make IntelliSense Transparent	<b>CTRL</b>	Hides the current IntelliSense® window so that you can see the code beneath. Must be invoked while IntelliSense is active.
Display a Filtered List	<b>CTRL + J</b>	Displays the IntelliSense® completion list for the current cursor position. <i>(Edit.ListMembers)</i>
Display the Global List or Complete a Word	<b>CTRL + SPACE</b>	If invoked while no list is active, displays the IntelliSense completion list for the current cursor position. If a substring has already been typed and there is an exact match in the list, completes the word without invoking the list. If invoked while a filtered list is active, switches to the global list. <i>(Edit.CompleteWord)</i>
Common Tab	<b>ALT + COMMA (,)</b>	Decreases the filter level of the active IntelliSense list to the <b>Common</b> tab.
All Tab	<b>ALT + PERIOD (.)</b>	Increases the filter level of the active IntelliSense list to the <b>All</b> tab.
Navigate Up	<b>CTRL + PAGE UP</b>	Navigates to the first item in the IntelliSense completion list.
Navigate Down	<b>CTRL + PAGE DOWN</b>	Navigates to the last item in the IntelliSense completion list.
Commit an Item	<b>TAB</b> or <b>SPACE</b> or <b>ENTER</b>	Inserts the currently selected item in the list. The following characters can also be used to commit: <code>{ } ( ) . , ; + - * / ^ ! = &lt; &gt; \</code>
Escape the List	<b>ESC</b>	Closes the IntelliSense completion list. This can be useful if you want to prevent the currently selected item from being inserted.