

Vim (text editor)


Vim (/vɪm/^[2] a contraction of **Vi** **IM**proved) is a clone, with additions, of Bill Joy's vi text editor program for Unix. Vim's author, Bram Moolenaar, based it on the source code for a port of the Stevie editor to the Amiga^[3] and released a version to the public in 1991. Vim is designed for use both from a command-line interface and as a standalone application in a graphical user interface. Vim is free and open-source software and is released under a license that includes some charityware clauses, encouraging users who enjoy the software to consider donating to children in Uganda.^[4] The license is compatible with the GNU General Public License through a special clause allowing distribution of modified copies "under the GNU GPL version 2 or any later version".^[5]

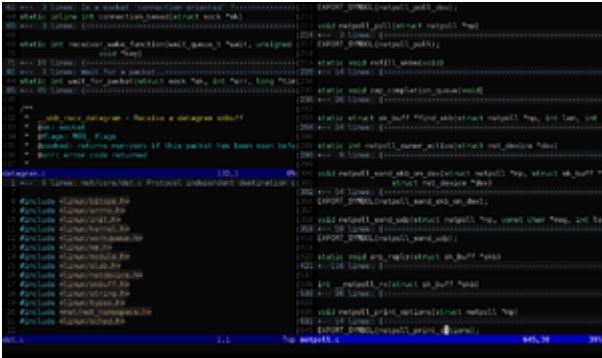
Since its release for the Amiga, cross-platform development has made it available on many other systems. In 2006, it was voted the most popular editor amongst *Linux Journal* readers;^[6] in 2015 the Stack Overflow developer survey found it to be the third most popular text editor,^[7] and in 2019 the fifth most popular development environment.^[8]

Contents
History
Interface
Modes
Customization
Features and improvements over vi
Vim script
Examples
Availability
Neovim
See also
References
External links

History

Vim





Vim running in a terminal emulator

Original author(s)	Bram Moolenaar
Initial release	2 November 1991
Stable release	8.2.0000 ^[1] (13 December 2019) [±] (https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Vim&action=edit)
Repository	github.com/vim/vim (https://github.com/vim/vim)
Written in	C , Vim script
Operating system	Unix , Linux , Windows NT , MS-DOS , macOS , iOS , Android , Haiku , AmigaOS , MorphOS
Type	Text editor
License	Free software (Vim License (https://vimhelp.org/uganda.txt.html#license)), charityware
Website	www.vim.org (https://www.vim.org/)

Vim's forerunner, Stevie (ST Editor for VI Enthusiasts), was created by Tim Thompson for the Atari ST in 1987^{[9][10]} and further developed by Tony Andrews^{[9][11]} and G.R. (Fred) Walter.^{[12][13]}

Basing his work on Stevie, Bram Moolenaar began working on Vim for the Amiga computer in 1988, with the first public release (Vim v1.14) in 1991.^{[14][15]}

At the time of its first release, the name "Vim" was an acronym for "Vi IMitation", but this changed to "'Vi IMproved" late in 1993.^[16]

Release history		
Date	Version	Changes and additions
June, 1987	N/A	Tim Thompson releases Stevie (ST editor for VI enthusiasts), a limited vi clone for the <u>Atari ST</u> , posting the source on <u>Usenet</u> . ^{[9][10]}
June, 1988	N/A	Tony Andrews improves Stevie, and ports it to Unix and <u>OS/2</u> , releasing version 3.10 on <u>Usenet</u> . ^{[9][11]}
1988	1.0	Bram Moolenaar creates Vi IMitation for the Amiga, based on Stevie, never publicly released
November 2, 1991	1.14 ^[17]	First public release for the <u>Amiga</u> on <u>Fred Fish</u> disk #591 ^[18]
1992	1.22 ^[17]	Port to Unix, never publicly released. ^[16] Vim now competes with <u>vi</u> .
December 14, 1993	2.0 ^[19]	This is the first release using the name Vi IMproved . ^[16]
August 12, 1994	3.0 ^[17]	Support for multiple windows
May 29, 1996	4.0 ^{[17][20]}	<u>Graphical user interface</u>
February 19, 1998	5.0 ^{[17][21]}	<u>Syntax highlighting</u> , basic <u>scripting</u> (user defined functions, commands, etc.)
April 6, 1998	5.1	<u>Bug fixes</u> , various improvements
April 27, 1998	5.2	Long line support, file browser, dialogs, popup menu, select mode, session files, user defined functions and commands, <u>Tcl</u> interface, etc.
August 31, 1998	5.3	Bug fixes, etc.
July 25, 1999	5.4	Basic file encryption, various improvements
September 19, 1999	5.5	Bug fixes, various improvements
January 16, 2000	5.6	New syntax files, bug fixes, etc.
June 24, 2000	5.7	New syntax files, bug fixes, etc.
May 31, 2001	5.8	New syntax files, bug fixes, etc.
September 26, 2001	6.0 ^{[17][22]}	<u>Folding</u> , <u>plugins</u> , multi-language, etc.
March 24,	6.1	Bug fixes

2002		
June 1, 2003	6.2	GTK2 and libgnome2 support, Arabic language support, :try command, minor features, bug fixes
June 7, 2004	6.3	Bug fixes, translation updates, mark improvements
October 15, 2005	6.4	Bug fixes, updates to Perl, Python, and Ruby support
May 7, 2006	7.0 ^[23]	Spell checking, code completion, tab pages (multiple viewports/window layouts), current line and column highlighting, undo branches, and more
May 12, 2007	7.1	Bug fixes, new syntax and runtime files, etc.
August 9, 2008	7.2 ^[24]	Floating point support in scripts, refactored screen drawing code, bug fixes, new syntax files, etc.
August 15, 2010	7.3	<u>Lua</u> support, Python3 support, <u>Blowfish</u> encryption, persistent undo/redo
August 10, 2013	7.4 ^[25]	A new, faster regular expression engine.
September 12, 2016	8.0 ^[26]	Asynchronous I/O support, jobs, lambdas, etc.
May 18, 2018	8.1 ^[27]	Terminal window support and terminal gdb plugin.
December 13, 2019	8.2 ^[28]	Popup windows, text properties.

Interface



Code at the top (Go), opened files, registers ("clipboard manager" and macros history)

Like vi, Vim's interface is not based on menus or icons but on commands given in a text user interface; its GUI mode, gVim, adds menus and toolbars for commonly used commands but the full functionality is still expressed through its command line mode. Vi (and by extension Vim) tends to allow a typist to keep their fingers on the home row, which can be an advantage for a touch typist.^[29]



Graphical Vim (gVim) under GTK+ 2.

Vim has a built-in tutorial for beginners called vimtutor. It's usually installed along with Vim, but it exists as a separate executable and can be run with a shell command.^[30] There is also the Vim Users' Manual that details Vim's features and a FAQ (https://vimhelp.appspot.com/vim_faq.txt.html). This manual can be read from within Vim, or found online.^{[31][32]}

Vim also has a built-in help facility (using the `:help` command) that allows users to query and navigate through commands and features.

Modes

Vim has 12 different editing modes, 6 of which are variants of the 6 basic modes.^[33] The basic modes are:

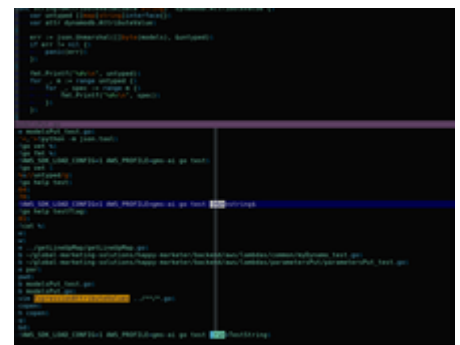
- **Normal** mode – used for editor commands. This is also the default mode, unless the `insertmode` option is specified.
- **Visual** mode – similar to normal mode, but used to highlight areas of text. Normal commands can be run on the highlighted area, for instance to move or edit a selection.
- **Select** mode – works similarly to visual mode. However, if a printable character, carriage return, newline or line feed is entered, Vim inserts the character, and starts insert mode.^[34]
- **Insert** mode – similar to editing in most modern editors. In this mode, buffers can be modified with the text inserted.
- **Command-line** or **Cmdline** mode – supports a single line input at the bottom of the Vim window. Normal commands (beginning with `:`), and some other keys for specific actions (including pattern search and the filter command) activate this mode. On completion of the command, Vim returns to the previous mode.
- **Ex** mode – similarly to Cmdline mode, prompts for command input at the bottom of the window. On completion of the command, Vim prompts for another Ex mode command.



Search (grep) inside Vim across files on disk, without plugins

Customization

Vim is highly customizable and extensible, making it an attractive tool for users who demand a large amount of control and flexibility over their text editing environment.^[35] Text input is facilitated by a variety of features designed to increase keyboard efficiency. Users can execute complex commands with "key mappings," which can be customized and extended. The "recording" feature allows for the creation of macros to automate sequences of keystrokes and call internal or user-defined functions and mappings. Abbreviations, similar to macros and key mappings, facilitate the expansion of short strings of text into longer ones and can also be used to correct mistakes. Vim also features an "easy" mode for users looking for a simpler text editing solution.^[36]



Commands history below: we can edit every command and/or run it again

There are many plugins available that extend or add new functionality to Vim, such as linters (<https://github.com/dense-analysis/ale>), integration with Git (<https://github.com/airblade/vim-gitgutter>), showing colors in CSS (<https://github.com/ap/vim-css-color>). These complex scripts are usually written in Vim's internal scripting language, vimscript (also known as VimL),^[37] but can be written in other languages as well.

There are projects bundling together complex scripts and customizations and aimed at turning Vim into a tool for a specific task or adding a major flavour to its behaviour. Examples include Cream, which makes Vim behave like a click-and-type editor, or VimOutliner, which provides a comfortable outliner for users of Unix-like systems.

Features and improvements over vi

Vim has a vi compatibility mode, but when that mode isn't used, Vim has many enhancements over vi.^[38] However, even in compatibility mode, Vim is not entirely compatible with vi as defined in the Single Unix Specification^[39] and POSIX (e.g., Vim does not support vi's open mode, only visual mode). Vim's developers state that it is "very much compatible with Vi".^[40]

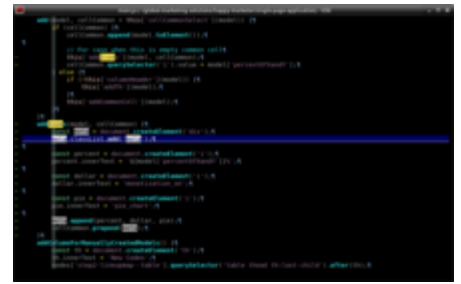
Some of Vim's enhancements include completion, comparison and merging of files (known as vimdiff), a comprehensive integrated help system, extended regular expressions, scripting languages (both native and through alternative scripting interpreters such as Perl, Python, Ruby, Tcl, etc.) including support for plugins, a graphical user interface (known as gvim), limited integrated development environment-like features, mouse interaction (both with and without the GUI), folding, editing of compressed or archived files in gzip, bzip2, zip, and tar format and files over network protocols such as SSH, FTP, and HTTP, session state preservation, spell checking, split (horizontal and vertical) and tabbed windows, Unicode and other multi-language support, syntax highlighting, trans-session command, search and cursor position histories, multiple level and branching undo/redo history which can persist across editing sessions, and visual mode.

While running, Vim saves the user's changes in a swap file with the ".swp" extension. The swap file can be used to recover after a crash. If a user tries to open a file and a swap file already exists, Vim will warn the user, and if the user proceeds, Vim will use a swap file with the extension ".swo" (or, if there is already more than one swap file, ".swn", ".swm", etc.).^{[41][42]} This feature can be disabled.^[43]

Vim script

Vim script (also called **Vimscript** or **VimL**)^[44] is the scripting language built into Vim.^[45] Based on the ex editor language of the original vi editor, early versions of Vim added commands for control flow and function definitions. Since version 7, Vim script also supports more advanced data types such as lists and dictionaries and (a simple form of) object-oriented programming. Built-in functions such as map() and filter() allow a basic form of functional programming, and Vim script has lambda since version 8.0. Vim script is mostly written in an imperative programming style.

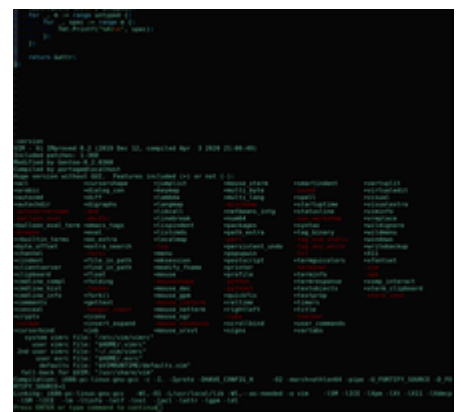
Vim macros can contain a sequence of *normal-mode* commands, but can also invoke ex commands or functions written in Vim script for more complex tasks. Almost all extensions (called plugins or more commonly scripts) of the core Vim functionality are written in Vim



Vim editing JavaScript, with installed plugin GitGutter: pluses at the left show new lines (according to information from Git).



File manager inside vim



:version command display
compilation flags (supported features)

script, but plugins can also utilize other languages like [Perl](#),^[46] [Python](#),^[47] [Lua](#),^[48] [Ruby](#),^[49] [Tcl](#),^[50] or [Racket](#).^[51] These plugins can be installed manually, or through a plugin manager such as [Vundle](#), [Pathogen](#), or [Vim-Plug](#).

Vim script files are stored as plain text, similarly to other code, and the filename extension is usually `.vim`. One notable exception to that is Vim's config file, `.vimrc`.

Examples

```
" This is the Hello World program in Vim script.
echo "Hello, world!"

" This is a simple while loop in Vim script.
let i = 1
while i < 5
    echo "count is" i
    let i += 1
endwhile
unlet i
```

Availability

Whereas `vi` was originally available only on [Unix](#) operating systems, Vim has been [ported](#) to many operating systems including [AmigaOS](#) (the initial target platform), [Atari MiNT](#), [BeOS](#), [DOS](#), [Windows](#) starting from [Windows NT 3.1](#), [OS/2](#), [OS/390](#), [MorphOS](#), [OpenVMS](#), [QNX](#), [RISC OS](#), [Linux](#), [BSD](#), and [Classic Mac OS](#).^[52] Also, Vim is shipped with every copy of [Apple macOS](#).^[53]

Independent ports of Vim are available for [Android](#)^{[54][55]} and [iOS](#).^[56]

Neovim

Neovim^[57] is a fork – with additions^[58] – of Vim that strives to improve the extensibility and maintainability of Vim.^[59] Neovim has the same configuration syntax as Vim; thus the same [configuration file](#) can be used with both editors, although there are minor differences in details of options.^[60] If the added features of Neovim are not used, Neovim is compatible with almost all of Vim's features.^[61]



The Neovim project was started in 2014, with some Vim community members offering early support of the high-level refactoring effort to provide better scripting, plugins, and integration with modern [GUIs](#).^{[62][63]} The project is [free software](#) and its source code is available on [GitHub](#).^[64]



Screenshot of Neovim

Neovim had a successful fundraising in March 2014,^[65] supporting at least one full-time developer. Several frontends are under development, making use of Neovim's capabilities.^{[66][67][68]}

The Neovim editor is available in a [personal package archive](#),^[69] hosted by [Ubuntu](#) and some more conventional package managers,^[70] making it possible to install it on a variety of operating systems.

See also

- *Learning the vi and Vim Editors*, a tutorial book for vi and vim, published by O'Reilly Media
- *Editor war* – the rivalry between users of the Emacs and vi (Vim) text editors
- *List of text editors*
- *Comparison of text editors*
- *Vimperator*
- *Pentadactyl*



Neovim with a status bar plugin

References


1. "Releases" (<https://github.com/vim/vim/releases>). Retrieved 13 December 2019 – via *GitHub*.
2. Vim documentation: intro (<https://vimhelp.org/intro.txt.html>): "Vim is pronounced as one word, like Jim, not vi-ai-em. It's written with a capital, since it's a name, again like Jim."
3. Zapletal, Lukáš (18 April 2005), "Interview: Bram Moolenaar" (<http://www.linuxexpres.cz/rozhovor/rozhovor-bram-moolenaar?highlightWords=Bram+moolenaar>), *LinuxEXPRES*: 21–22, retrieved 5 February 2015, "Is VIM derivate of other VI clone or you started from scratch? I started with Stevie. This was a Vi clone for the Atari ST computer, ported to the Amiga. It had quite a lot of problems and could not do everything that Vi could, but since the source code was available I could fix that myself. (English translation (http://www.moolenaar.net/interv_czech_en.html))"
4. "Vim documentation: uganda" (<https://vimhelp.org/uganda.txt.html#license>). *vimhelp.org*.
5. "Vim documentation: uganda" (<https://vimhelp.org/uganda.txt.html>).
6. "Linux Journal: 2003 Readers' Choice Awards" (<http://www.linuxjournal.com/article/7029#N0x850ca10.0x85cf4c4>). 1 November 2003. Retrieved 24 May 2006.; "Linux Journal: 2004 Readers' Choice Awards" (<http://www.linuxjournal.com/article/7724#N0x850cd80.0x85d3e3c>). 1 November 2004. Retrieved 24 May 2006.; "Linux Journal: 2005 Readers' Choice Awards" (<http://www.linuxjournal.com/article/8520#N0x850cd80.0x87983bc>). 28 September 2005. Retrieved 24 May 2006.
7. "Stack Overflow Developer Survey 2015 § IV. Text Editor" (<https://insights.stackoverflow.com/survey/2015#tech-editor>). Stack Overflow. Retrieved 25 July 2016.
8. "Stack Overflow Developer Survey 2019 Results" (<https://insights.stackoverflow.com/survey/2019#development-environments-and-tools>). Stack Overflow § VII. Development Environments. Retrieved 20 July 2019.
9. Thompson, Tim (26 March 2000). "Stevie" (<http://nosuch.com/tjt/stevie/>). Retrieved 27 December 2010.
10. Tim Thompson (28 June 1987). "A mini-vi for the ST" (<https://groups.google.com/group/comp.sys.atari.st/msg/8db96f888d778a32?dmode=source>). Newsgroup: comp.sys.atari.st (news:comp.sys.atari.st). Usenet: 129@glimmer.UUCP (news:129@glimmer.UUCP). Retrieved 27 December 2010.
11. Tony Andrews (6 June 1988). "v15i037: Stevie, an "aspiring" VI clone for Unix, OS/2, Amiga" (<https://groups.google.com/group/comp.sources.unix/msg/1fccf6a82259beed?dmode=source>). Newsgroup: comp.sources.unix (news:comp.sources.unix). Usenet: 893@fig.bbn.com (news:893@fig.bbn.com). Retrieved 27 December 2010.
12. Vim (20 January 2015). "intro.txt" (<https://web.archive.org/web/20160709041643/https://vimhelp.appspot.com/intro.txt.html>). *Vim Help*. Vim. Archived from the original (<https://vimhelp.appspot.com/intro.txt.html>) on 9 July 2016. Retrieved 9 July 2016.

13. "vim(1)" (<https://web.archive.org/web/20160709133642/http://linux.die.net/man/1/vim>). *die.net*. Vim. 11 April 2006. Archived from the original (<http://linux.die.net/man/1/vim>) on 9 July 2016. Retrieved 9 July 2016. "Vim is based on Stevie, worked on by: Tim Thompson, Tony Andrews and G.R. (Fred) Walter. Although hardly any of the original code remains."
14. "The continuing story of Vim" (<http://moolenaar.net/vimstory.pdf>) (PDF).
15. "The history of Vim – Jovica Ilic" (<https://jovicailic.org/2014/06/the-history-of-vim/>). Retrieved 25 January 2020.
16. "VILE (Vi Like Emacs) – Frequently Asked Questions (FAQ)" (https://invisible-island.net/vile/vile.faq.html#clone_began). Retrieved 7 September 2019.
17. Moolenaar, Bram (15 January 2002). "Vim, an open-source text editor" (<http://www.free-soft.org/FSM/english/issue01/vim.html>). Retrieved 24 October 2005.
18. "Textfiles.com" (http://cd.textfiles.com/fredfish/v1.6/FF_Disks/571-600/FF_591/Contents).
19. "Filewatcher" (<https://web.archive.org/web/20110711001335/http://www.filewatcher.com/b/ftp/ftp.twaren.net/pub/Unix/Editors/Vim/old.0.0.html>). Archived from the original (<http://www.filewatcher.com/b/ftp/ftp.twaren.net/pub/Unix/Editors/Vim/old.0.0.html>) on July 11, 2011. Retrieved February 26, 2011.
20. "Official Vim Manual, Version 4 summary" (<http://www.vim.org/html/doc/version4.html>). 12 March 2004. Retrieved 6 August 2008.
21. "Official Vim Manual, Version 5 summary" (<http://www.vim.org/html/doc/version5.html>). 17 January 2004. Retrieved 6 August 2008.
22. "Official Vim Manual, Version 6 summary" (<http://www.vim.org/html/doc/version6.html>). 12 March 2004. Retrieved 6 August 2008.
23. "Vim Reference Manual, Version 7" (<https://vimhelp.org/version7.txt.html>). 17 July 2016. Retrieved 13 January 2019.
24. "Google Groups" (https://groups.google.com/group/vim_announce/browse_thread/thread/2c89671dd928812f). *groups.google.com*.
25. Google Discussiegroepen (https://groups.google.com/forum/#!topic/vim_announce/knOQ_t_H5to). *Groups.google.com*. Retrieved on 2013-12-09.
26. Bram Moolenaar. "Vim 8.0 released!" (https://groups.google.com/forum/#!topic/vim_announce/EKTuhjF3ET0). Retrieved 12 September 2016.
27. Bram Moolenaar. "Vim 8.1 is released!" (<https://www.vim.org/vim-8.1-released.php>). Retrieved 18 May 2018.
28. Bram Moolenaar. "Vim 8.2 is released!" (<https://www.vim.org/vim-8.2-released.php>). Retrieved 13 December 2019.
29. Lamb, Linda; Robbins, Arnold (1998). *Learning the Vi Editor* (<https://archive.org/details/learningvieditor00lamb/page/305>). O'Reilly Media, Inc. p. 305 (<https://archive.org/details/learningvieditor00lamb/page/305>). ISBN 9781565924260.
30. Moolenaar, Bram (3 November 2010). "Vim documentation: usr_01" (http://vimdoc.sourceforge.net/html/doc/usr_01.html#tutor). Retrieved 28 August 2019.
31. Vim help files (<https://vimhelp.org/>) at vimhelp.org
32. Oualline, Steve (April 2001). *Vi IMproved (VIM)* (<ftp://ftp.vim.org/pub/vim/doc/book/vimbook-OP.L.pdf>) (PDF). New Riders Publishers. ISBN 0-7357-1001-5.
33. Moolenaar, Bram (8 December 2010). "Vim documentation: intro" (<http://vimdoc.sourceforge.net/html/doc/intro.html#vim-modes>). *vimdoc.sourceforge.net*. Retrieved 28 August 2019.
34. Moolenaar, Bram (17 February 2010). "Vim documentation: visual" (<http://vimdoc.sourceforge.net/html/doc/visual.html#Select>). *vimdoc.sourceforge.net*. Retrieved 28 August 2019.
35. Melendez, Steven. "Why Vim, An '80s Text Editor, Is Still The UI Of Choice For Power Users" (<https://www.fastcompany.com/3037629/why-vim-an-80s-text-editor-is-still-the-ui-of-choice-for-power-users>). *FastCompany*. Retrieved 5 May 2019.

36. "Tips: Making Vim easy" (<https://www.linux.com/news/tips-making-vim-easy>). *Linux.com*. 10 April 2007. Retrieved 6 May 2019.
37. "Vim documentation: usr_41" (https://vimhelp.org/usr_41.txt.html#vim-script-intro). *vimhelp.org*.
38. Vim help system (type `":help (https://vimhelp.org/)"` within Vim)
39. The Open Group (2008), "*vi — screen-oriented (visual) display editor*", *Single Unix Specification, Version 4 (IEEE Std 1003.1–2008)* (<http://pubs.opengroup.org/onlinepubs/9699919799/utilities/vi.html>), retrieved 27 December 2010
40. Peppe; Benji; Campbell, Charles (2 January 2004). "Vim FAQ" (https://vimhelp.org/vim_faq.txt.html#faq-1.3). Retrieved 27 December 2010. (question 1.3)
41. "Vim documentation: recover" (<http://vimdoc.sourceforge.net/html/doc/recover.html>). *vimdoc.sourceforge.net*. Retrieved 17 December 2020.
42. "How to handle swapfiles in Vim" (<https://cs.longwood.edu/VimSwap.html>). *cs.longwood.edu*. Retrieved 17 December 2020.
43. "'swapfile' option" (<https://vimhelp.org/options.txt.html#swapfile>). *vimhelp.org*. 10 October 2020. Retrieved 18 December 2020.
44. Klein, Benjamin. "The VimL Primer: Edit Like a Pro with Vim Plugins and Scripts by Benjamin Klein | The Pragmatic Bookshelf" (<https://web.archive.org/web/20200125123547/https://pragprog.com/book/bkviml/the-viml-primer>). Archived from the original (<https://pragprog.com/book/bkviml/the-viml-primer>) on 25 January 2020. Retrieved 25 January 2020.
45. "Vim documentation: usr_41" (https://vimhelp.org/usr_41.txt.html). *vimhelp.org*.
46. Verdoolaege, Sven; Gerassimof, Matt. "Vim documentation: if_perl" (http://vimdoc.sourceforge.net/html/doc/if_perl.html). Retrieved 27 August 2019.
47. Moore, Paul. "Vim documentation: if_pyth" (http://vimdoc.sourceforge.net/html/doc/if_pyth.html). Retrieved 27 August 2019.
48. Carvalho, Luis. "Vim documentation: if_lua" (http://vimdoc.sourceforge.net/html/doc/if_lua.html). Retrieved 27 August 2019.
49. Maeda, Shugo. "Vim documentation: if_ruby" (http://vimdoc.sourceforge.net/html/doc/if_ruby.html). Retrieved 27 August 2019.
50. Wilken, Ingo. "Vim documentation: if_tcl" (http://vimdoc.sourceforge.net/html/doc/if_tcl.html). Retrieved 27 August 2019.
51. Khorev, Sergey. "Vim documentation: if_mzsch" (http://vimdoc.sourceforge.net/html/doc/if_mzsch.html). Retrieved 27 August 2019.
52. `":help sys-file-list (https://vimhelp.org/#sys-file-list)"`
"Vim Online: Downloads" (<http://www.vim.org/download.php>). Retrieved 7 January 2007.
53. "Mac OS X Manual Page For vim(1)" (<https://developer.apple.com/library/mac/#DOCUMENTATION/Darwin/Reference/ManPages/man1/vim.1.html>). *developer.apple.com*. Apple Inc. Retrieved 12 January 2010.
54. "VimTouch, the development has stalled on this app" (<https://github.com/momodalo/vimtouch>). Retrieved 9 August 2015.
55. "DroidVim, under active development" (<https://github.com/shiftrot/droidvim>). Retrieved 5 March 2017.
56. "Vim – Applidium, mobile agency in Paris" (<https://web.archive.org/web/20120120032008/http://applidium.com/en/applications/vim/>). Archived from the original (<http://applidium.com/en/applications/vim/>) on 20 January 2012. Retrieved 11 August 2015.
57. "Neovim: vim, out of the box" (<http://neovim.io/>).
58. "Nvim documentation: vim_diff" (https://neovim.io/doc/user/vim_diff.html).
59. "Neovim Vision" (<https://neovim.io/charter/>).
60. "Switching to NeoVim" (<http://arusahni.net/blog/2015/03/switching-to-neovim-part-1.html>).

61. "How to start using Neovim instead of Vim" (<http://veelenga.com/editors/how-to-start-using-neo-vim-instead-of-vim/>).
62. "Neovim: Rebuilding Vim For the 21st Century" (<http://news.slashdot.org/story/14/03/22/147222/neovim-rebuilding-vim-for-the-21st-century>).
63. "Vimcasts.org blog post" (<http://vimcasts.org/blog/2014/03/support-neovim/>).
64. "GitHub – neovim/neovim: Vim-fork focused on extensibility and usability" (<https://github.com/neovim/neovim>). 29 August 2019.
65. "Neovim" (<https://www.bountysource.com/teams/neovim/fundraiser>). *Bountysource*. 23 March 2014. Retrieved 20 March 2021.
66. "NyaoVim frontend" (<https://github.com/rhysd/NyaoVim>). 29 August 2019.
67. "Mac OS X frontend" (<https://github.com/rogual/neovim-dot-app>). 29 August 2019.
68. "Neovim-Qt frontend" (<https://github.com/equelsraf/neovim-qt>). 21 July 2020.
69. "Neovim PPA information" (<https://launchpad.net/~neovim-ppa/+archive/ubuntu/unstable>).
70. "Neovim wiki: installation instructions" (<https://github.com/neovim/neovim/wiki/Installing-Neovim>). 29 August 2019.

External links

- Official website (<https://www.vim.org/>) 
 - TechTalk by Bram Moolenaar held on 25th Anniversary of first vim release (https://www.youtube.com/watch?v=ayc_qpB-93o) on YouTube
-

Retrieved from "[https://en.wikipedia.org/w/index.php?title=Vim_\(text_editor\)&oldid=1013162356](https://en.wikipedia.org/w/index.php?title=Vim_(text_editor)&oldid=1013162356)"

This page was last edited on 20 March 2021, at 11:02 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.