vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git log

commit 75b6cd1fc86f1d5dd404e62bea8d6839dee393d5 (HEAD -> main)

Author: Sarra <Khmilasarra@gmail.com>

Date: Tue Jun 29 11:17:15 2021 +0200

first commit

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ ls

README.md fourth.txt.txt third.txt.txt work.docx

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git status

On branch main

Untracked files:

(use "git add <file>..." to include in what will be committed)

fourth.txt.txt

third.txt.txt

work.docx

nothing added to commit but untracked files present (use "git add" to track)

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git add <third.txt>

bash: syntax error near unexpected token `newline'

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git commit -m "adding third.txt"

On branch main

Untracked files:

(use "git add <file>..." to include in what will be committed)

fourth.txt.txt

third.txt.txt

work.docx

nothing added to commit but untracked files present (use "git add" to track)

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git add

Nothing specified, nothing added.

hint: Maybe you wanted to say 'git add .'?

hint: Turn this message off by running

hint: "git config advice.addEmptyPathspec false"

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git commit "adding fourth.txt"

error: pathspec 'adding fourth.txt' did not match any file(s) known to git

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ git push

fatal: The current branch main has no upstream branch.

To push the current branch and set the remote as upstream, use

git push --set-upstream origin main

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$ 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-patt

ern]

--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-pa

ttern'

-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 <> 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, com

mand line)

--show-scope show scope of config (worktree, local, global, system,

command)

--default <value> with --get, use default value when missing entry

vega@DESKTOP-0I3NVU9 MINGW64 ~/Desktop/learn\_git\_again (main)

$