

# MinIDE version 2

## Usage Manual

Copyright © 2002 Dave Taubler. All rights reserved.

This software is provided "as is." The developer and distributor of this software makes no expressed or implied warranties of the merchantability or fitness of this software for any particular purpose. The developer and distributor of this software shall not be liable for any damages suffered by the user arising in any way out of the use of this software.



## General Information

### About the MinIDE program

This program is designed for use by software developers who want to develop on the Sharp Zaurus, or any other PDA that runs the Personal Java platform. It is designed for use as an IDE (Integrated Development Environment); that is, a tool to compose and, where relevant, compile software files. It was designed for use in developing Java applications, but can be used for developing in other programming and markup languages.

### Features

To help you program, the MinIDE application offers the following features:

- Auto-brackets (for the following bracket pairs – { }, [ ], ( ), < > – you can choose to have the pair automatically appear when you just type the first.)
- Ability to create custom auto-complete templates
- Auto-indenting
- Line break conversion (line breaks from other operating systems can be detected and converted to Linux line breaks [\n])
- Compiling (compiler must be pre-installed)
- Line number search
- Find and replace
- Font settings
- Save to specific filesystem location.

## How to Use the MinIDE Application

### Installing the application

Load the *minide\_1.2\_arm.ipk* file onto your Zaurus, either via the PC sync cable, or via Secure Digital or Compact Flash card. Go to the *Settings* tab and launch the *Add/Remove Software* application. Click *Install Packages* and select *minide*. Choose to where you want to install the application and select *OK*.

### Launching the application

Go to the Jeode tab and launch the MinIDE application.

### Dealing with files

To open a file, go to the *File* menu and select *Open...* A file dialog will appear. Navigate to the location of the file you wish to open and select *Ok*.

To create a new file, go to the *File* menu and select *New*. A blank editing screen will appear.

To save a file, go to the *File* menu and select *Save*. If the file has already been saved, the most recent changes will be saved to the same file. Otherwise, a file dialog will appear asking to where the file should be saved.

To save a file to the location of your choosing, go to the *File* menu and select *Save As...* A file dialog will appear asking to where the file should be saved.

Once more than one file has been opened, you can navigate among them by using the file combo box that appears above the editing screen (see Figure 1). It's worth noting that "open files" do not actually remain open in memory if they are not currently displayed in the editing screen. For that reason, when switching from File A to File B, all of File A's recent changes must be saved, or they will be lost.



**Figure 1:** Switch between open files using the file combo box.

### Using the Edit features

MinIDE has one level of Undos. To undo the most recent change, select the *Edit* menu and choose *Undo XXX*, where “XXX” describes the recent change made. To redo the change, select the *Edit* menu and then choose *Redo*, which will have replaced *Undo XXX*.

To find a word or phrase in a file, select *Edit* and then *Find*. In the dialog that appears, enter the word or phrase for which to search. To find the phrase again, go to the *Edit* menu and select *Find again*.

To find a word or phrase and replace it with another, select *Edit* and then *Replace*. In the dialog that appears, enter the word or phrase that you are searching for in the first textbox; enter the word or phrase that you want to replace it with in the second textbox. Click *Replace* to replace the first instance of the word/phrase found; you can subsequently use the *Edit -> Replace Again* menu item to replace the next instance found. Click *Replace All* to replace all instances in the file.

*Cut*, *Copy*, and *Paste*, can also be accessed via the *Edit* menu.

### Using the special tools

In-IDE compiling can be done via the *Tools* menu by accessing the menu and selecting *Compile*. A dialog will appear to display the compilation results. Note that this command assumes two things:

1. You have a compiler pre-installed
2. You have entered the “compile command” in the *Settings* menu. See the *Settings* section for details

Once you have compiled your code, a dialog will appear with the results of your compilation. By default, the *Compile Output* menu item in the *Tools* menu is deactivated. Once the results dialog appears, however, that menu item will be activated, allowing you to bring the dialog to the front while fixing your code. When you close the dialog, the menu item will once again become deactivated.

Jump to a specific line number in the open file via the *Tools* menu by selecting *Line #*. In the resulting dialog box, enter a number representing the line that you want to jump to.

Because you may first create files on a desktop PC, with a larger screen than on the Zaurus, you may have files that use tabs for indentation. Because of the limited screen real estate of the Zaurus, these files can be hard to read on the Zaurus. For this reason, MinIDE lets you convert tabs to the string of your choice. Select the *Tools* menu and choose *Convert tabs...* A dialog will appear, asking to what string tabs should be converted. Note that the default two spaces (“ ”) appears in the dialog. Either let the default remain, or enter your own string, then click *OK*.

Similarly, because you may be using files that were originally created on a different platform, and may therefore contain different line-break characters than the standard UNIX `\n`, you can convert your file’s existing line-breaks to the UNIX `\n` character by activating the *Tools* menu and choosing *Convert breaks to newline*.

### Settings

Some large files that have been created on other platforms may not open properly if they were saved on a platform that uses a line-break character other than the UNIX `\n` character. For that reason, you can tell Zaurus to convert these files’ line-break characters to `\n` while opening. Just activate the *Settings* menu and ensure that *Convert breaks on file open* item is checked.

To auto-close your brackets, use the *Settings* menu. You will find five items pre-pended with “Auto”. The first four correlate to sets of brackets (`{ }`, `[ ]`, `( )`, `< >`). For any brackets that you would like auto-closed, ensure that the pair is checked. For example, if *Auto { }* is checked, then when you type “{”, a “}” character will also automatically appear.

The fifth “Auto” item is *Auto Indent*. Ensure that this item is selected if you want auto indenting to be activated.

You can also create your own auto-completion templates. For example, if you are editing HTML files, you might want all of your `<p>` tags to be auto-completed with `</p>` tags. To do this, select *Settings -> Custom Auto Templates*. Then click *New...* in the resultant dialog box. Enter `<p>` in the *When entering this* textbox, and `</p>` in the *complete with* textbox. Then click *Save*. Be sure the *Activate* checkbox is selected, then click *Done*. To edit an auto-template that you’ve already created, select *Settings -> Custom Auto Templates* and select the desired template from the pulldown menu. Then click *Edit...*. Make the desired edits in the resulting dialog box. To remove this auto-template, leave the *When entering this* textbox empty. To temporarily prevent your auto-templates from being activated, de-select the *Activate* checkbox.

The *Settings* menu also allows you to adjust your font settings. Select *Font* to be presented with a font selection window. You can choose the relative size of your font (small, medium, and big) and the style of your font (fixed-width (*Fixed*) or variable-width (*Variable*)). Although fixed-width generally makes code more legible, some users may find that the variable-width font allows more code to be visible in the editing window.

As noted above, you can use the *Tools* menu to compile files that you have edited in MinIDE. First, you must enter the compile command as you would type it in the Terminal. Activate the *Settings* menu and select *Compile Command*. In the dialog that appears, enter the compile command, as you would type it into the Terminal (see Figure 2.) Be sure to EXCLUDE the filename; MinIDE will insert that for you.



**Figure 2:** Use *Settings* -> *Compile Command* to enter the terminal command used to compile your files.