# Git config file location

<https://www.theserverside.com/blog/Coffee-Talk-Java-News-Stories-and-Opinions/Where-system-global-and-local-Windows-Git-config-files-are-saved>

Table

Description automatically generated

# Autocrlf

<https://stackoverflow.com/questions/10418975/how-to-change-line-ending-settings>

## How to change line-ending settings

Graphical user interface, text, application

Description automatically generated

Is there a file or menu that will let me change the settings on how to deal with line endings?

I read there are 3 options:

1. **Checkout Windows-style, commit Unix-style**

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true")

1. **Checkout as-is, commit Unix-style**

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects this is the recommended setting on Unix ("core.autocrlf" is set to "input").

1. **Checkout as-is, commit as-is**

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false")

## Details

<http://git-scm.com/book/en/v2/Customizing-Git-Git-Configuration#_formatting_and_whitespace>

#### core.autocrlf

If you’re programming on Windows and working with people who are not (or vice-versa), you’ll probably run into line-ending issues at some point. This is because Windows uses both a carriage-return character and a linefeed character for newlines in its files, whereas macOS and Linux systems use only the linefeed character. This is a subtle but incredibly annoying fact of cross-platform work; many editors on Windows silently replace existing LF-style line endings with CRLF, or insert both line-ending characters when the user hits the enter key.

Git can handle this by auto-converting CRLF line endings into LF when you add a file to the index, and vice versa when it checks out code onto your filesystem. You can turn on this functionality with the core.autocrlf setting. If you’re on a Windows machine, set it to true — this converts LF endings into CRLF when you check out code:

$ git config --global core.autocrlf true

If you’re on a Linux or macOS system that uses LF line endings, then you don’t want Git to automatically convert them when you check out files; however, if a file with CRLF endings accidentally gets introduced, then you may want Git to fix it. You can tell Git to convert CRLF to LF on commit but not the other way around by setting core.autocrlf to input:

$ git config --global core.autocrlf input

This setup should leave you with CRLF endings in Windows checkouts, but LF endings on macOS and Linux systems and in the repository.

If you’re a Windows programmer doing a Windows-only project, then you can turn off this functionality, recording the carriage returns in the repository by setting the config value to false:

$ git config --global core.autocrlf false