

```

● carole@mbp-von-carole git-cours % mkdir learn_git
● carole@mbp-von-carole git-cours % cd learn_git
● carole@mbp-von-carole learn_git % touch third.txt
● carole@mbp-von-carole learn_git % git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/carole/Dropbox/Mac (2)/Desktop/GoMyCode/git-cours/learn_git/.git/
● carole@mbp-von-carole learn_git % git add .
⊗ carole@mbp-von-carole learn_git % git commit "adding third.txt"
error: pathspec 'adding third.txt' did not match any file(s) known to git
● carole@mbp-von-carole learn_git % git commit -m "adding third.txt"
[master (root-commit) dfabf73] adding third.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 third.txt
● carole@mbp-von-carole learn_git % git log
commit dfabf735960707a1b4542d527cb6d261ddcbadeb (HEAD -> master)
Author: cocoPastell <schmutz.carole@gmail.com>
Date:   Wed Mar 8 13:15:48 2023 +0000

    adding third.txt
● carole@mbp-von-carole learn_git % touch fourth.txt
● carole@mbp-von-carole learn_git % git add .
● carole@mbp-von-carole learn_git % git commit -m "adding fourth.txt"
[master c90a97c] adding fourth.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 fourth.txt
● carole@mbp-von-carole learn_git % git rm third.txt
rm 'third.txt'
● carole@mbp-von-carole learn_git % git add .
● carole@mbp-von-carole learn_git % git commit -m "removing third.txt"
[master 0aa43f6] removing third.txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 third.txt
● carole@mbp-von-carole learn_git % git log

```

```
● carole@mbp-von-carole learn_git % git log
commit 0aa43f6277adfd65e4c088db3c8344770c1fb275 (HEAD -> master)
Author: cocoPastell <schmutz.carole@gmail.com>
Date: Wed Mar 8 13:19:10 2023 +0000
```

removing third.txt

```
commit c90a97c0882216a7ec6a169795e61a44287a02ac
Author: cocoPastell <schmutz.carole@gmail.com>
Date: Wed Mar 8 13:17:40 2023 +0000
```

adding fourth.txt

```
commit dfabf735960707a1b4542d527cb6d261ddcbadeb
Author: cocoPastell <schmutz.carole@gmail.com>
Date: Wed Mar 8 13:15:48 2023 +0000
```

adding third.txt

- ⊗ carole@mbp-von-carole learn_git % git config --global core.pager=cat
error: invalid key: core.pager=cat
- carole@mbp-von-carole learn_git % git config --global core.pager="cat
dquote>
- ⊗ carole@mbp-von-carole learn_git % git config --global core.pager="cat"
error: invalid key: core.pager=cat
- carole@mbp-von-carole learn_git % git config --global core.pager "cat"
- ⊗ carole@mbp-von-carole learn_git % git config --global
usage: git config [<options>]

Config file location

--global	use global config file
--system	use system config file
--local	use repository config file
--worktree	use per-worktree config file
-f, --file <file>	use given config file
--blob <blob-id>	read config from given blob object

Action	
--get	get value: name [value-pattern]
--get-all	get all values: key [value-pattern]
--get-regexp	get values for regexp: name-regex [value-pattern]
--get-urlmatch	get value specific for the URL: section[.var] URL
--replace-all	replace all matching variables: name value [value-pattern]
--add	add a new variable: name value
--unset	remove a variable: name [value-pattern]
--unset-all	remove all matches: name [value-pattern]
--rename-section	rename section: old-name new-name
--remove-section	remove a section: name
-l, --list	list all
--fixed-value	use string equality when comparing values to 'value-pattern'
-e, --edit	open an editor
--get-color	find the color configured: slot [default]
--get-colorbool	find the color setting: slot [stdout-is-tty]

Type	
-t, --type <type>	value is given this type
--bool	value is "true" or "false"
--int	value is decimal number
--bool-or-int	value is --bool or --int
--bool-or-str	value is --bool or string
--path	value is a path (file or directory name)
--expiry-date	value is an expiry date

Other	
-z, --null	terminate values with NUL byte
--name-only	show variable names only
--includes	respect include directives on lookup
--show-origin	show origin of config (file, standard input, blob, command line)
--show-scope	show scope of config (worktree, local, global, system, command)
--default <value>	with --get, use default value when missing entry

```

carole@mbp-von-carole learn_git % git config --global --list --show-origin
file:/Users/carole/.gitconfig user.name=cocoPastell
file:/Users/carole/.gitconfig user.email=schmutz.carole@gmail.com
file:/Users/carole/.gitconfig core.pager=cat

```

```

○ carole@mbp-von-carole learn_git % █

```