

# Ignoring directories in Git repositories on Windows

Asked 10 years, 11 months ago   Active 17 days ago   Viewed 957k times

▲ How can I ignore directories or folders in Git using msysgit on Windows?

992

windows

git

gitignore

git-bash

msysgit



166

edited Nov 14 at 13:48



Peter Mortensen

24.1k 19 89 118

asked Dec 5 '08 at 12:17



sf.

17.3k 10 44 56

Do you want the cache folder to be excluded from the repository completely, or just its contents? – Gareth Jun 29 '12 at 10:09

I'm guessing that the OP has multiple cache directories, whose contents should be ignored, but wants to make sure that those directories are created for anyone who clones the repository. – Mark Longair Jun 29 '12 at 10:19

@Gareth: since empty folders aren't tracked with git, if the content is ignored, the folder also will be, won't it? – eckes Jun 29 '12 at 10:34

2 I only mentioned that because some people use hidden files (commonly `.gitkeep`) to indicate a directory that should be kept – Gareth Jun 29 '12 at 18:17

2 is the `/` at the end what makes it know its a directory it should ignore? – Charlie Parker Jan 18 '17 at 21:59

## 17 Answers

▲ Create a file named `.gitignore` in your project's directory. Ignore directories by entering the directory name into the file (with a slash appended):

1459

`dir_to_ignore/`



More information is [here](#).

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



Peter Mortensen

24.1k 19 89 118



stew

14.9k 2 14 8

- 125 In a windows cmd prompt you can either use 'edit .gitignore' or 'notepad .gitignore' to create the correct file. – [Joey Green](#) May 9 '11 at 16:08
- 43 Or you can also use 'touch .gitignore' from within the windows git bash command prompt and that will create the correctly named file which can then in turn be edited by notepad or the like... – [SGB](#) Nov 22 '11 at 21:48
- 38 Or just create a file named .gitignore. with explorer and edit it with notepad (the trailing dot will be removed). That way you don't have to use command prompt. – [P. Galbraith](#) Mar 30 '12 at 0:01
- 6 Or you could vim .gitignore from the terminal. :) – [MollyCat](#) Aug 19 '13 at 23:33
- 33 Or, which surprisingly has not been mentioned even though it is the fastest way, just type "echo folder\_to\_ignore>> .gitignore" in the console. – [Godsmith](#) Feb 9 '14 at 6:49

▲ You can create the ".gitignore" file with the contents:

7

```
*
!.gitignore
```

▼ It works for me.

edited Nov 14 at 23:00

answered Feb 5 '15 at 14:27



Claudionor Oliveira

1,166 1 19 30

- 2 This is the best answer, it also brings added benefits in project deployment and maintenance. – [Nitin...](#) Jun 20 '16 at 9:24
- Who is "*simples*"? – [Peter Mortensen](#) Nov 14 at 14:28

▲ On Windows and Mac, if you want to ignore a folder named Flower\_Data\_Folder in the current directory, you can do:

4

```
echo Flower Data Folder >> .gitignore
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
echo data.txt >> .gitignore
```

If it's a path like "Data/passwords.txt"

```
echo "Data/passwords.txt" >> .gitignore.
```

edited Nov 14 at 14:31



Peter Mortensen

24.1k 19 89 118

answered Aug 14 '18 at 11:17



Olusola Omosola

379 2 6

On Unix:

5

```
touch .gitignore
```

On Windows:

```
echo > .gitignore
```

These commands executed in a terminal will create a `.gitignore` file in the current location.

Then just add information to this `.gitignore` file (using Notepad++ for example) which files or folders should be ignored. Save your changes. That's it :)

More information: [.gitignore](#)

edited Nov 14 at 14:30



Peter Mortensen

24.1k 19 89 118

answered Jun 27 '18 at 11:22



fvolodimir

151 2 2

To ignore an entire directory in Git, the easiest way is to include a `.gitignore` file within the target directory which simply contains `""`.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
/root/  
  .gitignore  
  /dirA/  
    someFile1.txt  
    someFile2.txt  
  /dirB/  
    .gitignore  
    someFile3.txt  
    someFile4.txt
```

## Goal

- ignore the contents of /dirB/

## Top Level .gitignore (/root/.gitignore)

- This is where your standard gitignore info goes

## Ignored Directory .gitignore (/root/dirB.gitignore)

- This file just reads as '\*' and the directory is ignored completely, itself and all files!

And it's that simple :)

edited Nov 14 at 14:29



Peter Mortensen

24.1k 19 89 118

answered Jun 15 '16 at 21:51



J-Dizzle

2,896 3 27 40

In Windows there's an extra catch with slashes. Excluding a single directory in **.gitignore** with

16

dir\_to\_exclude/

will possibly work, but excluding all directories with

/

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

To exclude all directories, better use:

```
**/
```

Two consecutive asterisks signify directory contents.

edited Nov 14 at 14:26



Peter Mortensen

24.1k 19 89 118

answered Jan 13 '14 at 13:49



wortwart

1,788 17 21



1



I had similar issues. I work on a Windows tool chain with a shared repository with Linux guys, and they happily create files with the same (except for case) names in a given folder.

The effect is that I can clone the repository and immediately have dozens of 'modified' files that, if I checked in, would create havoc.

I have Windows set to case sensitive and Git to not ignore case, but it still fails (in the Win32 API calls apparently).

If I gitignore the files then I have to remember to not track the .gitignore file.

But I found a good answer here:

<http://archive.robwilkerson.org/2010/03/02/git-tip-ignore-changes-to-tracked-files/index.html>

edited Nov 14 at 14:22



Peter Mortensen

24.1k 19 89 118

answered Aug 6 '12 at 15:01



Chris Aaaaa

11 1

The link appears to be broken (it first redirects to the 'https' version): "Secure Connection Failed. An error occurred during a connection to archive.robwilkerson.org. PR\_END\_OF\_FILE\_ERROR" – Peter Mortensen Nov 14 at 14:23



10

If you want to maintain a folder and not the files inside it, just put a ".gitignore" file in the folder with "\*" as the content. This file will make Git ignore all content from the repository. But .gitignore will be included in your repository.

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

If you add an empty folder, you receive this message (.gitignore is a hidden file)

```
The following paths are ignored by one of your .gitignore files:
path/to/folder/.gitignore
Use -f if you really want to add them.
fatal: no files added
```

So, use "-f" to force add:

```
$ git add path/to/folder/.gitignore -f
```

edited Nov 14 at 14:15



Peter Mortensen

24.1k 19 89 118

answered Jun 29 '12 at 10:29



sensorario

12.1k 15 60 115

I assume the problem is that your working tree is like:

5

```
a-cache/foo
a-cache/index.html
b-cache/bar
b-cache/foo
b-cache/index.html
.gitignore
```

... with the .gitignore you describe. This will give you `git status` output like:

```
$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#   .gitignore
#   a-cache/
#   b-cache/
```

if the `index.html` files have not yet been added to the repository. (Git sees that there are unignored files in the cache directories, but it

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
git add *cache/index.html
git commit -m "Adding index.html files to the cache directories"
```

... and your `git status` will then look like:

```
$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#   .gitignore
nothing added to commit but untracked files present (use "git add" to track)
```

(Obviously you do want to commit `.gitignore` as well. I was just being lazy with this test case.)

edited Nov 14 at 14:13



Peter Mortensen

24.1k 19 89 118

answered Jun 29 '12 at 10:11



Mark Longair

325k 63 368 308

- 1 You probably *do* want to commit `.gitignore` especially as you're likely to want to track changes to it, and so is your team (if you're working with one). See [stackoverflow.com/a/767213/123033](https://stackoverflow.com/a/767213/123033) – Dave Everitt Jan 15 '13 at 14:24
- 2 @Dave Everitt: That's exactly why i said "Obviously you do want to commit `.gitignore` as well". – Mark Longair Jan 15 '13 at 15:16
- 3 Oops, misread it! Consider your point reiterated. – Dave Everitt Jan 16 '13 at 17:05

It seems that for ignoring files and directories there are two main ways:

98

## 1. `.gitignore`

- Placing `.gitignore` file into the root of your repository besides the `.git` folder (in Windows, make sure you [see the true file extension](#) and then make `.gitignore.` (with the point at the end to make an empty file extension))
- Making the global configuration `~/.gitignore_global` and running `git config --global core.excludesfile ~/.gitignore_global` to add this to your Git configuration

Note: files tracked before can be untracked by running `git rm --cached filename`

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

To make exceptions in the list of ignored files, see [this question](#).

edited Nov 14 at 14:01



Peter Mortensen

24.1k 19 89 118

answered May 1 '12 at 10:30



Vairis

1,308 11 23

- 
- 32 good job pointing out the `git rm --cached <filename> .` Absolutely critical for repos that existed BEFORE you created the `.gitignore` – [MaurerPower](#) Jun 3 '12 at 1:06
- 
- 2 `git rm --cached <filename>` fixed the problem I was having with `.gitignore` :) – [JeremyFelix](#) Mar 17 '13 at 23:27
- 
- 1 Yeah `git rm --cached filename` is absolutely crucial. It was driving me crazy as git was still staging files I clearly stated to be ignored (created `gitignore` after initial commit). Thank you very much! – [Potaito](#) Oct 8 '13 at 16:55
- 
- 1 Solved my problem. Git was tracking vendor folder in my laravel repo even though I had added vendor folder in `gitignore`. Thanks alot. – [Jayant](#) Aug 31 '15 at 5:50
- 



I've had some problems getting Git to pick up the `.gitignore` file on Windows. The `$GIT_DIR/info/exclude` file always seems to work though.

7



The downside of this approach, however, is that the files in the `$GIT_DIR` directory are not included in the check-in, and therefore not shared.

p.s. `$GIT_DIR` is usually the hidden folder named `.git`

edited Nov 14 at 13:57



Peter Mortensen

24.1k 19 89 118

answered Feb 15 '12 at 19:43



Jason

545 6 12

---

Yes, git on windows is really finicky about `.gitignore` - the local exclude file does what I need though. Thanks! – [andersop](#) Jul 26 '12 at 0:39

---



By default, Windows Explorer will display `.gitignore` when in fact the file name is `.gitignore.txt` .

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



## Non command line solution:

You can rename a file to `".gitignore."`, and it will create `".gitignore"`

edited Nov 14 at 13:54



Peter Mortensen

24.1k 19 89 118

answered Jul 22 '10 at 11:23



brainwavedave

1,977 1 11 3

- 
- 45 That works? I've always just told people to open notepad and in the Save As dialog type the filename surrounded by doublequotes, so for example `".gitignore"` and it saves it without automatically adding an extension. – [Arrowmaster](#) Feb 10 '11 at 17:44
- 
- 9 Neat. Including a trailing period does work. Explorer strips off the last period, resulting in a file named `".gitignore"`. I think the quotes method is cleaner though and less likely to create surprises. – [Triynko](#) Sep 8 '11 at 22:29
- 
- 5 Wow, weird as heck but that definitely worked. Thanks! – [Nick Knowlson](#) Feb 17 '13 at 21:51
- 
- 3 Or, in the Save As dialog, change the file type to "All Files (.\*)" - then Windows will not append any extension. – [OsakaWebbie](#) Mar 26 '14 at 2:20
- 
- 5 or in git bash just type `"touch .gitignore"` in your folder – [Rayjax](#) May 27 '14 at 12:25
- 

To instruct Git to ignore certain files or folders, you have to create `.gitignore` file.

41

But in Windows Explorer you have to provide a name for the file. You just cannot create file with just an extension. The trick is that create a empty text file and go to command prompt and change the name of the file to `.gitignore` :

```
ren "New Text Document.txt" .gitignore
```

Now open the file with your favorite text editor and add the file/folder names you wish you ignore. You can also use wildcards like this:  
`*.txt` .

edited Nov 14 at 13:51



Peter Mortensen

24.1k 19 89 118

answered Mar 25 '10 at 18:14




Mahes

2,670 1 26 34

- 
- 13 Or just `echo dir |> ignore /> .gitignore` – [Oliver](#) Jun 6 '10 at 17:16

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

Or just call the file `.gitignore`. when the file extensions are not hidden in your Windows explorer – [Vairis](#) May 6 '15 at 11:43 



43



I had some issues creating a file in [Windows Explorer](#) with a `.` at the beginning.

A workaround was to go into the commandshell and create a new file using "edit".

edited Nov 14 at 13:50



[Peter Mortensen](#)

24.1k 19 89 118

answered Dec 5 '08 at 13:50



[sf.](#)

17.3k 10 44 56

1 or just add an extra `.` at the end so explorer stops thinking `.gitignore` is the extension. Then on entry that trailing dot with no extension just gets eaten and you are left with `.gitignore` TL;DR: try to name it `.gitignore.` => you end up with `.gitignore` – [leerssej](#) Feb 25 '18 at 20:13



8



When **everything else fails** try editing the file

`/.git/info/exclude`

and adding the directories you want to the end of the file, like this:

```
# git ls-files --others --exclude-from=.git/info/exclude
# Lines that start with '#' are comments.
# For a project mostly in C, the following would be a good set of
# exclude patterns (uncomment them if you want to use them):
# *.o[ad]
# *~
assets/
compiled/
```

I added the folders "assets" and "compiled" to the list of files and directories to ignore.

answered Apr 4 '18 at 21:38



[Xedret](#)

1,170 9 19

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

I tried all the others with my current configuration and this was my ultimate definitive answer. – [Xedret](#) Nov 14 '18 at 19:59



Just in case you need to exclude sub folders you can use the `**` wildcard to exclude any level of sub directory.

12

`**/build/output/Debug/`



answered Feb 17 '18 at 0:40



[Eat at Joes](#)

3,810 1 28 35



Also in your `\.git\info` projects directory there is an exclude file that is effectively the same thing as `.gitignore` (I think). You can add files and directories to ignore in that.

14



edited Apr 8 '16 at 17:24



[1ac0](#)

2,485 2 27 45

answered Feb 10 '11 at 10:12



[Si3](#)

141 1 2

I like this approach, I just can't get along with the path in there, I want to ignore a folder in the main repo, how to? – [Shimmy](#) May 18 '17 at 6:50

But aren't you supposed to leave files in `.git` alone (manipulate the content through official means)? – [Peter Mortensen](#) Nov 14 at 13:55



**Highly active question.** Earn 10 reputation in order to answer this question. The reputation requirement helps protect this question from spam and non-answer activity.