

PHP Mode Manual

Editing PHP in GNU Emacs

Aaron S. Hawley <ashawley@sourceforge.net>

This manual documents PHP mode version 1.5.0 for use with GNU Emacs.

Copyright © 2008 Aaron S. Hawley

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, and no Cover Texts. A copy of the license is included in the section entitled “Copying This Manual.”

A copy of the license is also available from the Free Software Foundation Web site at <http://www.gnu.org/licenses/fdl.html>.

The document was typeset with GNU Texinfo.

\$Date: 2008-11-04 11:51:51 -0500 (Tue, 04 Nov 2008) \$

\$Revision: 90 \$

Table of Contents

1	Introduction.....	1
2	Installation.....	2
3	Electricity.....	4
4	Motion.....	5
5	Indentation.....	6
6	Editing.....	7
7	Completion.....	8
8	PHP Comments.....	10
9	PHP Documentation.....	11
10	Index Menu.....	12
11	Change Logs.....	13
12	Speedbar.....	14
13	Customization.....	15
	Appendix A Embedded HTML.....	16
	Appendix B Contributors.....	17
	Appendix C Copying This Manual.....	18
	Concept Index.....	25
	Function Index.....	28
	Keyboard Index.....	29
	Variable Index.....	30

1 Introduction

PHP mode is a major mode for editing PHP source code in Emacs. As an extension of Emacs's C mode, it inherits all C mode's navigation and editing functionality. Syntax elements of PHP source code is colored according to the grammar of PHP versions 3, 4 and 5. By default, the indent rules follow the PHP PEAR coding guidelines. Additional aids for developing in PHP include documentation search of the PHP manual, code completion and a class browser.

To find out what version of PHP mode you have installed run *M-x php-mode-version*.

PHP mode was first released in 1999 and was written by Turadg Aleahmad. At the time of PHP mode's initial release, version 2 of PHP (PHP/FI) was only 2 years old and the release of PHP 3 had happened the year before. In May of 2000, PHP 4 was released. Subsequent versions of PHP mode scurried to support the quickly evolving syntax of this emerging programming language.

In 2001, PHP mode became a SourceForge project and version 1.0 was released later that year. Further refinements of the Emacs library included improvements to the syntax coloring rules, portability between XEmacs and GNU Emacs, integration of various development tools available in various Emacsen.

PHP mode's support for PHP 5 first appeared in 2004, the same year PHP 5 was released. 2004 was the last year Aleahmad made a release. In 2005, Lennart Borgman combined a forked version of PHP mode with his "new XML mode for Emacs" (nXhtml). The stalled development of PHP resulted in people sharing various patches and improvements to PHP mode over the Internet.

The 1.4 release in 2008 represented a renewed commitment to the Emacs Lisp library with the addition of considerable updates and improvements – the first in over 2 years.

This manual describes PHP mode for practiced Emacs users especially those who have read or can refer to the Emacs manual. The reader should already know how to program in PHP, although one could easily learn how to program PHP using PHP mode and Emacs. There are no explanations of PHP programming in this manual.

2 Installation

PHP mode can be installed like any Emacs Lisp library.

Download a source archive of the file. It is available from PHP mode's Web site.

<http://php-mode.sourceforge.net/>

Download either the file 'php-mode-1.5.0.tar.gz' or 'php-mode-1.5.0.zip'. Both source files have a respective GnuPG¹ signature file with a .sig file name extension.

To verify the source archive you need the key of the maintainer. The public key can be downloaded from the Web site and imported, or imported from a key server.

```
$ gpg --recv-keys 8E372922
gpg: requesting key 8E372922 from hkp server keys.gnupg.net
gpg: key 8E372922: public key
"Aaron S. Hawley (SourceForge) <ashawley@users.sourceforge.net>"
imported
```

The public key can also be imported from a file.

```
$ gpg --import ashawley-sf-pubkey.asc
gpg: key 8E372922: public key
"Aaron S. Hawley (SourceForge) <ashawley@users.sourceforge.net>"
imported
gpg: Total number processed: 1
gpg:             imported: 1
```

Assign the trust for the maintainer key.

```
$ gpg --edit-key 8E372922
Command> trust
pub 1024D/8E372922  created: 2008-01-25  expires: 2010-01-01  usage: SC
                        trust: unknown      validity: unknown
sub 2048g/A4B44512  created: 2008-01-25  expires: 2010-01-01  usage: E
[ unknown] (1). Aaron S. Hawley (SourceForge)
                        <ashawley@users.sourceforge.net>
```

Please decide how far you trust this user to correctly verify other users' keys (by looking at passports, checking fingerprints from different sources, etc.)

```
1 = I don't know or won't say
2 = I do NOT trust
3 = I trust marginally
4 = I trust fully
5 = I trust ultimately
m = back to the main menu
```

Your decision? 1

¹ More information on GPG and OpenPGP at <http://www.gnupg.org/>

```
pub 1024D/8E372922  created: 2008-01-25  expires: 2010-01-01  usage: SC
                        trust: undefined  validity: unknown
sub 2048g/A4B44512  created: 2008-01-25  expires: 2010-01-01  usage: E
[ unknown] (1). Aaron S. Hawley (SourceForge)
                        <ashawley@users.sourceforge.net>
Command> quit
```

Verify a source package by using the GPG verify command on the signature file with the source file located in the same directory.

```
$ gpg --verify php-mode-1.4.0.tar.gz.sig
gpg: Signature made Fri 25 Jan 2008 01:30:24 PM EST
gpg: using DSA key ID 8E372922
gpg: Good signature from
"Aaron S. Hawley (SourceForge) <ashawley@users.sourceforge.net>"
Primary key fingerprint: B425 30C2 4F72 D4F1 FD01
                        D07D 7087 6760 8E37 2922
```

After the source package is verified, unpack the source archive.

```
$ tar xzf php-mode-1.5.0.tar.gz
```

The files are extracted to the directory ‘php-mode-1.5.0’. Run *make* in the directory to build all the formats of the documentation – Info, HTML, PostScript and PDF. Read the notes in the ‘Makefile’ to learn how to run *make* to compile the manual into a single particular format.

If you have a local directory where Emacs Lisp libraries are stored, for example ‘~/elisp’, then put the following in your ‘.emacs’ file if you haven’t already

```
(add-to-list 'load-path "~/elisp")
```

Then, put the file ‘php-mode.el’ into your Emacs load-path, ‘~/elisp’, and add php-mode to your ‘.emacs’ file.

```
(load "php-mode")
```

From inside Emacs, run *M-x customize-variable RET php-file-patterns RET*. This variable instructs Emacs what file names should be opened in PHP mode. If the defaults are fine, then click or hit *RET* on ‘Save for Future Sessions’.

Alternatively, put these lines in your ‘.emacs’ file.

```
(add-to-list 'auto-mode-alist
  '("\\.php[34]?\\'|\\.phtml\\'" . php-mode))
```

If you work on projects, like Drupal, that use other extension for PHP source file, you may need to add this to your ‘.emacs’.

```
(add-to-list 'auto-mode-alist '("\\.module\\'" . php-mode))
(add-to-list 'auto-mode-alist '("\\.inc\\'" . php-mode))
```

3 Electricity

Certain characters when inserted in Emacs have *electricity*. Emacs recognizes them as “electric” and automatically reacts in some way to their insertion. This is unlike the `self-insert-command` – the conventional manner Emacs inserts most characters into the buffer and without side-effect.

For example, the indent commands for programming language modes – including PHP mode – are electric. Hitting `TAB` (or `C-i`) typically indents the line to the correct column based on the nested declaration blocks or conditional (control) syntax statements. Other examples of electric characters include braces (`{` and `}`), characters related to comments (`#`, `/` and `*`), and semicolons. Inserting “electric” characters causes the line to be automatically be indented appropriately by PHP mode.

Should PHP mode ever do something you wish it didn’t, like indenting incorrectly after inserting a character when you wish it didn’t, then you’ve been *electrocuted*. To avoid this, use `quoted-insert` by prefixing the character with `C-q`. For example, `C-q TAB` for a tab. Alternatively, disable the electric effect of PHP mode with `C-c C-l`. Issue `C-c C-l` again to activate the electric effect again.

4 Motion

The traditional point movement commands for Emacs¹ and Emacs's C mode² are supported in PHP mode. The following are noteworthy commands for working on PHP code or are extensions expressly provided by PHP mode.

- C-M-f*** Move forward past the current syntax (**forward-sexp**). If point is in a string literal, move to the end of the string. If point is at the beginning of a parenthetical expression, move to the end of it. If at the beginning of statement block that is surrounded by braces, move to the end of the block. If point is at an identifier, skip to the end of it.
- C-M-b*** Same as ***C-M-f***, except a syntactic element move to its beginning (**backward-sexp**).
- C-M-a*** Move point to the beginning of the top-level function definition (**c-beginning-of-defun**).
- C-M-e*** Move point to the end of the top-level function definition (**c-end-of-defun**).
- C-M-<HOME>*** Move point to the beginning of the current function (**beginning-of-defun**).
- C-M-<END>*** Move point to the end of the current function (**end-of-defun**).
- M-m*** Move to the indentation at the beginning of the line (**back-to-indentation**).
- C-c C-w*** Enable word motion across mixed case identifiers with **forward-word** and **backward-word** commands (**c-subword-mode**).

Most movement commands support prefixing the command with a *numeric argument* by using Emacs's *universal argument* command. Type ***C-u n***, where *n* is the number of times to repeat a command. If no number *n* is provided, Emacs will repeat the command 4 times by default. See [Section “Numeric Arguments” in GNU Emacs Manual](#).

¹ See [Section “Changing the Location of Point” in GNU Emacs Manual](#).

² See [Section “Movement Commands” in CC Mode Manual](#).

5 Indentation

PHP mode’s indentation settings inherit the defaults set by Emacs C mode, or the rules customized for C mode.¹

PHP mode provides indentation settings that follow the PHP PEAR “Coding Standards”². It has the consequence in PHP mode of making the indentation commands use four spaces, and not tabs. This setting is turned on whenever the file name associated with a buffer includes the string “PEAR”, case insensitive and the file ends in “.php”.

You can force the use of this setting for all files edited with PHP mode by setting `php-mode-force-pear` to non-nil. Configure the variable with either *M-x customize-variable RET php-mode-force-pear RET*, or by setting its value in your Emacs initialization file (`~/.emacs`).

PHP does not appropriately indent HTML tags. See [Appendix A \[Embedded HTML\], page 16](#).

To customize PHP mode’s indentation rules, change the following variables.

`c-basic-offset`

This style variable of C mode controls how many columns in the level of indentation. By default this is 4 spaces in PHP mode (and in C mode).

`indent-tabs-mode`

Variable of Emacs that determines whether indentation commands should insert the tab character or insert the equivalent number of space characters. In PHP mode, this is set to to nil by default.

`tab-width`

This variable of Emacs determines how “wide” a tab in the buffer should be considered. This is also traditionally described as setting the distance between “tab stops”. By default, PHP mode does not insert tabs for indentation.

`php-mode-force-pear`

Use the indentation rules of PHP PEAR for all files opened in PHP mode. See above.

The usual commands for indenting in Emacs³ and Emacs’s C mode⁴: are supported in PHP mode, following the indentation rules of PHP.

<code>TAB</code>	Properly indent current line, even from the middle of the line (<code>c-indent-command</code>).
<code>C-j</code>	Insert text after point to a newline, and properly indent the newline (<code>c-indent-command</code>).
<code>C-M-\</code>	Properly indent each line of region (<code>indent-region</code>).
<code>C-c C-q</code>	Properly indent each line of current top-level function or top-level class (<code>c-indent-defun</code>).

¹ See Section “Customizing Indentation” in *CC Mode Manual*.

² <http://pear.php.net/manual/en/standards.php>

³ See Section “Indentation Commands” in *GNU Emacs Manual*.

⁴ See Section “Indentation Commands” in *CC Mode Manual*.

6 Editing

M-q Fill the paragraph to current value of `fill-column`. The command does the right thing for indented code and PHP comments including Docblock comments.

C-u n C-x f Set the `fill-column` to *n* when using the *M-q* command (`fill-paragraph`).

C-M-h Mark the current function (`c-mark-function`). Puts the mark at the end and of the function and the point at the beginning. Useful if you want to kill the region with *C-w* or *M-w*, or indent the region with *C-M-* or *C-x TAB* (See [Chapter 5 \[Indentation\]](#), page 6.).

M-x mark-defun
Like *C-M-h* (`c-mark-function`), except place the mark at the end of the function even if it's not at the top-level. With functions that are methods defined in a class, they may be indented and not top-level functions, however this command will only mark the current function.

The traditional commands for narrowing to a region in Emacs¹ are supported in PHP mode. The following commands are notable to mention for working with PHP or are extensions expressly provided PHP mode.

C-x n d Narrow the buffer to the contents of the current function (`narrow-to-defun`). This will narrow to the function as identified to the current function, even if it is nested in an object or another function. The logic to narrow the function is the same as the `beginning-of-defun` and `end-of-defun` commands – see [Chapter 4 \[Motion\]](#), page 5, and the `mark-defun` command – see above.

C-x n w Undo any narrowing by widening the buffer again by making the rest of the buffer available (`widen`).

¹ See Section “Narrowing” in *GNU Emacs Manual*.

7 Completion

Completing function names, identifiers and PHP keywords is possible in PHP mode. In PHP mode, it is possible to complete identifiers defined by the PHP programming language, and to complete functions defined by you.

For PHP mode to know the functions of keywords of PHP, it must have access to the PHP manual. Download the HTML archive of the manual made of separate files for each function.¹ Unpack the files to a directory, and define the variable `php-manual-path` to the location of these files. Configure this setting with either *M-x customize-variable RET php-manual-path RET*, or by setting its value in your Emacs initialization file (`~/.emacs`).

For functions and classes you define, you'll need to provide them to Emacs as a *tag file*.² Creating a tag file (`'TAGS'`) from a PHP source code file is done by running the `etags` command on the file.³ The `etags` command can even generate tag files for providing completion across multiple files. For larger projects and PHP coded libraries, use a system like a Makefile to generate and maintain the tag tables.

Run,

```
etags file
```

to create a `'TAGS'` file for *file*.

To generate a `'TAGS'` file for all the PHP files in the current directory, run

```
etags *.php
```

To enable a tags file you need to use the following command.

C-u M-x visit-tags-table

Specify the `'TAGS'` file that PHP mode should use to find completion candidates for the current buffer.

M-x visit-tags-table

Specify the `'TAGS'` file that PHP mode should use to find completion candidates for all buffers, and even other modes other than PHP mode.

If you are interested in any one or more of the following consider using the variable `php-completion-file`.

1. avoid regenerating the completion table from the PHP manual every Emacs session,
2. generating a completion file from a facility external to Emacs tags,
3. or provide more granular control over the completion rules for files;

Configure this setting with either *M-x customize-variable RET php-completion-file RET*, or by setting its value in your Emacs initialization file (`~/.emacs`).

To create a file that can be set for `php-completion-file`, create with the following shell commands with the HTML version of the PHP manual installed at `/usr/share/doc/php-manual/en/html`.

¹ <http://www.php.net/download-docs.php>

² See Section “Tags Tables” in *GNU Emacs Manual*.

³ See Section “Creating Tags Tables” in *GNU Emacs Manual*.

```
$ cd /usr/share/doc/php-manual/en/html
$ ls -l function*.html \
  | sed -e 's/^function\.\([-a-zA-Z_0-9]*\)\.html/\1/' \
  | tr - _ \
  > ~/.emacs.d/php/php-completion-file
```

In larger projects or with other specific needs, conventional methods of setting these and other variables may be inadequate. As of 2008, third-party packages do exist for Emacs to configure Emacs settings on groups of files or for directory trees, including ‘`dirvars.el`’.

M-TAB

ESC TAB

C-[TAB Complete function name at point (`php-complete-function`).

C-. Show in the minibuffer the argument list for the current function at point (`php-show-arglist`). The argument list is generated by finding and jumping to the respective function’s location specified in the tag file (see above). The argument list is then read and printed to the minibuffer.

8 PHP Comments

The usual commands for inserting, managing and editing source code comments in Emacs¹ and Emacs's C mode² are supported in PHP mode. The following noteworthy commands are useful for working with PHP or are extensions expressly provided by PHP mode.

- M-*; Insert comment at point (`comment-dwim`). If comment already exist, indent comment (`comment-indent`).
- M-j* Insert a newline in a comment on the next line with the comment starting at the same column as the current line (`c-indent-new-comment-line`).
- C-x ;* Set the current column as the column to start comments from in the buffer (`comment-set-column`). This affects the variable `comment-column`.
- C-c C-c* Comment out the current region (`comment-region`). If region is already commented, the region is uncommented.
- M-- C-x ;*
- C-u M-*; Kill the comment on the current line (`comment-kill` called by `comment-set-column` or `comment-dwim`).
- C-u n M-*; Kill the next *n* comments (`comment-dwim`).
- M-x comment-box*
 Surround region with comment characters.

¹ See Section “Manipulating Comments” in *GNU Emacs Manual*.

² See Section “Comment Commands” in *CC Mode Manual*.

9 PHP Documentation

PHP mode supports the online documentation available from <http://www.php.net/>. The `browse-url` system of Emacs supports these commands. Run the Emacs command `M-x customize-group RET browse-url RET` to configure how `browse-url` behaves. For example, you may want to change the Web browser Emacs uses for visiting a page of the manual for PHP.

`C-c C-m`

`C-c RET` Browse PHP manual in a Web browser.

`C-c C-f` Search PHP manual using identifier at point.

10 Index Menu

PHP mode provides a menu of the functions and classes defined in the current buffer. The menu is pretty simple, but is helpful when navigating large files. The contents of the menu also provide a good overview to the user about the file's contents.

To enable this menu, run the following command.

M-x imenu-add-menubar-index

Scan the file for functions and class declarations, then add a menu item called “Index” to the menu bar.

In the menu will be any one or more of the following menu items.

‘All Functions’

A sub-menu containing the names of functions defined in the current buffer. Clicking on a function name will visit the location of the function's definition.

‘Classes’ A sub-menu containing the names of classes defined in the current buffer. Clicking on a class name will visit the location of the class's definition.

‘Public Methods, Private Methods, Protected Methods’

Each of these three “method” sub-menus list the names of objects methods defined as “public”, “private” or “protected”. Clicking on a method name will visit the location of the function's definition.

‘*Rescan*’

Rescan the contents of the current buffer, and update both the indexes the entries in the menu.

To have Emacs automatically rescan the current buffer, change the setting for `imenu-auto-rescan` to non-nil with either *M-x customize-variable RET php-completion-file RET*, or by setting its value in your Emacs initialization file (`~/.emacs`).

11 Change Logs

PHP mode can help generate ‘ChangeLog’ files, a widely-adopted format for documenting the changes made to source code—especially software. PHP mode and Emacs will determine the current function, file name, author information and date and use it all to prepare a properly formatted ‘ChangeLog’ entry in the file.

The `add-change-log-entry` commands can be run from within a buffer containing a PHP buffer, or from a buffer in Diff mode with a revision patch to a PHP source file. See Section “Diff Mode” in *GNU Emacs Manual*.

C-x 4 a Add an entry in a ‘ChangeLog’ file opened in another window, or add a new line to the already begun entry (`add-change-log-entry-other-window`). Provides the file name and the respective function name, when applicable, for the current point. The point can be in a PHP source file, or in the hunk of a diff file.¹ This command will also add the current author and date to the entry.

¹ See Section “Diff Mode” in *GNU Emacs Manual*.

12 Speedbar

FIXME: Document speedbar support for PHP mode.

13 Customization

Customization for PHP mode can be done using Emacs's “Customize” feature¹, or by adding bits of Emacs Lisp to your Emacs initialization file (`~/.emacs`).

M-x customize-group RET php RET

Start a `*Customize*` buffer with the customizable settings for PHP mode.

¹ See Section “Customization” in *GNU Emacs Manual*.

Appendix A Embedded HTML

The PHP programming language is commonly embedded with HTML but also with other markup formats. Unfortunately, PHP mode is currently unable to recognize these other bits and provide commands that behave appropriately. Other Emacs modes for properly indenting HTML include HTML mode, SGML mode and XML mode.

For example, when editing the HTML sections found in a PHP file, consider turning on HTML mode with *M-x html-mode*. Use the the mode-specific commands for manipulating the HTML. When finished, turn back on PHP mode with *M-x php-mode* to return working on the PHP sections of the file.

Alternatively, consider using a *multiple major mode* like *mumamo*, *mmm-mode* or *multi-mode*. These modes allow a single buffer to appoint different major modes to respective portions of the buffer.

Appendix B Contributors

- Juanjo
- Torsten Martinsen
- Vinai Kopp
- Sean Champ
- Doug Marcey
- Kevin Blake
- Rex McMaster
- Mathias Meyer
- Boris Folgmann
- Roland Rosenfeld
- Fred Yankowski
- Craig Andrews
- John Keller
- Ryan Sammartino
- ppercot
- Valentin Funk
- Stig Bakken
- Gregory Stark
- Chris Morris
- Nils Rennebarth
- Gerrit Riessen
- Eric Mc Sween
- Ville Skytta
- Giacomo Tesio
- Lennart Borgman
- Stefan Monnier
- Aaron S. Hawley
- Aaron Iba
- Dias Badekas

Appendix C Copying This Manual

Version 1.2, November 2002

Copyright © 2000,2001,2002 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released

under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none. The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and

that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements.”

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called

an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C) year your name.  
Permission is granted to copy, distribute and/or modify this document  
under the terms of the GNU Free Documentation License, Version 1.2  
or any later version published by the Free Software Foundation;  
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover  
Texts. A copy of the license is included in the section entitled ‘‘GNU  
Free Documentation License’’.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with...Texts.” line with this:

```
with the Invariant Sections being list their titles, with  
the Front-Cover Texts being list, and with the Back-Cover Texts  
being list.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Concept Index

•	
.emacs	15
.emacs, auto-mode-alist	3
.emacs, Drupal	3
.emacs, imenu-auto-rescan	12
.emacs, load-path	3
.emacs, php-completion-file	8
.emacs, php-manual-path	8
.emacs, php-mode-force-pear	6
A	
add ChangeLog entry	13
argument list, function	9
B	
beginning of function	5
block	5
braces	4, 5
browse PHP manual	11
bumpy case	5
C	
C mode	1
C mode, indentation commands	6
C mode, indentation rules	6
C mode, movement	5
camel case	5
case, mixed	5
ChangeLog	13
ChangeLog entry	13
class index	12
class, indent	6
column, comment	10
column, fill	7
comment	10
comment box	10
comment column	10
comment newline	10
comment region	10
comment, Docblock	7
comment, filling	7
comment, indent	10
comment, insert	10
comment, kill	10
comments	4
complete function	9
completion	8
conditional statements	4
customization	15
customization, browse-url	11
customization, Imenu auto rescan	12
customization, PHP completion file	8
customization, PHP manual path	8
D	
declaration blocks	4
defun, mark	7
diff file	13
Diff mode	13
Docblock comment	7
documentatino, PHP mode	3
documentation, PHP	11
downloading PHP mode	2
Drupal	3
E	
electric indentation	4
electricity	4
embedded code	16
end of function	5
etags	8
etags, running	8
extension, PHP file name	3
extracting PHP mode	3
F	
features	1
file name extension, PHP	3
file, ChangeLog	13
file, diff	13
file, patch	13
fill column	7
filling	7
filling comment	7
filling indentation	7
function argument list	9
function index	12
function narrowing	7
function, beginning of	5
function, complete	9
function, end of	5
function, indent	6
function, mark	7
function, narrow	7
G	
GnuPG	2
GPG signature	2

H

HTML 6, 16

I

indent class 6
 indent comment 10
 indent function 6
 indent line 6
 indent newline 6
 indent region 6
 indentation, commands for 6
 indentation, electric 4
 indentation, filling 7
 indentation, force PEAR rules 6
 indentation, level of 6
 indentation, markup 16
 indentation, move to 5
 indentation, rules for 6
 index menu 12
 index, class 12
 index, function 12
 initialization file 15
 insert comment 10
 installing PHP mode 2
 installing php-mode.el 3

K

kill comment 10

L

line, indent 6
 loading php-mode.el 3

M

Makefile 8
 markup formats 16
 menu bar 12
 menu, index 12
 minibuffer 9
 mixed case 5
 mmm-mode 16
 motion 5
 movement 5
 multi-mode 16
 multiple major mode 16
 mumamo 16

N

narrowing 7
 newline 6
 newline, comment 10
 numeric argument 5

O

OpenPGP 2

P

paragraph, fill 7
 parentheses 5
 patch file 13
 PEAR 1
 PEAR, indentation 6
 PHP completion file 8
 PHP documentation 11
 PHP file name extension 3
 PHP manual path 8
 PHP mode documentation 3
 PHP mode Web site 2
 PHP mode, downloading 2
 PHP mode, installing 2
 PHP mode, unpacking 3
 PHP versions 1
 php-mode.el, installing 3
 php-mode.el, loading 3
 point 5

R

region, comment 10
 region, comment box 10
 region, indent 6
 rescan 12
 rescan, automatically 12

S

search PHP manual 11
 SGML 16
 spaces 6
 statements 5
 string 5
 syntax highlighting 1
 syntax, backward 5
 syntax, forward 5

T

tab 6
 tabs 6
 tag 8
 tag file 8

U

universal argument 5
 unpacking PHP mode 3

V

verify PHP mode with GPG..... 2
version, PHP 1
version, PHP mode 1

W

Web site, PHP mode 2
widening 7

X

XML..... 16

Function Index

A

add-change-log-entry 13
 add-change-log-entry-other-window 13

B

back-to-indentation 5
 backward-sexp 5
 backward-word 5
 beginning-of-defun 5
 browse-url 11

C

c-beginning-of-defun 5
 c-end-of-defun 5
 c-fill-paragraph 7
 c-indent-command 6
 c-indent-defun 6
 c-indent-new-comment-line 10
 c-mark-function 7
 c-subword-mode 5
 c-toggle-electric-state 4
 comment-box 10
 comment-dwim 10
 comment-indent 10
 comment-kill 10
 comment-region 10
 comment-set-column 10
 customize-group 11, 15
 customize-variable 3, 6, 8, 12

E

end-of-defun 5

F

fill-paragraph 7
 forward-sexp 5
 forward-word 5

H

html-mode 16

I

imenu 12

imenu-add-menubar-index 12
 imenu-update-menubar 12
 indent-region 6, 7
 indent-rigidly 7

K

kill-region 7
 kill-ring-save 7

M

mark-defun 7

N

narrow-to-defun 7
 newline-and-indent 6

P

php-browse-manual 11
 php-complete-function 9
 php-mode 3, 16
 php-mode-version 1
 php-search-documentation 11
 php-show-arglist 9

Q

quoted-insert 4

S

self-insert-command 4
 set-fill-column 7

U

universal-argument 5

V

visit-tags-table 8

W

widen 7

Keyboard Index

C

C-.....	9
C-[TAB.....	9
C-c C-c.....	10
C-c C-f.....	11
C-c C-l.....	4
C-c C-m.....	11
C-c C-q.....	6
C-c C-w.....	5
C-c RET.....	11
C-j.....	6
C-M-<END>.....	5
C-M-<HOME>.....	5
C-M-\.....	6, 7
C-M-a.....	5
C-M-b.....	5
C-M-e.....	5
C-M-f.....	5
C-M-h.....	7
C-q.....	4
C-u.....	5
C-w.....	7
C-x ;.....	10
C-x 4 a.....	13
C-x f.....	7
C-x n d.....	7
C-x n w.....	7

E

ESC TAB.....	9
--------------	---

M

M-- C-x ;.....	10
M-;.....	10
M-j.....	10
M-m.....	5
M-q.....	7
M-TAB.....	9
M-w.....	7
M-x comment-box.....	10
M-x comment-indent.....	10
M-x customize-group.....	15
M-x customize-variable.....	3, 6, 8, 12
M-x html-mode.....	16
M-x mark-defun.....	7
M-x php-mode.....	16
M-x php-mode-version.....	1
M-x visit-tags-table.....	8

T

TAB.....	4, 6
----------	------

Variable Index

A

add-log-current-defun-header-regexp..... 13

C

c-basic-offset..... 6

comment-column..... 10

F

fill-column..... 7

I

imenu-auto-rescan..... 12

indent-tabs-mode..... 6

L

load-path..... 3

P

php-beginning-of-defun-regexp..... 13

php-completion-file..... 8

php-manual-path..... 8

php-mode-force-pear..... 6

T

tab-width..... 6