



git

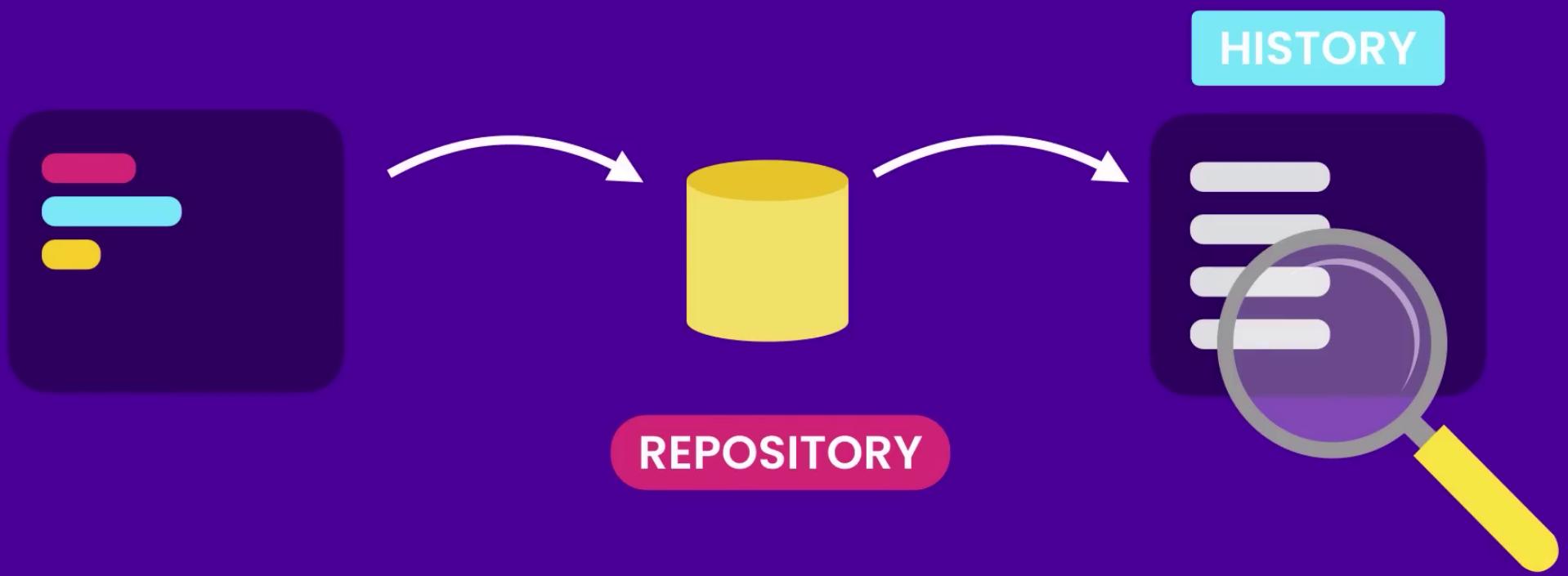


HISTORY



REPOSITORY





## John's Code



v1.0.0



v1.0.1



v1.0.2

## Amy's Code



v1.0.0



v1.0.1



v1.0.2

**John's Code**

**Amy's Code**



**v1.0.2**

## **Version Control System**

**Track History**

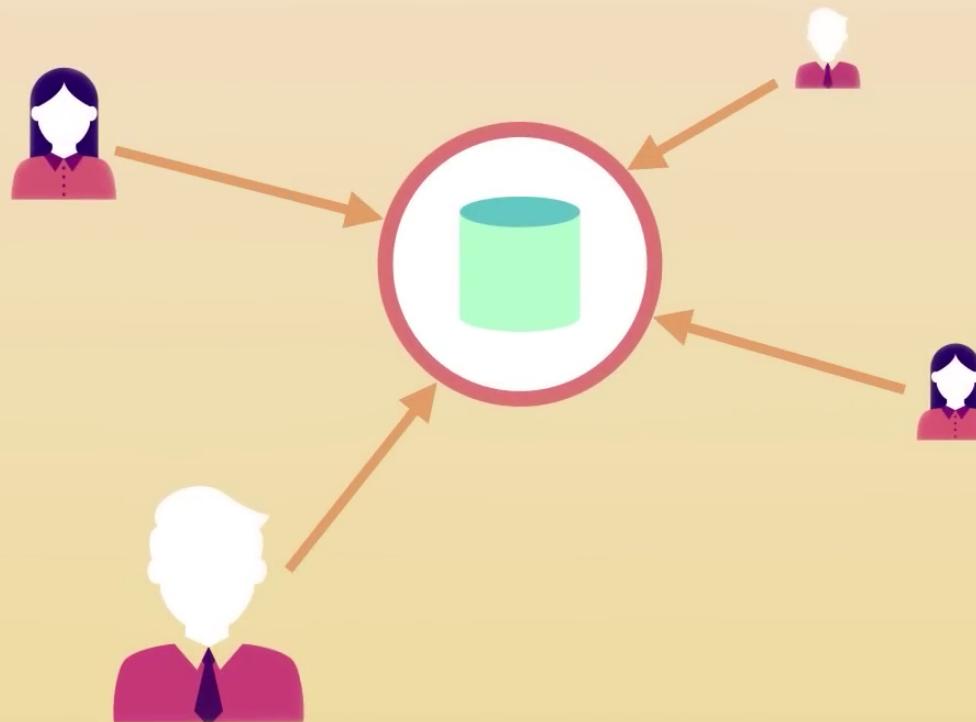
**Work Together**

## **Version Control System**

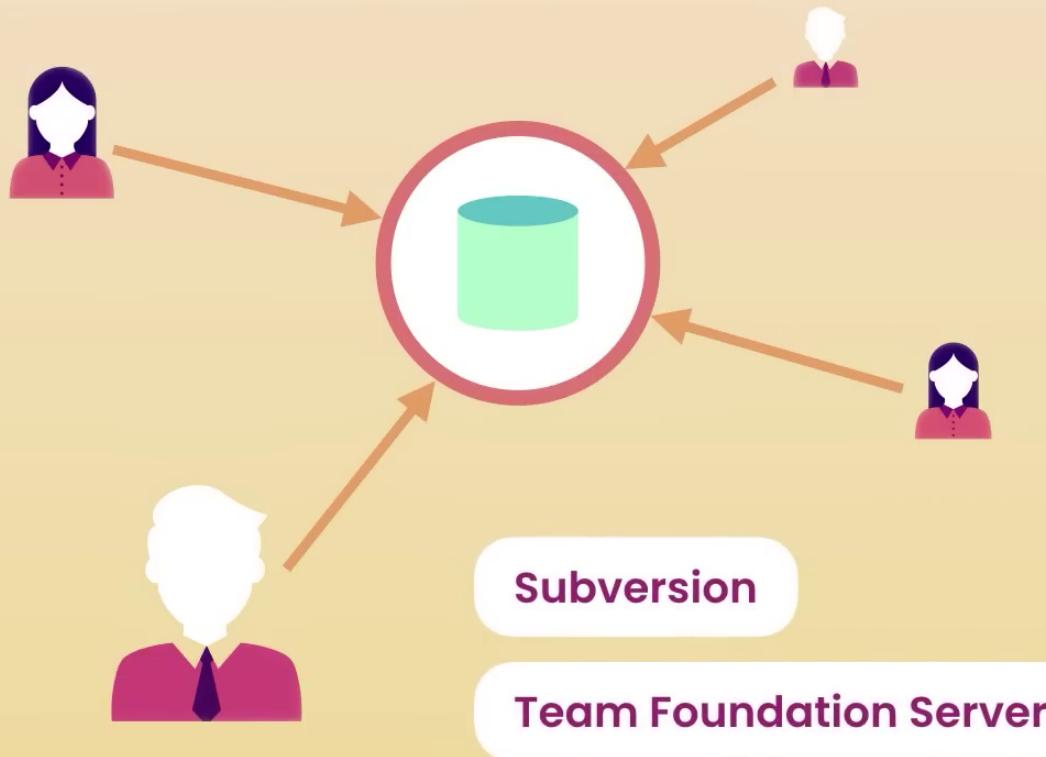
**Centralized**

**Distributed**

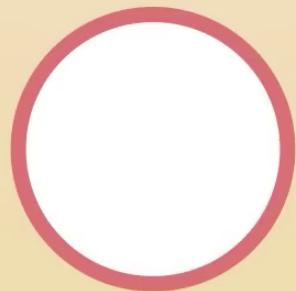
**CENTRALIZED**



**CENTRALIZED**



**CENTRALIZED**



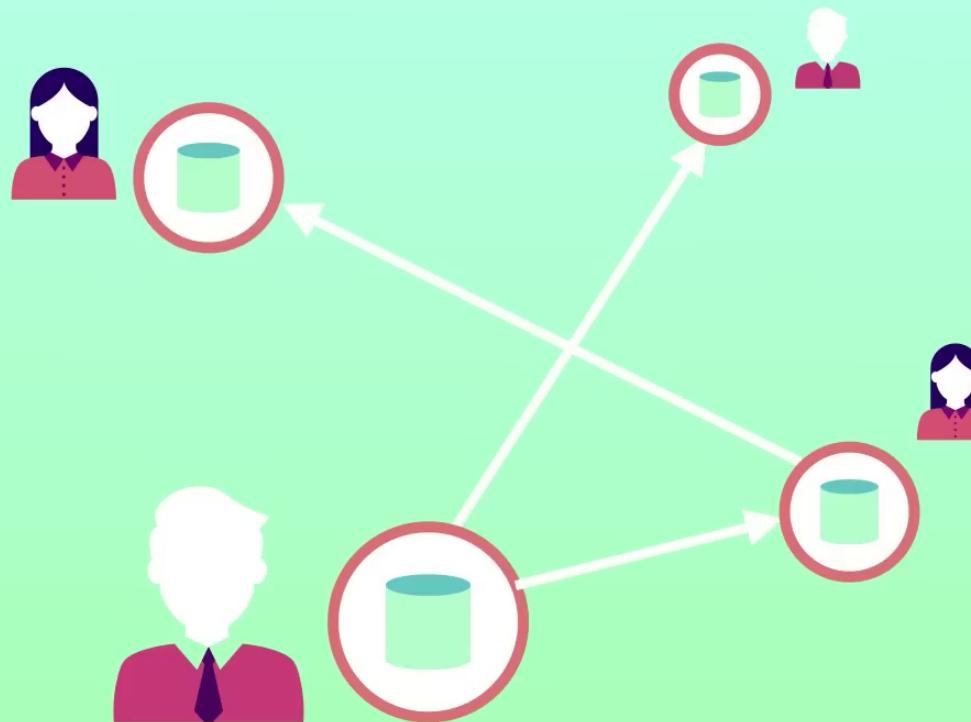
**Subversion**

**Team Foundation Server**

**DISTRIBUTED**



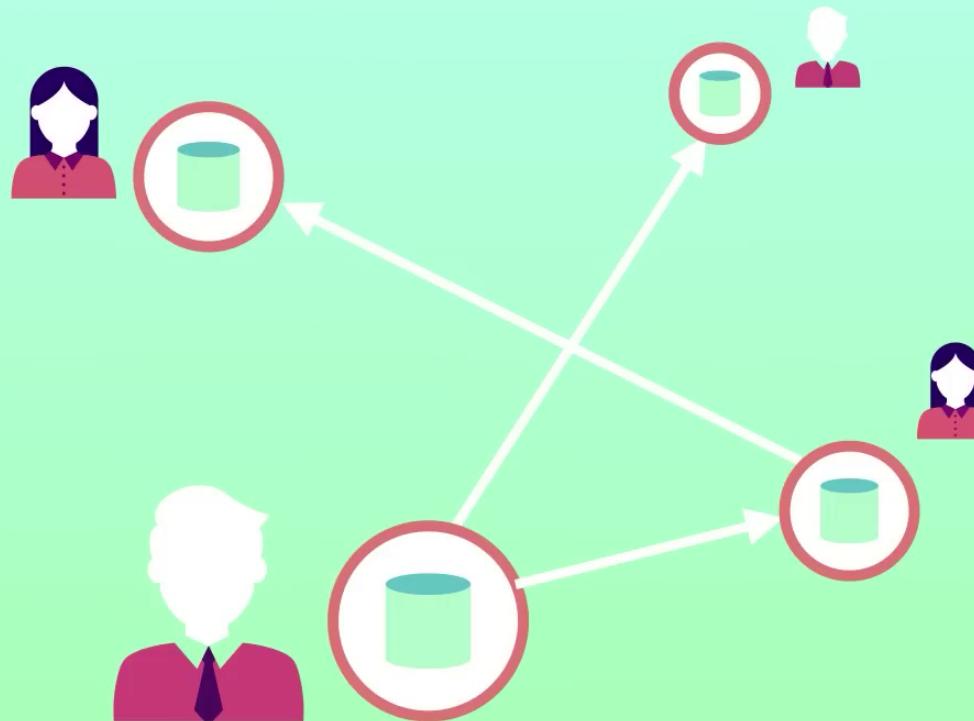
**DISTRIBUTED**



DISTRIBUTED

Git

Mercurial



## WHY GIT

- Free
- Open Source
- Super Fast
- Scalable

## WHY GIT

- Free
- Open Source
- Super Fast
- Scalable
- Cheap Branching/Merging

```
3. git log --oneline --graph (less)  
* 260f5ac (HEAD -> master, origin/master, origin/HEAD) Update the link to the co  
urse.  
* 403b20f Update readme file.  
* 2528bc6 Define a theme.  
* c14d1a1 Align the NavBar items.  
* 26a9bba Add icons.  
* 6e51c80 Export and re-import modules.  
* 3a8c4f1 Create the core module.  
* a26a343 Create the shopping module.  
* f7692a8 Create the admin module.  
* e86d335 Move admin artefacts.  
* 953a4c9 Create the shared module.  
* 85ced65 Move shared module artefacts.  
* d831dc5 Add baseUrl and paths to tsconfig.  
* 97e4096 Fixing a bug: ensure the user cannot submit the shipping form without  
providing values.  
* 0be3d4c Display the orders.  
* 05b3834 Refactor: Extract ShippingFormComponent.  
* 62584bf Add an order summary widget  
* a8f94cc Clearing the shopping cart.  
* 691e46e Redirect the user upon submitting an order.  
* 98692d6 Refactoring: Extract an Order Model  
* 20d53fc Associate the order with the current user.  
* 858366 Some styling in Final
```

**Track History**

**Work Together**

That's what you're going to learn in this course

## USING GIT

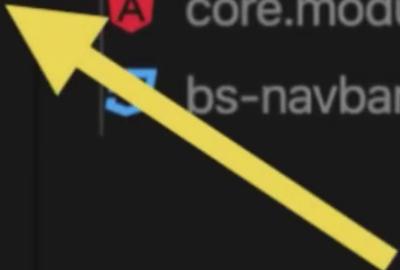
### The command line

## USING GIT

- The command line
- Code editors & IDEs

Message (⌘Enter to commit on...)

- Changes 4
- README.md M
  - main.ts src ↗ ↘ + M
  - A core.module.ts src/app/core M
  - bs-navbar.component.css... M





# GitLens — Git supercharged

eamodio.gitlens

Eric Amodio

5,856,529

★★★★★

Repository

License

Supercharge the Git capabilities built into Visual Studio Code — Visualize code auth

Install

*This extension is recommended because of the current workspace configuration* [Ignore](#) [Report](#)

[Details](#) [Feature Contributions](#) [Changelog](#)

VS Marketplace v10.2.2

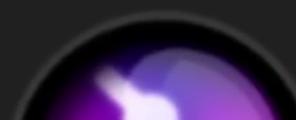
downloads 39.05M

rating 4.86/5 (378)



Live Share enabled

vscode-dev



GitLens

## USING GIT

- The command line
- Code editors & IDEs
- Graphical user interfaces



Search entire site...

## About

## Documentation

## Downloads

### GUI Clients

Logos

## Community

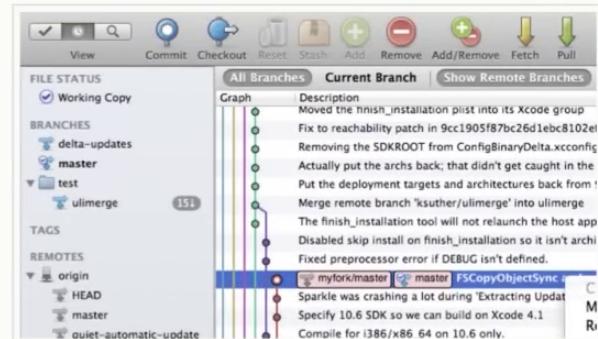
The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to read [online for free](#). Dead tree versions are available on [Amazon.com](#).

# GUI Clients

Git comes with built-in GUI tools for committing ([git-gui](#)) and browsing ([gitk](#)), but there are several third-party tools for users looking for platform-specific experience.

If you want to add another GUI tool to this list, just [follow the instructions](#).

All    Windows    Mac    Linux    Android    iOS

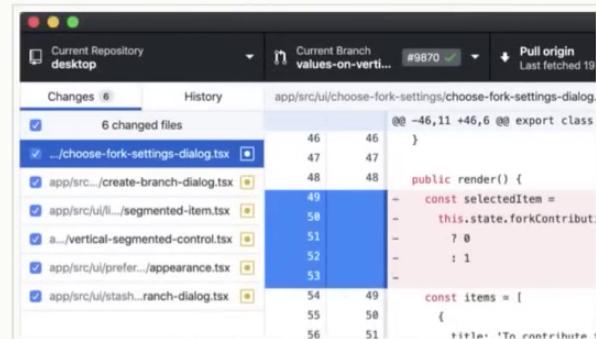


### SourceTree

**Platforms:** Mac, Windows

**Price:** Free

**License:** Proprietary



### GitHub Desktop

**Platforms:** Mac, Windows

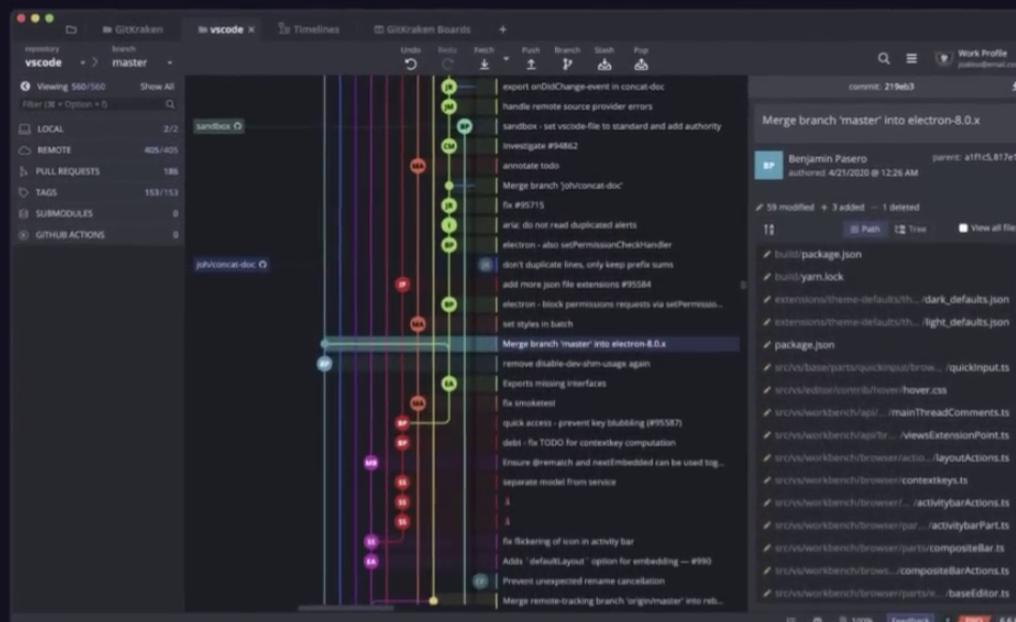
**Price:** Free

**License:** MIT



# GitKraken Git GUI

Legendary Git client for Windows, Mac & Linux. Free for open source



[Download free](#)

## Simplicity and power in a beautiful Git GUI

[Download for Mac OS X](#)

Also available for Windows

The screenshot shows the Sourcetree application window for the repository "sourcetree-website (Git)". The left sidebar includes sections for WORKSPACE (Commit, Pull, Push), BRANCHES, BOOKMARKS, TAGS, REMOTES, SHELVED, and SUBREPOSITORIES. The main area features a "Graph" view on the left showing a complex branching structure with colored lines (purple, yellow, green) and a detailed commit list on the right. The commit list includes columns for Commit ID, Author, Description, and Date. Notable commits include "Removing old Atlassian logo" by Rahul Chhabria on March 3, 2016, and "Merge in update-google-verification" by Rahul Chhabria on February 18, 2016.

Commit	Author	Description	Date
b7358c7	Rahul Chhab...	Removed master origin/master origin/HEAD Removing old Atlassian logo	Mar 3, 2016, 11:...
bdb8bef	Rahul Chhab...	Merged in update-google-verification (pull request #14)	Feb 18, 2016, 1:3...
dfe975d	Tyler Tadej...	origin/update-google-verification Update google verificati...	Feb 11, 2016, 2:2...
3bc3290	Tyler Tadej...	Replace outdated Atlassian logo in footer with base-64 en...	Feb 11, 2016, 2:1...
dba47f9	Tyler Tadej...	Add gitignore	Feb 11, 2016, 1:3...
ff67b45	Mike Minns...	Updated Mac min-spec to 10.10	Feb 15, 2016, 11:...
72d32a8	Michael Min...	Merged in hero_images (pull request #13)	Feb 15, 2016, 10:...
246c4ff	Joel Unger...	origin/hero_images hero_images Used Tinypng to c...	Feb 11, 2016, 3:3...
9d9438c	Joel Unger...	Replacing hero images with new version of SourceTree	Feb 9, 2016, 2:59...
ce75b63	Michael Min...	Merged in bug/date-https (pull request #12)	Feb 15, 2016, 10:...
85367bb	Patrick Tho...	origin/bug/date-https fixed date and https errors	Jan 7, 2016, 12:2...
4f9b557	Joel Unger...	New Favicon	Feb 8, 2016, 3:55...
384e6d5	Rahul Chhab...	origin/search-console-access search console google ver...	Feb 3, 2016, 2:09...
6fa47a9	Mike Minns...	updated to move supported version to OSX 10.9+	Dec 15, 2015, 2:0...
8dd87bb	Mike Minns...	remove extra , when a line is skipped due to empty server	Nov 23, 2015, 2:2...
faa195e	Mike Minns...	Skip records with empty server/user id as gas rejects them	Nov 23, 2015, 2:1...
0cdfe96	Mike Minns...	corrected paths after merge	Nov 23, 2015, 2:0...
051ab1b	Mike Minns...	corrected column counting	Nov 23, 2015, 1:5...
a723bc2	Mike Minns...	Merge branch 'au2gex'	Nov 23, 2015, 1:5...
65fd580	Mike Minns...	deal with invalid instanceids	Nov 23, 2015, 1:5...
500a892	Michael Min...	Merged in au2gex (pull request #11)	Nov 23, 2015, 1:0...

## A free Git client for Windows and Mac

Sourcetree simplifies how you interact with your Git repositories so you can focus on coding. Visualize and manage your repositories through Sourcetree's simple Git GUI.

## WHY COMMAND LINE

- GUI tools have limitations
- GUI tools are not always available

## TERMINAL



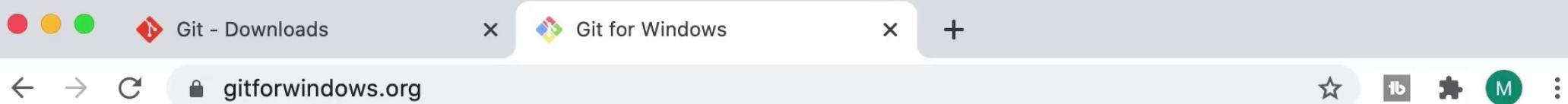
macOS

Press **Command + Space** and type **terminal**



Windows

Click the search icon and type **cmd**



of tools that bring the full feature set of the [Git SCM](#) to Windows while providing appropriate user interfaces for experienced Git users and novices alike.

## Git BASH

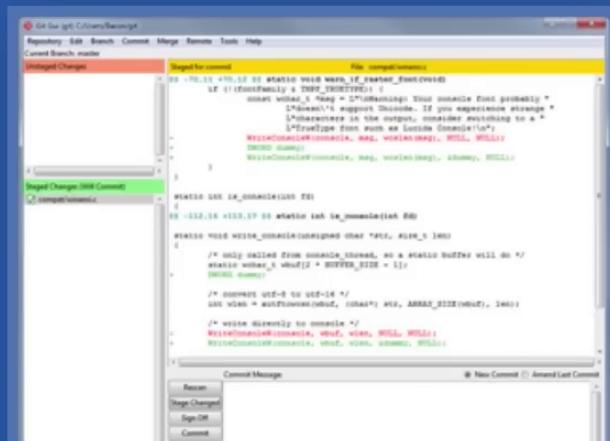
Git for Windows provides a BASH emulation used to run Git from the command line. \*NIX users should feel right at home, as the BASH emulation behaves just like the "git" command in LINUX and UNIX environments.

A screenshot of a Windows command prompt window titled 'Git BASH'. It shows a session where the user cloned a GitHub repository and checked its status. The terminal window has a blue border and a white background with black text.

Git BASH

## Git GUI

As Windows users commonly expect graphical user interfaces, Git for Windows also provides the Git GUI, a powerful alternative to Git BASH, offering a graphical version of just about every Git command line function, as



## SETTINGS

- Name
- Email
- Default Editor
- Line Ending

**SYSTEM**

All users

**GLOBAL**

All repositories of the current user

**LOCAL**

The current repository

```
home ~
> ✓ git config --global
```



```
git config --global user.name ""
```

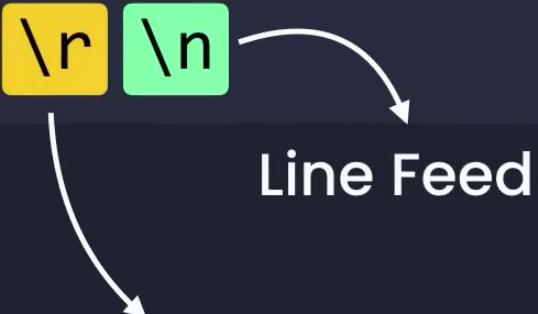
```
git config --global user.name "Mosh Hamedani"
```

```
git config --global user.email programmingwithmosh@gmail.com
```

```
git config --global core.editor "code --wait"
```

## WINDOWS

abc **\r** **\n** Line Feed  
Carriage Return



WINDOWS

abc **\r** **\n** Line Feed  
Carriage Return

macOS / Linux

abc **\n**

`core.autocrlf`



`core.autocrlf`



Windows



macOS

**core.autocrlf**

CR LF



Windows



macOS

**core.autocrlf**

CR LF



Windows

LF



macOS



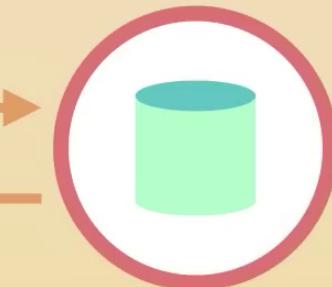
## core.autocrlf

CR LF



Windows

LF



macOS



`core.autocrlf`

CR LF



Windows

LF



macOS

true

`core.autocrlf`

CR LF



Windows

LF



LF



macOS

true

`core.autocrlf`

CR LF



Windows

LF



LF



macOS

true

input



```
git config --global core.autocrlf true
```

```
git config --global core.autocrlf input
```

```
🏠 ~ ➜ ✓ git config --help
```

**NAME**

**git-config** - Get and set repository or global options

**SYNOPSIS**

```
git config [<file-option>] [--type=<type>] [--show-origin] [-z  
|--null] name [value [value_regex]]  
git config [<file-option>] [--type=<type>] --add name value  
git config [<file-option>] [--type=<type>] --replace-all name  
value [value_regex]  
git config [<file-option>] [--type=<type>] [--show-origin] [-z  
|--null] --get name [value_regex]  
git config [<file-option>] [--type=<type>] [--show-origin] [-z  
|--null] --get-all name [value_regex]  
git config [<file-option>] [--type=<type>] [--show-origin] [-z  
|--null] [--name-only] --get-regexp name_regex [value_regex]  
git config [<file-option>] [--type=<type>] [-z|--null] --get-u
```

# Initializing a Repository



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects ➔  
❯ ✓ ➔ mkdir Moon
```

```
📁 ~/Projects ➔  
❯ ✓ ➔ cd Moon
```

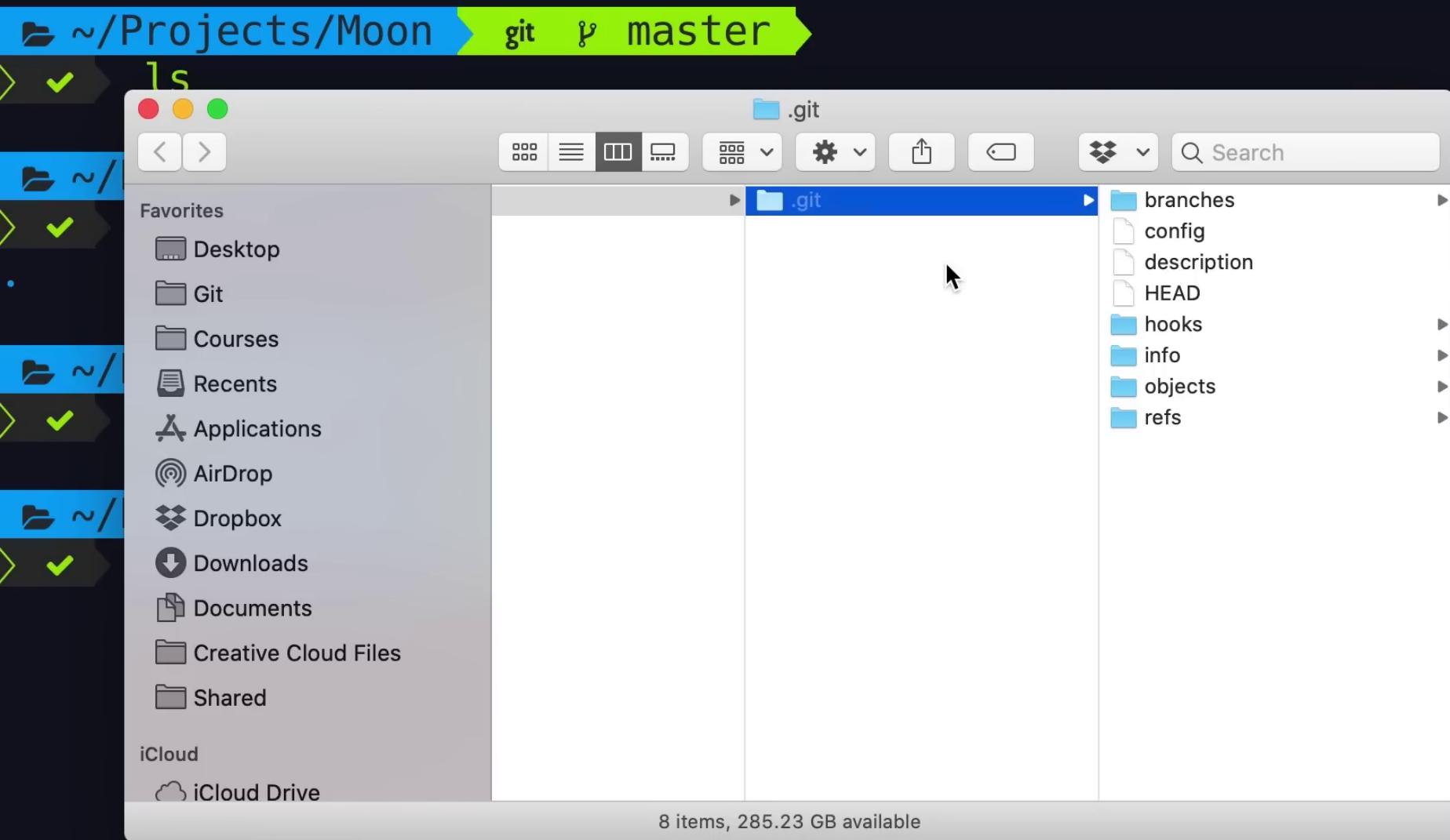
```
📁 ~/Projects/Moon ➔  
❯ ✓ ➔ git init  
Initialized empty Git repository in /Users/moshfeghamedani/Projects/  
Moon/.git/
```

```
📁 ~/Projects/Moon ➔ git ↵ master ➔  
❯ ✓ ➔
```

```
📁 ~/Projects/Moon ➜ git ⌂ master  
❯ ✓ ➔ ls
```

```
📁 ~/Projects/Moon ➜ git ⌂ master  
❯ ✓ ➔ ls -a  
. .. .git
```

```
📁 ~/Projects/Moon ➜ git ⌂ master  
❯ ✓ ➔ ⌂
```

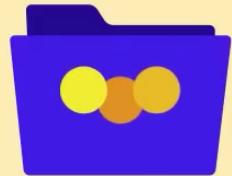


# Git Workflow

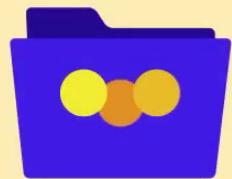


**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

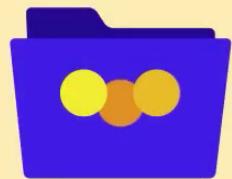
## WORKFLOW



# WORKFLOW

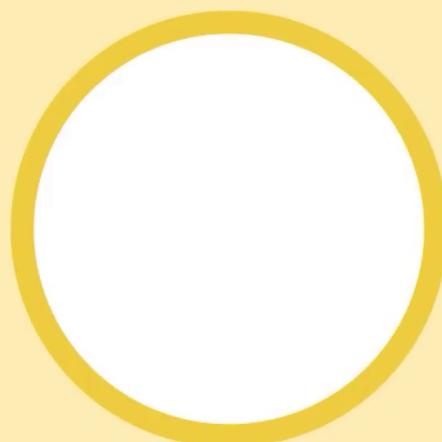
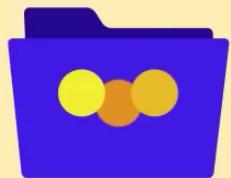


## WORKFLOW



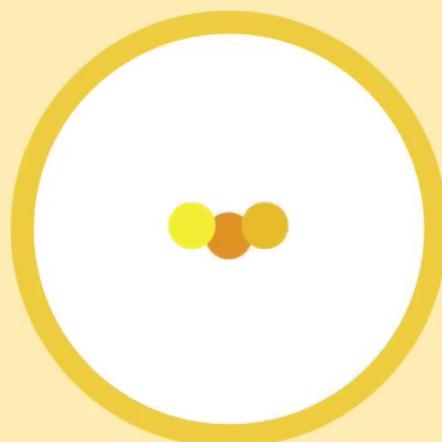
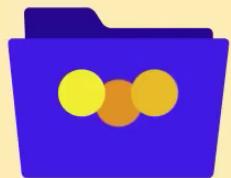
# WORKFLOW

Staging Area



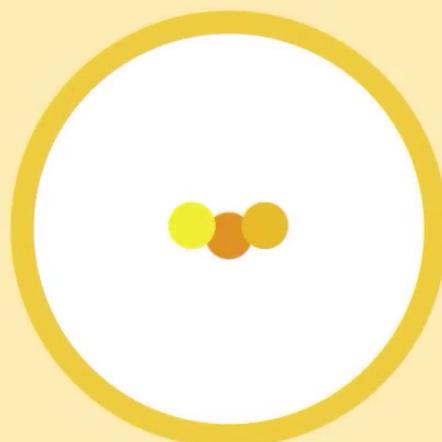
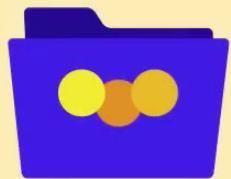
# WORKFLOW

Staging Area



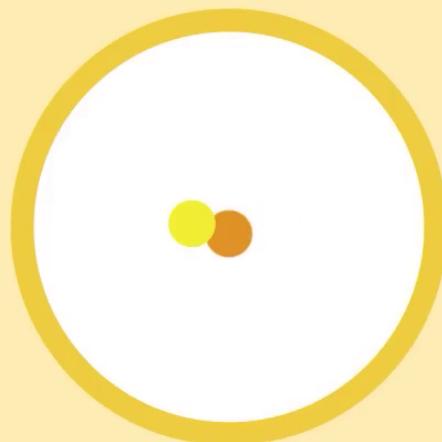
# WORKFLOW

Staging Area



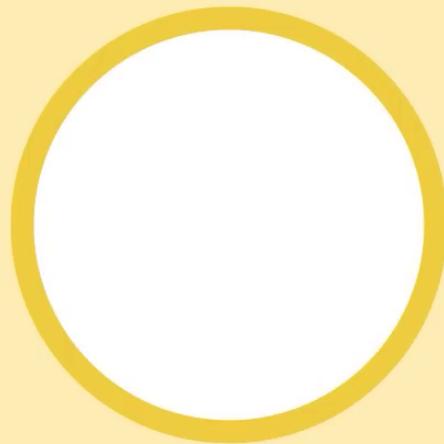
# WORKFLOW

Staging Area

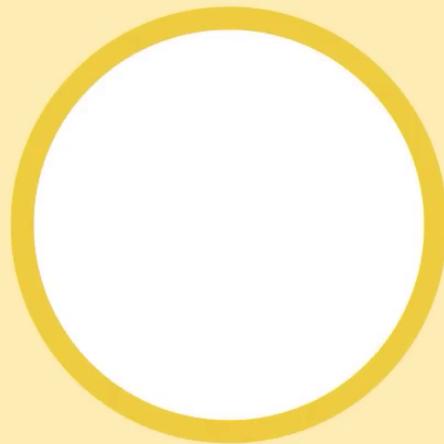


**REAL EXAMPLE**

## WORKFLOW



## WORKFLOW



`git add file1 file2`

## WORKFLOW



```
git add file1 file2
```

## WORKFLOW



## WORKFLOW



```
git commit -m "Initial commit."
```

## WORKFLOW



```
git commit -m "Initial commit."
```

## HISTORY

- 
- 
- 
- Initial commit

## HISTORY

- Initial commit

**HISTORY**

## HISTORY

- 
- Initial commit

## HISTORY

- 
- 
- 
- **Initial commit**

## HISTORY

- 
- 
- Initial commit

## HISTORY

- 
- 
- 
- Initial commit

## WORKFLOW



# WORKFLOW



`git add file1`

# WORKFLOW



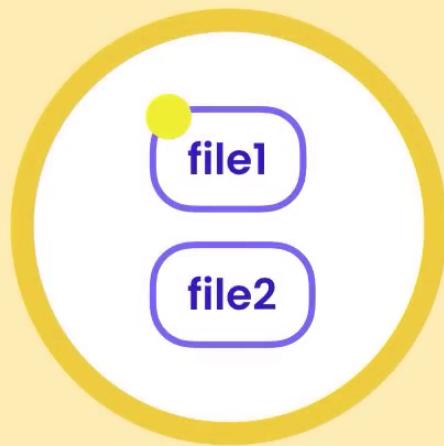
`git add file1`

## WORKFLOW



```
git commit -m "Fixed the bug that..."
```

## WORKFLOW



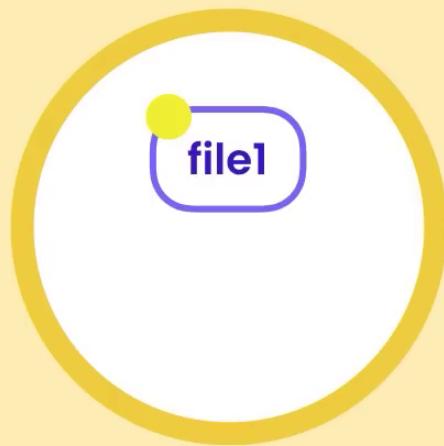
```
git commit -m "Fixed the bug that..."
```

## WORKFLOW



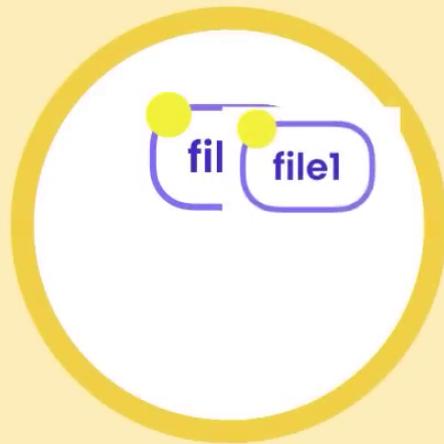
```
git commit -m "Fixed the bug that..."
```

## WORKFLOW



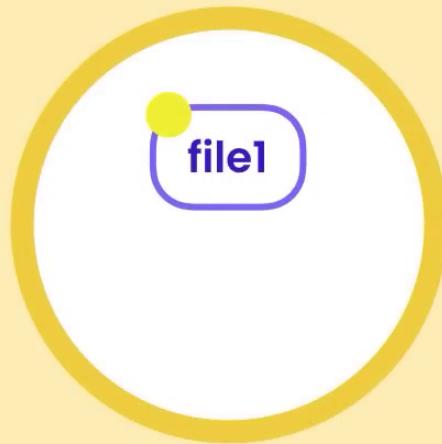
git add file2

## WORKFLOW



```
git commit -m "Removed unused code"
```

## WORKFLOW



```
git commit -m "Removed unused code"
```

**COMMIT**

**ID**

**COMMIT**

ID

Message

Date/time

Author

**COMMIT**

ID

Message

Date/time

Author

Complete snapshot



**Compresses the content**

**Doesn't store duplicate content**

## Staging Files



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➔ git 🌟 master  
➤ ✓ ➔ echo hello > file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master
> ✓ ➤ echo hello > file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ⓘ
> ✓ ➤ echo hello > file2.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ⓘ
> ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ?  
➤ ✓ ➜ git status
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
❯ ✓ ➜ git status  
On branch master
```

No commits yet

Untracked files:

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

file1.txt  
file2.txt

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

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
❯ ✓ ➜
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ?  
❯ ✓ ➜ git add
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
➤ ✓ ➜ git add file1.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
❯ ✓ ➜ git add file1.txt file2.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
❯ ✓ ➜ git add *.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
➤ ✓ ➜ git add _.
```

```
📁 ~/Projects/Moon ➜ git p master ?  
➤ ✓ ➤ git add _
```

```
📁 ~/Projects/Moon ➜ git p master +  
➤ ✓ ➤ git status  
On branch master
```

No commits yet

Changes to be committed:  
(use "git rm --cached <file>..." to unstage)

```
new file:   file1.txt  
new file:   file2.txt
```

```
📁 ~/Projects/Moon ➜ git p master +  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ echo world >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤
```

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

```
new file:    file1.txt  
new file:    file2.txt
```

Changes not staged for commit:

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

(use "git checkout -- <file>..." to discard changes in working directory)

}

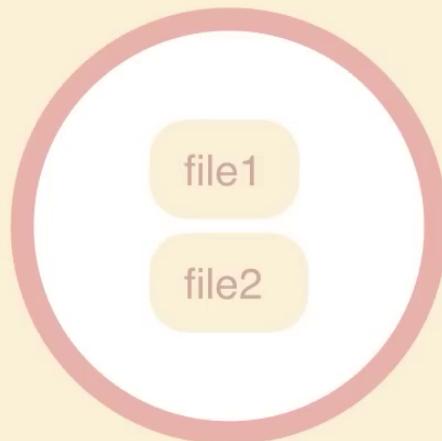
```
modified:   file1.txt
```

~Projects/Moon ➜ git 🌐 master + !

## WORKFLOW



## WORKFLOW



```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
➤ ✓ ➜ git add █
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
❯ ✓ ➜ git add file1.txt
```

```
📁 ~/Projects/Moon ➜ git p master + !  
➤ ✓ ➤ git add file1.txt
```

```
📁 ~/Projects/Moon ➜ git p master +  
➤ ✓ ➤ git status  
On branch master
```

No commits yet

Changes to be committed:  
(use "git rm --cached <file>..." to unstage)

```
new file:   file1.txt  
new file:   file2.txt
```

```
📁 ~/Projects/Moon ➜ git p master +  
➤ ✓ ➤
```

# Committing Changes



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨  
➤ ✓ ➜ git commit -m "█"
```

```
📁 ~/Projects/Moon ➜ git p master +  
➤ ✓ ➜ git commit -m "Initial commit." █
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨  
➤ ✓ ➜ git commit
```

 COMMIT\_EDITMSG 

Users &gt; moshfeghhamedani &gt; Projects &gt; Moon &gt; .git &gt; COMMIT\_EDITMSG

```
|  
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the  
#  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
# new file: file1.txt  
# new file: file2.txt  
#
```

```
📁 ~/Projects/Moon ➜ git 🌟 master ✨  
➤ ✓ ➔ git commit  
hint: Waiting for your editor to close the file... █
```

 COMMIT\_EDITMSG 

Users &gt; moshfeghhamedani &gt; Projects &gt; Moon &gt; .git &gt; COMMIT\_EDITMSG

# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the  
#  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
# new file: file1.txt  
# new file: file2.txt  
#



## COMMIT\_EDITMSG ●



Users > moshfeghhamedani > Projects > Moon > .git > COMMIT\_EDITMSG

### A short description

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the
#
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   new file:    file1.txt
#   new file:    file2.txt
#
```

1 COMMIT\_EDITMSG ●

Users > moshfeghhamedani > Projects > Moon > .git > COMMIT\_EDITMSG

A short description.

Long description.

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the
#
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   new file:    file1.txt
#   new file:    file2.txt
```

1 COMMIT\_EDITMSG ●

Users > moshfeghhamedani > Projects > Moon > .git > COMMIT\_EDITMSG

Initial commit.

Long description.

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the
#
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   new file:    file1.txt
#   new file:    file2.txt
```

 COMMIT\_EDITMSG ●

Users &gt; moshfeghhamedani &gt; Projects &gt; Moon &gt; .git &gt; COMMIT\_EDITMSG

 Initial commit. This is our first commit. |  
  
  
  
  
  
 # Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the  
#  
# On branch master  
#  
# Initial commit  
#  
# Changes to be committed:  
# new file: file1.txt  
# new file: file2.txt

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨  
❯ ✓ ➜ git commit  
[master (root-commit) 3b0003b] Initial commit.  
2 files changed, 3 insertions(+)  
create mode 100644 file1.txt  
create mode 100644 file2.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ➞  
❯ ✓ ➜ ⟳
```

# Committing Best Practices

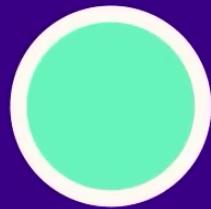


**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

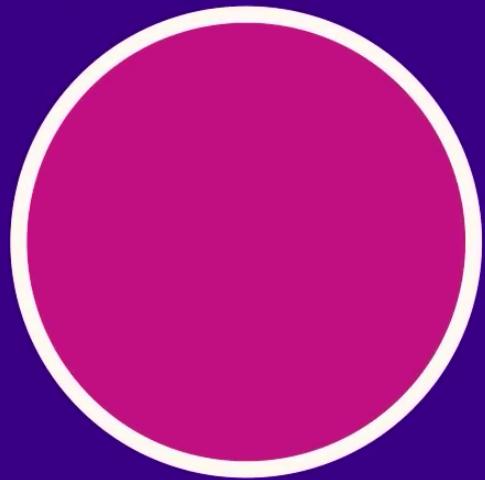
## COMMIT SIZE MATTERS



XS



M



XXL

## HISTORY

- Update file1

## HISTORY

- **Update file3**
- **Update file2**
- **Update file1**

## HISTORY

- Update file3
- Update file2
- Update file1

## HISTORY

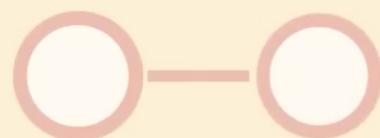
- Build the forum

**COMMIT OFTEN**

**COMMIT OFTEN**



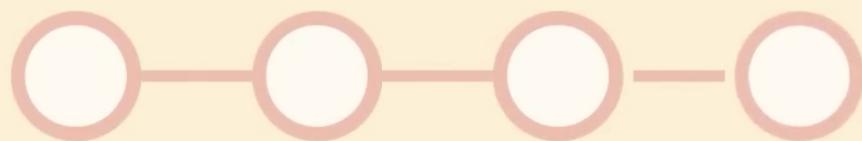
**COMMIT OFTEN**



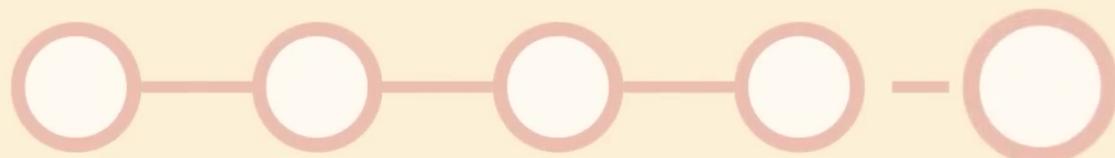
**COMMIT OFTEN**



**COMMIT OFTEN**



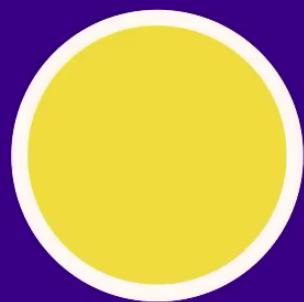
**COMMIT OFTEN**



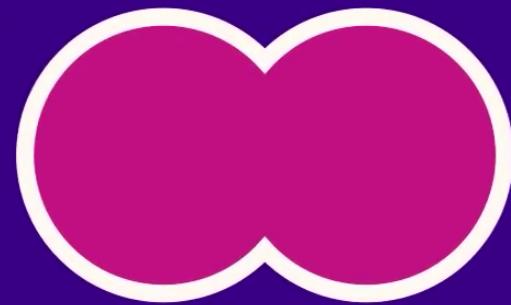
## **COMMIT OFTEN**

**As you reach a state you want to record  
THEN make a commit**



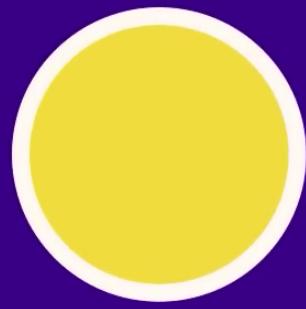






**Bug Fix**

**Typo**



Bug Fix

Typo

Make it  
mean  
something



# Skipping the Staging Area



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➤ git p master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ⓘ  
➤ ✓ ➤ git commit -a
```

```
📁 ~/Projects/Moon ➤ git ⚡ master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git ⚡ master ⓘ  
➤ ✓ ➤ git commit -a -m
```

```
📁 ~/Projects/Moon ➤ git ⚡ master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git ⚡ master ⚡  
➤ ✓ ➤ git commit -am
```

```
📁 ~/Projects/Moon ➤ git ⚡ master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git ⚡ master !  
➤ ✓ ➤ git commit -am "Fix the bug that pr
```

```
📁 ~/Projects/Moon ➤ git ↵ master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git ↵ master !  
➤ ✓ ➤ git commit -am "Fix the bug that prevented the users from signi  
ng up." █
```

```
📁 ~/Projects/Moon ➤ git 🌐 master  
➤ ✓ ➤ echo test >> file1.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master !  
➤ ✓ ➤ git commit -am "Fix the bug that prevented the users from signing up."  
[master 8f092f7] Fix the bug that prevented the users from signing up  
·  
1 file changed, 1 insertion(+)  
}
```

```
📁 ~/Projects/Moon ➤ git 🌐 master  
➤ ✓ ➤
```

# Removing Files



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git 🌐 master  
❯ ✓ ➜ rm file2.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master  
➤ ✓ ➤ rm file2.txt
```

```
📁 ~/Projects/Moon ➜ git 🌐 master !
```

```
➤ ✓ ➤ git status
```

```
On branch master
```

```
Changes not staged for commit:
```

```
  (use "git add/rm <file>..." to update what will be committed)  
  (use "git restore <file>..." to discard changes in working directory)  
y)
```

```
    deleted:    file2.txt
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

```
📁 ~/Projects/Moon ➜ git 🌐 master !  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ⚡  
➤ ✓ ➔ git ls-files  
file1.txt  
file2.txt
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ⚡  
➤ ✓ ➔
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ⚡  
❯ ✓ ➜ git add file2.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ⚡  
➤ ✓ ➤ git add file2.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git ls-files  
file1.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ❗  
➤ ✓ ➤ git add file2.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git ls-files  
file1.txt  
}
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git status  
On branch master  
Changes to be committed:  
(use "git restore --staged <file>..." to unstage)  
  deleted:    file2.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git commit -m "Remove unused code."  
[master 138c8ef] Remove unused code.  
1 file changed, 1 deletion(-)  
delete mode 100644 file2.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git p master  
➤ ✓ ➜ git rm file2.txt *.txt
```

# Renaming or Moving Files



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔ ls  
file1.txt
```

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔
```

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔ ls  
file1.txt
```

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔ mv file1.txt main.js
```

**LIKE AND SHARE!**

IF YOU ENJOY THIS VIDEO

```
📂 ~/Projects/Moon ➤ git 🌐 master ⚠ ?  
➤ ✓ ➤ git status  
On branch master  
Changes not staged for commit:  
  (use "git add/rm <file>..." to update what will be committed)  
    (use "git restore <file>..." to discard changes in working directory)  
      deleted:    file1.txt  
  
Untracked files:  
  (use "git add <file>..." to include in what will be committed)  
    main.js  
  
no changes added to commit (use "git add" and/or "git commit -a")
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ⚠ ?  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ❗ ?  
➤ ✓ ➤ git add file1.txt
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + ?  
➤ ✓ ➤ git add file1.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ⚡ ?  
➤ ✓ ➔ git add file1.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + ?  
➤ ✓ ➔ git add main.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➔ git add main.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ⚡ ?  
➤ ✓ ➔ git add file1.txt
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + ?  
➤ ✓ ➔ git add main.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➔ git status  
On branch master  
Changes to be committed:  
(use "git restore --staged <file>..." to unstage)  
  renamed:   file1.txt -> main.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➔
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨  
➤ ✓ ➜ git mv
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ✨
> ✓ ➤ git mv main.js file1.js
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ✨
> ✓ ➤ git mv main.js file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git mv main.js file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git status  
On branch master  
Changes to be committed:  
(use "git restore --staged <file>..." to unstage)  
  renamed:    file1.txt -> file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤
```

```
📂 ~/Projects/Moon ➤ git 🌐 master +  
➤ ✓ ➤ git commit -m "Refactor code."  
[master 7e3f6b1] Refactor code.  
1 file changed, 0 insertions(+), 0 deletions(-)  
rename file1.txt => file1.js (100%)
```

```
📂 ~/Projects/Moon ➤ git 🌐 master  
➤ ✓ ➤
```

# Ignoring Files



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔ mkdir logs
```

```
📁 ~/Projects/Moon ➔ git 🌐 master  
❯ ✓ ➔
```

```
📁 ~/Projects/Moon ➜ git 🌟 master  
❯ ✓ ➜ mkdir logs
```

```
📁 ~/Projects/Moon ➜ git 🌟 master  
❯ ✓ ➜ echo hello > logs/dev.log
```

```
📁 ~/Projects/Moon ➜ git 🌐 master  
➤ ✓ ➔ mkdir logs
```

```
📁 ~/Projects/Moon ➜ git 🌐 master  
➤ ✓ ➔ echo hello > logs/dev.log
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
➤ ✓ ➔ git status  
On branch master  
Untracked files:  
(use "git add <file>..." to include in what will be committed)  
  logs/
```

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

```
📁 ~/Projects/Moon ➜ git 🌐 master ?  
➤ ✓ ➔
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ?  
➤ ✓ ➤ echo logs/ > .gitignore
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ?  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git p master ?  
➤ ✓ ➤ echo logs/ > .gitignore
```

```
📁 ~/Projects/Moon ➜ git p master ?  
➤ ✓ ➤ code .gitignore
```

✖ .gitignore



Users > moshfeghamedani > Projects > Moon > .gitignore

logs/

|



1 .gitignore



Users > moshfeghamedani > Projects > Moon > .gitignore

logs/

main.log

|



1 .gitignore



Users > moshfeghamedani > Projects > Moon > .gitignore

logs/

main.log

\*.log



```
📂 ~/Projects/Moon ➤ git 🌐 master ?  
➤ ✓ ➤ 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)

```
📂 ~/Projects/Moon ➤ git 🌐 master ?  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ?
```

```
› ✓ ➜ 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)

```
📁 ~/Projects/Moon ➜ git 🌐 master ?
```

```
› ✓ ➜ git add .gitignore
```

```
📁 ~/Projects/Moon ➜ git 🌐 master +
```

```
› ✓ ➜ gil add .gitignore
```

```
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)

```
~$ cd ~/Projects/Moon
~$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#     .gitignore
```

```
~$ git add .gitignore
~$ git commit -m "Add gitignore"
[master e5fd9a2] Add gitignore
 1 file changed, 3 insertions(+)
 create mode 100644 .gitignore
```

```
~$ git status
# On branch master
# Your branch is up-to-date with 'origin/master'.
```

```
📁 ~/Projects/Moon ➜ git ⚡ master  
❯ ✓ ➜ mkdir bin
```

```
📁 ~/Projects/Moon ➜ git ⚡ master  
❯ ✓ ➜
```

GitHub - GitHub / gitignore: A collection of .gitignore files for various projects.

github.com/github/gitignore

Sign up

github / gitignore

Watch 3.1k

Star 106k

Fork 56.2k

Code Pull requests 168 Actions Projects Security ...

Join GitHub today

Dismiss

GitHub is home to over 50 million developers working together to host and review code, manage projects, and build software together.

Sign up



gitignore/Java.gitignore at mas



github.com/github/gitignore/blob/master/Java.gitignore



23 lines (18 sloc) | 278 Bytes

[Raw](#)[Blame](#)

```
1 # Compiled class file
2 *.class
3
4 # Log file
5 *.log
6
7 # BlueJ files
8 *.ctxt
9
10 # Mobile Tools for Java (J2ME)
11 .mtj.tmp/
12
13 # Package Files #
14 *.jar
15 *.war
```

## Short Status



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git ⚡ master ⚡ ?  
➤ ✓ ➤ git status -s  
M file1.js  
?? file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ⚡ ?  
➤ ✓ ➤
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ❗ ?  
❯ ✓ ➜ git status -s  
M file1.js  
?? file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ❗ ?  
❯ ✓ ➜ git add file1.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + ?  
❯ ✓ ➜ git status -s  
M file1.js  
?? file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + ?  
❯ ✓ ➜
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + ?  
❯ ✓ ➜ git status -s  
M file1.js  
?? file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + ?  
❯ ✓ ➜ git add file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master +  
❯ ✓ ➜ git status -s  
M file1.js  
A file2.js  
❯
```

```
📁 ~/Projects/Moon ➜ git ⚡ master +  
❯ ✓ ➜
```

# Viewing the Staged & Unstaged Changes



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨  
➤ ✓ ➔ git diff --staged
```

```
diff --git a/file1.js b/file1.js
index badfb70..47c3216 100644
--- a/file1.js
+++ b/file1.js
@@ -1,3 +1,5 @@
 hello
 world
 test
+sky
+ocean
diff --git a/file2.js b/file2.js
new file mode 100644
index 0000000..f5e95e7
--- /dev/null
+++ b/file2.js
@@ -0,0 +1 @@
+sky
```

⚡ ~/Projects/Moon ➔ git 🌐 master +

```
📁 ~/Projects/Moon ➜ git ⚡ master +  
➤ ✓ ➜ git diff
```

```
📁 ~/Projects/Moon ➜ git ⚡ master +  
➤ ✓ ➜
```

```
📂 ~/Projects/Moon ➜ git 🌐 master + !  
❯ ✓ ➜ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➜ git 🌐 master + !  
❯ ✓ ➜ git diff  
diff --git a/file1.js b/file1.js  
index 47c3216..8636dbe 100644  
--- a/file1.js  
+++ b/file1.js  
@@ -1,4 +1,4 @@  
-hello  
+hello world  
world  
test  
sky
```

```
📂 ~/Projects/Moon ➜ git 🌐 master + !
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
❯ ✓ ➜ git diff --staged
```

## Visual Diff Tools



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

## DIFF TOOLS

- KDiff3
- P4Merge

## DIFF TOOLS

- KDiff3
- P4Merge
- WinMerge (Windows Only)

## DIFF TOOLS

- KDiff3
- P4Merge
- WinMerge (Windows Only)
- VSCode

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
❯ ✓ ➜ git config --global diff.tool vscode
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global diff.tool vscode
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global difftool.vscode.cmd "code --wait"
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global diff.tool vscode
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global difftool.vscode.cmd "code --wait --diff"
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global diff.tool vscode
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git config --global difftool.vscode.cmd "code --wait --diff $LOCAL $REMOTE"
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git config --global diff.tool vscode
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git config --global difftool.vscode.cmd "code --wait --diff $LOCAL $REMOTE"
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git config --global -e
```

.gitignore

JS file1.js

gitconfig X



Users > moshfeghhamedani > .gitconfig

```
 嫌う = git-ls 嫌う -- そ  
process = git-lfs filter-process  
required = true  
[diff]  
  tool = vscode  
[difftool "vscode"]  
  cmd = "code --wait --diff "
```

.gitignore

JS file1.js

Git .gitconfig X

Users > moshfeghhamedani > .gitconfig

```
 嫌う = git-ls 嫌う -- そ  
process = git-lfs filter-process  
required = true  
[diff]  
  tool = vscode  
[difftool "vscode"]  
  cmd = "code --wait --diff $LOCAL $REMOTE"
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
❯ ✓ ➜ git difftool
```

```
📁 ~/Projects/Moon ➜ git 🌐 master + !  
➤ ✓ ➔ git difftool
```

```
Viewing (1/1): 'file1.js'  
Launch 'vscode' [Y/n]? █
```

/var/folders/ms/\_6d\_3754zg7tw6sr0wnp0c80000gn/T/FHtS6J\_file1.js ↵ ...

Users &gt; moshfeghhamedani &gt; Projects &gt; Moon &gt; file1.js

- hello

world

test

sky

ocean

+ hello world

world

test

sky

ocean

You, an h



```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git difftool
```

Viewing (1/1): 'file1.js'  
Launch 'vscode' [Y/n]? y

```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git difftool --staged
```

Viewing (1/2): 'file1.js'  
Launch 'vscode' [Y/n]? █

 JS /var/folders/ms/\_6d\_3754zg7tw6sr0wnp0c80000gn/T/sUHeBb\_file1.js ↔ /var/folders/ms/\_6d\_3754zg7tw6sr0wnp0c80000gn/T/qhOxAB\_file1.js      ...

var > folders > ms > \_6d\_3754zg7tw6sr0wnp0c80000gn > T >  JS qhOxAB\_file1.js

hello

world

test

hello

world

test

+ sky

+ ocean

```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git difftool
```

Viewing (1/1): 'file1.js'  
Launch 'vscode' [Y/n]? y

```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git difftool --staged
```

Viewing (1/2): 'file1.js'  
Launch 'vscode' [Y/n]? y

Viewing (2/2): 'file2.js'  
Launch 'vscode' [Y/n]? █ }

## Viewing the History



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git 🌐 master ✨ !  
❯ ✓ ➜ git log
```

```
commit 921a2ffd7042ed0eb773676509fda87b5c05abf5 (HEAD -> master)
```

```
Author: Mosh Hamedani <programmingwithmosh@gmail.com>
```

```
Date: Tue Aug 4 16:56:10 2020 -0700
```

Remove the bin directory that was accidentally committed.

```
commit d601b900c717da53806a7126b3fe587c7ae16d9e
```

```
Author: Mosh Hamedani <programmingwithmosh@gmail.com>
```

```
Date: Tue Aug 4 16:52:21 2020 -0700
```

Include bin/ in gitignore.

```
commit 42ce2fbf51d4020b64fe8b81f736d85546d622bf
```

```
Author: Mosh Hamedani <programmingwithmosh@gmail.com>
```

```
Date: Tue Aug 4 16:50:34 2020 -0700
```

Add bin.

```
commit e5fd9a2734d53d335e21c2a4314af990370a21ad
```

```
⌚ ~/Projects/Moon ➜ git 🌐 master + !
```

```
➤ ✓ ➤ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
⌚ ~/Projects/Moon ➜ git 🌐 master + !
```

```
➤ ✓ ➤
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git log --oneline --reverse  
3b0003b Initial commit.  
8f092f7 Fix the bug that prevented the users from signing up.  
138c8ef Remove unused code.  
7e3f6b1 Refactor code.  
e5fd9a2 Add gitignore  
42ce2fb Add bin.  
d601b90 Include bin/ in gitignore.  
921a2ff (HEAD -> master) Remove the bin directory that was accidentally committed.
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !
```

```
➤ ✓
```

## Viewing a Commit



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git show d601b90
```

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git show d60
```

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➔ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➔ git show HEAD
```

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git show HEAD~
```

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git log --oneline
```

921a2ff (**HEAD -> master**) Remove the bin directory that was accidentally committed.

d601b90 Include bin/ in gitignore.

42ce2fb Add bin.

e5fd9a2 Add gitignore

7e3f6b1 Refactor code.

138c8ef Remove unused code.

8f092f7 Fix the bug that prevented the users from signing up.

3b0003b Initial commit.

```
📁 ~/Projects/Moon ➜ git 🌟 master
```

```
➤ ✓ ➤ git show HEAD~1
```

Author: Mosh Hamedani <programmingwithmosh@gmail.com>  
Date: Tue Aug 4 16:52:21 2020 -0700

Include bin/ in gitignore.

```
diff --git a/.gitignore b/.gitignore
index 8432ad3..1dcc30c 100644
--- a/.gitignore
+++ b/.gitignore
@@ -1,3 +1,4 @@
 logs/
 main.log
-*.
log
\ No newline at end of file
+*.
log
+bin/
\ No newline at end of file
```

▶ ~/Projects/Moon ➔ git ↵ master ➔

```
📁 ~/Projects/Moon ➜ git p master  
➤ ✓ ➜ git show HEAD~1:.gitignore
```

```
📁 ~/Projects/Moon ➜ git p master  
➤ ✓ ➜ git show HEAD~1:bin/app.bin
```

```
📁 ~/Projects/Moon ➜ git 🌐 master  
➤ ✓ ➜ git show HEAD~1:.gitignore  
logs/  
main.log  
*.log  
bin/
```

```
📁 ~/Projects/Moon ➜ git 🌐 master  
➤ ✓ ➜
```

bin/

```
git ~/Projects/Moon git master
> ✓ git show HEAD
commit 921a2ffd7042ed0eb773676509fda87b5c05abf5 (HEAD -> master)
Author: Mosh Hamedani <programmingwithmosh@gmail.com>
Date:   Tue Aug 4 16:56:10 2020 -0700
```

Remove the bin directory that was accidentally committed.

```
diff --git a/bin/app.bin b/bin/app.bin
deleted file mode 100644
index ce01362..0000000
--- a/bin/app.bin
+++ /dev/null
@@ -1 +0,0 @@
-hello
```

```
git ~/Projects/Moon git master
```

```
📁 ~/Projects/Moon ➜ git ⚡ master  
❯ ✓ ➜ git ls-tree
```

```
📂 ~/Projects/Moon ➤ git 🌐 master
> ✓ ➤ git ls-tree HEAD~1
100644 blob 1dcc30c4cd47f8915741af2cfef91e16e0dc7d89 .gitignore
040000 tree 64629cd51ef4a65a9d9cb9e656e1f46e07e1357f bin
100644 blob badfb70fd8b1725682b26674f7b2882e94078579 file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master
> ✓ ➤
```

## GIT OBJECTS

- Commits
- Blobs (Files)
- Trees (Directories)
- Tags

# Unstaging Files



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ✨ !  
➤ ✓ ➤ git [status] -s
```

```
📁 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📁 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git add file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git reset --hard
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git reset --soft
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git restore
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git --version  
git version 2.28.0
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git restore --staged
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git --version  
git version 2.28.0
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git restore --staged file1.js file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git --version  
git version 2.28.0
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git restore --staged .
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git status -s  
MM file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git --version  
git version 2.28.0
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤ git restore --staged file1.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
❯ ✓ ➤
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
M file1.js  
A file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤
```

```
}
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git status -s  
M file1.js  
A file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master + !  
❯ ✓ ➜ git restore --staged file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ! ?  
❯ ✓ ➜
```

```
📁 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git status -s  
M file1.js  
A file2.js
```

```
📁 ~/Projects/Moon ➤ git 🌐 master + !  
➤ ✓ ➤ git restore --staged file2.js
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ! ?  
➤ ✓ ➤ git status -s  
M file1.js  
?? file2.js
```

```
📁 ~/Projects/Moon ➤ git 🌐 master ! ?  
➤ ✓ ➤
```

# Discarding Local Changes



**Mosh Hamedani**  
[codewithmosh.com](http://codewithmosh.com)

```
📂 ~/Projects/Moon ➤ git 🌐 master ❗ ?  
❯ ✓ ➤ git status -s  
M file1.js  
?? file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ❗ ?  
❯ ✓ ➤ git restore _
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ❓  
❯ ✓ ➤ git status -s  
?? file2.js
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ❓  
❯ ✓ ➤ git clean  
fatal: clean.requireForce defaults to true and neither -i, -n, nor -f  
given; refusing to clean  
}
```

```
📂 ~/Projects/Moon ➤ git 🌐 master ❓
```

```
📁 ~/Projects/Moon ➜ git 🌐 master ⓘ
> ✓ ➜ git clean -h
usage: git clean [-d] [-f] [-i] [-n] [-q] [-e <pattern>] [-x | -X] [-]
-> <paths>...
```

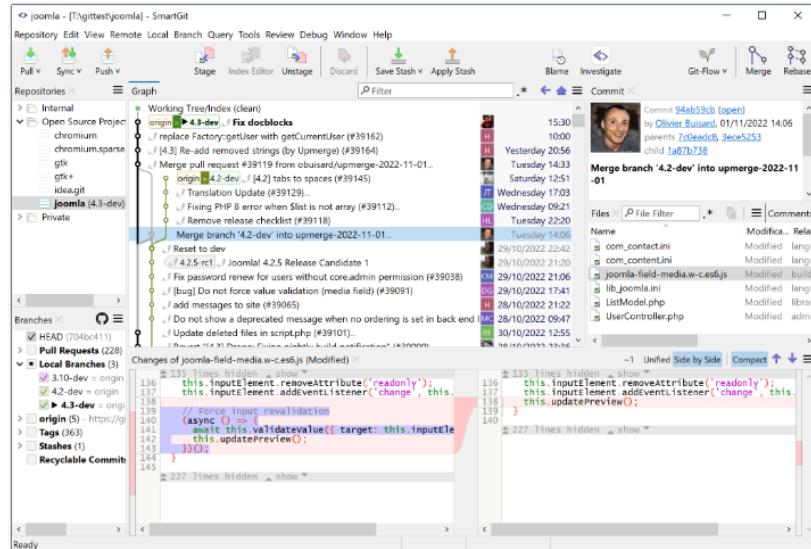
-q, --quiet	do not print names of files removed
-n, --dry-run	dry run
-f, --force	force
-i, --interactive	interactive cleaning
-d	remove whole directories
-e, --exclude <pattern>	add <pattern> to ignore rules
-x	remove ignored files, too
-X	remove only ignored files

```
📁 ~/Projects/Moon ➜ git 🌐 master ⓘ
> ↵ SIGHUP(1) ➜
```

```
📁 ~/Projects/Moon ➜ git ⚡ master ?  
▶ ↵ SIGHUP(1) ➜ git clean -fd  
Removing file2.js
```

```
📁 ~/Projects/Moon ➜ git ⚡ master  
▶ ✓ ➜ git status -s
```

```
📁 ~/Projects/Moon ➜ git ⚡ master  
▶ ✓ ➜
```



## Get your commit done.

SmartGit is the most popular Git GUI client at [slant.co.](https://slant.co/)

100,000+ developers around the world trust in SmartGit.

SmartGit supports GitHub, Bitbucket, GitLab and Azure DevOps.

[Purchase](#)[Download](#)