Click to verify

An online notepad is a plain text editor that you can use to create, edit, and save text-based documents in real-time on the Internet. It is the best Wordpad online over conventional notepad software that you can use offline.How to Use Our Free Online Notepad?You can use it quickly and easily with only a few steps.Open the online text editor by entering the URL ( )Once the tool is open, you can start typing or paste your text in the provided space. You can also edit an existing document by uploading a file from your local storage.Use the provided formatting options to style your text as needed (e.g., bold, italic, bullet points).After the document is complete, you can Duplicate, Download, or Share it. If you close the website or browser window, your text will be automatically saved.Why should you use our Online Text Editor?Text Editor Online is a web-based tool that allows users to create, edit, format, and manipulate text directly within a web browser. It serves as a convenient and accessible editpad for working with text-based content without the need to install specialized software on your device. Here are some of the key functions of our free online notepad:Quick Note-takingFormatting TextSharing and CollaborationCreating and Editing Text DocumentsKey Features of Our Wordpad OnlineThis online wordpad offers a range of features designed to help users easily create, organize, and manage their notes. Here are some key features you might find in this tool.1. User-friendly Interface:Online Notepad has a very user-friendly interface. All the options you will need are displayed clearly around the interface. Even if you are a beginner and don’t have much experience using word processors or text editors, you can do your work easily with an online notepad.2. Text Editing and Formatting:There are multiple text formatting options that you can utilize when using our online text editor. These options can help you style your text as you want by bolding it, italicizing it, underlining it, adding lists, etc.3. Auto-Save:It automatically saves your work as you type, helping you avoid accidental data loss.4. Sync Across Devices:Our notepad online is quite different from normal text editors since it allows you to create an account and sync all files/documents across different platforms. You can access and edit your notes on computers, smartphones, and tablets as long as you're connected to the internet.5. Notes Search:To quickly locate specific notes, our online notepad usually has a search bar that helps you search through your notes by keywords, tags, or titles.6. Collaboration and Sharing:With our online Wordpad, you can create notes and seamlessly share them with other individuals over the Internet. Each online file has a unique URL you can send to anyone.This can be a convenient way to share files and allow multiple users to work on the same note simultaneously.7. Full-Screen ModeYou can type out your documents in the normal mode, i.e., with all the toolbars and tabs on your browser visible and showing. But if you want to make the process more immersive, you can click on the “Full Screen” button. This will spread the entire tool interface across the screen.8. Dark ModeIt has a dark mode that you can use to change the interface to a black-dominant look. If you want to work on your documents in a dark setting, you can use this feature to make everything easier on the eyes.9. Download FileYou can also download your files to your local storage. This can help you share the files more easily and open them on your device’s native apps if required.10. Upload FileYou can upload files from your local storage to work on them instead of creating new documents from scratch. The good thing about the file uploading feature is that you can fetch files in any text-related format. You upload:DOCX filesDOC files (the older version of DOCX)TXT filesPDF files11. Duplicate NotesThis is also a fairly useful option that you can enjoy when using our WordPad. The “Duplicate” feature basically opens up a copy of the current document in another tab.12. Cross-Platform Compatibility:This free notepad can be accessed through web browsers, making it compatible with various operating systems and devices without needing specific software installations.13. User RegistrationUnlike other text editors and word processors, our online notepad allows you to create an account and register as a user. This helps you synchronize your work across all the devices or platforms you work on.14. Multilingual Support:This Notepad Online offers Multilingual Support to meet the needs of our global users. It breaks down language barriers and offers a versatile writing experience.It supports more than 20 languages, including English, Spanish, Indonesian, Russian, French, German, Mandarin, Japanese, Korean, Italian, Portuguese, Dutch, Turkish, Swedish, Arabic, Norwegian, Polish, Vietnamese, Ukrainian, Hindi, and Bengali.FAQsHow to bold text in Notepad?You can bold text in an online notepad by selecting the bold option from the formatting toolbar or using a shortcut key (Ctrl+B).How to count words in Notepad?The total word count is displayed at the bottom right corner of the text box. You can count your words and characters from there.How to write notes online?You can take notes online using our free notepad tool. Just open it up, start typing, format your text, and download it all back to your device. Online Notepad is a free browser-based text editor that allows you to create and edit multiple plain-text files in your browser. No registration or login required. It is great for writing quick notes and printing simple pages. What makes it special is the autosave functionality, which saves your draft every second. This prevents data loss in case you accidentally close the tab or the browser window suddenly crashes. The document you're working on will be automatically restored when you visit again, even when you close and reopen your browser. There's also support for saving documents directly to your computer. Online Notepad is packed with core features that your common text editor provides, including undo, redo, copy, cut, paste, find and replace, font formatting, character map, insert date and time, emoji list, spell checker, word counter, open and save files, and print page. How does it work? Changes are saved automatically as you work thanks to HTML5 localStorage API. It grabs a copy of the content from the text editor and saves it to your computer. With this method, your data never leaves your device. The default time interval at which drafts are saved is 1000ms or 1 second. Your notes will stay in the web browser until you clear the cookies and other site data. You can try it out by typing anything in the editor and refreshing the page. Browser Compatibility Online Notepad supports modern web browsers including Google Chrome, Mozilla Firefox, Safari, Opera, Edge, Internet Explorer and Steam browser. You need to enable JavaScript in order to use the app. About This web app is a free product that can be used by any individual, company, school, government office, etc. I originally made this for myself to help me remember ideas that would eventually end up on my personal blog. This simple tool, which started as a simple project, has helped my productivity immensely, and I hope it helps you too. Features Our online notepad comes with several powerful features that make it a versatile tool for your writing needs: Rich Text Formatting: Although primarily a plain-text editor, you can format your text with basic styling options like bold, italic, and underline when needed. Keyboard Shortcuts: Increase your productivity with common keyboard shortcuts for operations like copy (Ctrl+C), paste (Ctrl+V), undo (Ctrl+Z), and more. Mobile Friendly: The editor is fully responsive and works seamlessly across all devices, from smartphones to desktop computers. No Sign-up Required: Start writing immediately without any registration process or login requirements. Tips for Using Online Notepad To get the most out of your Online Notepad experience: Use the 'Save As' feature regularly to backup important notes to your computer Take advantage of the find and replace feature for quick text editing Use the word counter to track your writing progress Remember that your notes persist between sessions unless you clear your browser data Privacy & Security We take your privacy seriously. Since all data is stored locally in your browser using localStorage, your notes remain private and secure. No data is transmitted to or stored on our servers. This makes it perfect for sensitive information, though we recommend backing up important notes to your local device. Quick Access For convenient access to Online Notepad, we recommend bookmarking this page in your browser. This will allow you to quickly access your notes with just one click, making it easier to jot down ideas or access your saved content whenever you need it. Most browsers allow you to bookmark a page by pressing Ctrl+D (Windows/Linux) or Command+D (Mac), or by clicking the star icon in your browser's address bar. DeepL TranslatorTranslate securely, efficiently, and accuratelyDeepL WriteEnhance your writing and adapt to your audienceDeepL VoiceConverse across languages in real timeDeepL APIBuild multilingual experiences into your productsDeepL IntegrationsCombine Language AI with essential productivity tools transitive verb 1 a : to form (characters, symbols, etc.) by or as if by marking on a surface b : to form (words) by or as if by marking characters or symbols on a surface c : to spell in writing words written alike but pronounced differently d : to cover, fill, or fill in by writing 2 : to set down in writing: such as a : draw up, draft c : to express in literary form If I could write the beauty of your eyes …—William Shakespeare d : to communicate by letter writes that they are coming e : to use or exhibit (a specific script, language, or literary form or style) in writing f : to write contracts or orders for especially : underwrite 3 : to make a permanent impression of 4 : to communicate with in writing we'll write you when we get there 5 : ordain, fate So be it; it is written.—D. C. Peattie 6 : to make evident or obvious guilt written on his face 7 : to force, effect, introduce, or remove by writing ... write oneself into fame and fortune.—Charles Lee 8 : to take part in or bring about (something worth recording) 9 a : to introduce (information) into the storage device or medium of a computer b : to transfer (information) from the main memory of a computer to a storage or output device 10 : sell intransitive verb 1 a : to make significant characters or inscriptions also : to permit or be adapted to writing b : to form or produce written letters, words, or sentences 2 : to compose, communicate by, or send a letter 3 a : to produce a written work Welcome to Calmly Writer Onlinean aesthetic, minimalist and distraction-free text editor Start writing Need help? The internet is for connecting with people. Write.as helps you reach them no matter where they live online. Friends and family can read your words without creating yet another account online. They can read in their browser, via email or RSS, on their e-reader, and on federated platforms like Mastodon. Read more: Starting a newsletter → Enabling eBook downloads → Joining the fediverse → Read this blog → Focus on your words Write.as is a pared-down blogging platform — a distraction-free place for you to write, and a clean space for your audience to read. Instead of content management, we focus on writing. You'll see our editor screen first, every time you open the app. You can write in Markdown or switch to our friendly Rich Text editor. Your work is automatically saved to your browser as you write. Try it out → Build a home for your writing, outside of walled gardens and locked-down platforms. Add your domain name to keep control of your web presence. Make it your own with a custom theme. Then, know you're never stuck here — you can always export your work, or self-host your site with WriteFreely, the open source software that powers Write.as. Free your thoughts Built entirely around privacy, we don't track you or gather excessive data, so you can write exactly what you feel. Create a professional blog next to one written under a pen name. Publicize your work, or keep it quiet. Write anonymously, or create as many identities as you like. On Write.as, you have full control over who knows what about you. Write.as helps you create a blog, journal, or newsletter that readers will love to read. Create a home for your words, then fill it with writing — by yourself or with others. $6 per month, billed yearly For prolific writers. Customize, build a readership, and publish however you like. Up to three blogs on write.as or your own domain RSS feed ActivityPub integration Micropayments Unlimited posts Unlimited social cross-posts Password-protected blogs Private blogs Publish to Read Write.as Rich text editing Email subscriptions Photo hosting Pinned pages Full design customization API access Command-line publishing Email publishing Team support Start free 14-day trial Subscribe monthly, yearly, or every five years. $25 per month, billed yearly For serious publishers. Gather your team and manage content with our optimized workflow. Everything in Pro One shared blog Asynchronous collaborative editing Team blog access Team photo hosting Five team members User roles Post signatures E-book export Priority support Start free 30-day trial Subscribe monthly or yearly. Write.as is part of Musing Studio, a suite of simple tools for creating and sharing on the open web. You can also gather writing submissions with Submit.as, share photos with Snap.as, and socialize with Remark.as. Read about our company and our principles, and subscribe to our blog. write(2) System Calls Manual write(2) write - write to a file descriptor Standard C library (libc, -lc) #include ssize_t write(size_t count; int fd, const void buf[count], size_t count); write() writes up to count bytes from the buffer starting at buf to the file referred to by the file descriptor fd. The number of bytes written may be less than count if, for example, there is insufficient space on the underlying physical medium, or the RLIMIT_FSIZE resource limit is encountered (see setrlimit(2)), or the call was interrupted by a signal handler after having written less than count bytes. (See also pipe(7).) For a seekable file (i.e., one to which lseek(2) may be applied, for example, a regular file) writing takes place at the file offset, and the file offset is incremented by the number of bytes actually written. If the file was open(2)ed with O_APPEND, the file offset is first set to the end of the file before writing. The adjustment of the file offset and the write operation are performed as an atomic step. POSIX requires that a read(2) that can be proved to occur after a write() has returned will return the new data. Note that not all filesystems are POSIX conforming. According to POSIX.1, if count is greater than SSIZE_MAX, the result is implementation-defined; see NOTES for the upper limit on Linux. On success, the number of bytes written is returned. On error, -1 is returned, and errno is set to indicate the error. Note that a successful write() may transfer fewer than count bytes. Such partial writes can occur for various reasons; for example, because there was insufficient space on the disk device to write all of the requested bytes, or because a blocked write() to a socket, pipe, or similar was interrupted by a signal handler after it had transferred some, but before it had transferred all of the requested bytes. In the event of a partial write, the caller can make another write() call to transfer the remaining bytes. The subsequent call will either transfer further bytes or may result in an error (e.g., if the disk is now full). If count is zero and fd refers to a regular file, then write() may return a failure status if one of the errors below is detected. If no errors are detected, or error detection is not performed, 0 is returned without causing any other effect. If count is zero and fd refers to a file other than a regular file, the results are not specified. EAGAIN The file descriptor fd refers to a file other than a socket and has been marked nonblocking (O_NONBLOCK), and the write would block. See open(2) for further details on the O_NONBLOCK flag. EAGAIN or EWOULDBLOCK The file descriptor fd refers to a socket and has been marked nonblocking (O_NONBLOCK), and the write would block. POSIX.1-2001 allows either error to be returned for this case, and does not require these constants to have the same value, so a portable application should check for both possibilities. EBADF fd is not a valid file descriptor or is not open for writing. EDESTADDRREQ fd refers to a datagram socket for which a peer address has not been set using connect(2). EDQUOT The user's quota of disk blocks on the filesystem containing the file referred to by fd has been exhausted. EFAULT buf is outside your accessible address space. EFBIG An attempt was made to write a file that exceeds the implementation-defined maximum file size or the process's file size limit, or to write at a position past the maximum allowed offset. EINTR The call was interrupted by a signal before any data was written; see signal(7). EINVAL fd is attached to an object which is unsuitable for writing; or the file was opened with the O_DIRECT flag, and either the address specified in buf, the value specified in count, or the file offset is not suitably aligned. EIO A low-level I/O error occurred while modifying the inode. This error may relate to the write-back of data written by an earlier write(), which may have been issued to a different file descriptor on the same file. Since Linux 4.13, errors from write-back come with a promise that they may be reported by subsequent. write() requests, and will be reported by a subsequent fsync(2) (whether or not they were also reported by write()). An alternate cause of EIO on networked filesystems is when an advisory lock had been taken out on the file descriptor and this lock has been lost. See the Lost locks section of fcntl(2) for further details. ENOSPC The device containing the file referred to by fd has no room for the data. EPERM The operation was prevented by a file seal; see fcntl(2). EPIPE fd is connected to a pipe or socket whose reading end is closed. When this happens the writing process will also receive a SIGPIPE signal. (Thus, the write return value is seen only if the program catches, blocks or ignores this signal.) Other errors may occur, depending on the object connected to fd. POSIX.1-2008. SVr4, 4.3BSD, POSIX.1-2001. Under SVr4 a write may be interrupted and return EINTR at any point, not just before any data is written. A successful return from write() does not make any guarantee that data has been committed to disk. On some filesystems, including NFS, it does not even guarantee that space has successfully been reserved for the data. In this case, some errors might be delayed until a future write(), fsync(2), or even close(2). The only way to be sure is to call fsync(2) after you are done writing all your data. If a write() is interrupted by a signal handler before any bytes are written, then the call fails with the error EINTR; if it is interrupted after at least one byte has been written, the call succeeds, and returns the number of bytes written. On Linux, write() (and similar system calls) will transfer at most 0x7ffff000 (2,147,479,552) bytes, returning the number of bytes actually transferred. (This is true on both 32-bit and 64-bit systems.) An error return value while performing write() using direct I/O does not mean the entire write has failed. Partial data may be written and the data at the file offset on which the write() was attempted should be considered inconsistent. According to POSIX.1-2008/SUSv4 Section XSI 2.9.7 ("Thread Interactions with Regular File Operations"): All of the following functions shall be atomic with respect to each other in the effects specified in POSIX.1-2008 when they operate on regular files or symbolic links: ... Among the APIs subsequently listed are write() and writev(2). And among the effects that should be atomic across threads (and processes) are updates of the file offset. However, before Linux 3.14, this was not the case: if two processes that share an open file description (see open(2)) perform a write() (or writev(2)) at the same time, then the I/O operations were not atomic with respect to updating the file offset, with the result that the blocks of data output by the two processes might (incorrectly) overlap. This problem was fixed in Linux 3.14. close(2), fcntl(2), fsync(2), ioctl(2), lseek(2), open(2), pwrite(2), read(2), select(2), writev(2), fwrite(3) This page is part of the man-pages (Linux kernel and C library user-space interface documentation) project. Information about the project can be found at ⟨ ��. If you have a bug report for this manual page, see ⟨ ��. This page was obtained from the tarball man-pages-6.15.tar.gz fetched from ⟨ �� on 2025-08-11. If you discover any rendering problems in this HTML version of the page, or you believe there is a better or more up- to-date source for the page, or you have corrections or improvements to the information in this COLOPHON (which is not part of the original manual page), send a mail to man-pages@man7.org Pages that refer to this page: ps(1), pv(1), strace(1), telnet-probe(1), close(2), epoll_ctl(2), eventfd(2), fcntl_locking(2), F_GET_SEALS(2const), F_NOTIFY(2const), fsync(2), getpeername(2), getrlimit(2), io_uring_enter2(2), io_uring_enter(2), lseek(2), memfd_create(2), mmap(2), open(2), pipe(2), pread(2), read(2), readv(2), seccomp(2), select(2), select_tut(2), send(2), sendfile(2), socket(2), socketpair(2), sync(2), syscalls(2), aio_error(3), aio_return(3), aio_write(3), curs_print(3x), dbopen(3), fclose(3), fflush(3), fgetc(3), fopen(3), fread(3), gets(3), io_uring_prep_write(3), io_uring_prep_writev2(3), io_uring_prep_writev(3), libexpect(3), mkfifo(3), mpool(3), printf(3), puts(3), size_t(3type), stdio(3), wprintf(3), xdr(3), xfsctl(3), dsp56k(4), fuse(4), lirc(4), st(4), proc_pid_io(5), proc_sys_kernel(5), systemd.exec(5), aio(7), cgroups(7), cpuset(7), epoll(7), fanotify(7), inode(7), inotify(7), landlock(7), pipe(7), sched(7), signal(7), signal-safety(7), socket(7), spufs(7), tcp(7), time_namespaces(7), udp(7), user_namespaces(7), vsock(7), x25(7), fsfreeze(8), netsniff-ng(8), wipefs(8), xfs_io(8)