

Git Tutorial

08/02/2021

```
[aparnashedigumme@Aparnas-MacBook-Pro ~ % git --version
git version 2.30.1 (Apple Git-130)
[aparnashedigumme@Aparnas-MacBook-Pro ~ % git config --global user.name "aparna.s"
[aparnashedigumme@Aparnas-MacBook-Pro ~ % git config --global user.email "aparna.s@consultadd.com"
[aparnashedigumme@Aparnas-MacBook-Pro ~ % mkdir git_learning
[aparnashedigumme@Aparnas-MacBook-Pro ~ % cd git_learning
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % 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/aparnashedigumme/git_learning/.git/
aparnashedigumme@Aparnas-MacBook-Pro git_learning %
```

```
[aparnashedigumme@Aparnas-MacBook-Pro ~ % git config --global user.name "aparna.s"
[aparnashedigumme@Aparnas-MacBook-Pro ~ % git config --global user.email "aparna.s@consultadd.com"
[aparnashedigumme@Aparnas-MacBook-Pro ~ % mkdir git_learning
[aparnashedigumme@Aparnas-MacBook-Pro ~ % cd git_learning
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % 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/aparnashedigumme/git_learning/.git/
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls
index.html
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add index.html
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
\On branch master

No commits yet

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

[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add -all
error: did you mean `--all` (with two dashes)?
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add --all
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .DS_Store
    new file:   bluestyle.css
    new file:   index.html
    new file:   readme.txt

[aparnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -m "First release of Hello World!"
[dquote]
[dquote] k
[dquote] "
[master (root-commit) 07c70fc] First release of Hello World
  4 files changed, 56 insertions(+)
  create mode 100644 .DS_Store
  create mode 100644 bluestyle.css
  create mode 100644 index.html
  create mode 100644 readme.txt
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls
bluestyle.css  index.html      readme.txt
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls -a
.          ..          .DS_Store      .git      bluestyle.css  index.html      readme.txt
[aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls -la
total 12
drwxr-xr-x  2 aparna.s  staff  68 Jun 20 10:45 .
drwxr-xr-x  2 aparna.s  staff  68 Jun 20 10:45 ..
-rw-r--r--  1 aparna.s  staff  12 Jun 20 10:45 .DS_Store
-rw-r--r--  1 aparna.s  staff  12 Jun 20 10:45 .git
-rw-r--r--  1 aparna.s  staff  12 Jun 20 10:45 bluestyle.css
-rw-r--r--  1 aparna.s  staff  12 Jun 20 10:45 index.html
-rw-r--r--  1 aparna.s  staff  12 Jun 20 10:45 readme.txt
```

```

See 'git help <command>' to read about a specific subcommand

Main Porcelain Commands
  add          Add file contents to the index
  am           Apply a series of patches from a mailbox
  archive      Create an archive of files from a named tree
  bisect       Use binary search to find the commit that introduced a bug
  branch       List, create, or delete branches
  bundle       Move objects and refs by archive
  checkout     Switch branches or restore working tree files
  cherry-pick  Apply the changes introduced by some existing commits
  citool       Graphical alternative to git-commit
  clean        Remove untracked files from the working tree
  clone        Clone a repository into a new directory
  commit       Record changes to the repository
  describe     Give an object a human readable name based on an available ref
  diff         Show changes between commits, commit and working tree, etc
  fetch        Download objects and refs from another repository
  format-patch Prepare patches for e-mail submission
  gc           Cleanup unnecessary files and optimize the local repository
  gitk         The Git repository browser
  grep         Print lines matching a pattern
  gui          A portable graphical interface to Git
  init         Create an empty Git repository or reinitialize an existing one
  log          Show commit logs
  maintenance Run tasks to optimize Git repository data
  merge        Join two or more development histories together
  mv           Move or rename a file, a directory, or a symlink
  notes        Add or inspect object notes
  pull         Fetch from and integrate with another repository or a local branch
  push         Update remote refs along with associated objects
  range-diff   Compare two commit ranges (e.g. two versions of a branch)
  rebase       Reapply commits on top of another base tip
  reset        Reset current HEAD to the specified state
  restore      Restore working tree files
  revert      Revert some existing commits
  rm           Remove files from the working tree and from the index
  shortlog    Summarize 'git log' output
  show         Show various types of objects
  sparse-checkout Initialize and modify the sparse-checkout
  stash        Stash the changes in a dirty working directory away
  status       Show the working tree status
  submodule    Initialize, update or inspect submodules
  switch       Switch branches
  tag          Create, list, delete or verify a tag object signed with GPG
  worktree    Manage multiple working trees

Ancillary Commands / Manipulators
  config      Get and set repository or global options
  fast-export Git data exporter
  fast-import Backend for fast Git data importers
  filter-branch Rewrite branches
  mergetool   Run merge conflict resolution tools to resolve merge conflicts
  pack-refs   Pack heads and tags for efficient repository access
  prune      Prune all unreachable objects from the object database
  reflog     Manage reflog information
  remote     Manage set of tracked repositories
  repack     Pack unpacked objects in a repository
  replace    Create, list, delete refs to replace objects

Ancillary Commands / Interrogators
  annotate    Annotate file lines with commit information
  blame       Show what revision and author last modified each line of a file
  bugreport   Collect information for user to file a bug report
  count-objects Count unpacked number of objects and their disk consumption
  difftool    Show changes using common diff tools
  fsck        Verifies the connectivity and validity of the objects in the database
  gitweb      Git web interface (web frontend to Git repositories)
  help        Display help information about Git
  instaweb    Instantly browse your working repository in gitweb
  merge-tree  Show three-way merge without touching index
  rerere      Reuse recorded resolution of conflicted merges
  show-branch Show branches and their commits
  verify-commit Check the GPG signature of commits
  verify-tag   Check the GPG signature of tags
  whatchanged Show logs with difference each commit introduces

Interacting with Others
:■

```

```
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -help
usage: git commit [<options>] [--] <pathspec>...

-q, --quiet           suppress summary after successful commit
-v, --verbose          show diff in commit message template

Commit message options
  -F, --file <file>      read message from file
  --author <author>       override author for commit
  --date <date>           override date for commit
  -m, --message <message>
                        commit message
  -c, --reedit-message <commit>
                        reuse and edit message from specified commit
  -C, --reuse-message <commit>
                        reuse message from specified commit
  --fixup <commit>        use autosquash formatted message to fixup specified commit
  --squash <commit>        use autosquash formatted message to squash specified commit
  --reset-author          the commit is authored by me now (used with -C/-c/--amend)
  -s, --signoff           add a Signed-off-by trailer
  -t, --template <file>
                        use specified template file
  -e, --edit              force edit of commit
  --cleanup <mode>         how to strip spaces and #comments from message
  --status                include status in commit message template
  -S, --gpg-sign[=<key-id>]
                        GPG sign commit

Commit contents options
  -a, --all               commit all changed files
  -i, --include            add specified files to index for commit
  --interactive           interactively add files
  -p, --patch              interactively add changes
  -o, --only               commit only specified files
  -n, --no Verify          bypass pre-commit and commit-msg hooks
  --dry-run                show what would be committed
  --short                 show status concisely
  --branch                show branch information
  --ahead-behind           compute full ahead/behind values
  --porcelain             machine-readable output
  --long                  show status in long format (default)
  -z, --null               terminate entries with NUL
  --amend                 amend previous commit
  --no-post-rewrite        bypass post-rewrite hook
  -u, --untracked-files[=<mode>]
                        show untracked files, optional modes: all, normal, no. (Default: all)
  --pathspec-from-file <file>
                        read pathspec from file
  --pathspec-file-nul     with --pathspec-from-file, pathspec elements are separated with NUL character
```

```
aparnashedigumme@Aparnas-MacBook-Pro git_learning %
```

```
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git branch hello-world-images
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git branch
  hello-world-images
* master
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout hello-world-images
error: pathspec 'hello-worl-images' did not match any file(s) known to git
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout hello-world-images
Switched to branch 'hello-world-images'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add --all
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch hello-world-images
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   img_hello_world.jpg
    modified:   index.html

aparnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -m "Added image to Hello world"
[hello-world-images c58e603] Added image to Hello world
 2 files changed, 41 insertions(+), 47 deletions(-)
  create mode 100644 img_hello_world.jpg
  rewrite index.html (61%)
aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls
bluestyle.css      img_hello_world.jpg      index.html          readme.txt
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout master
Switched to branch 'master'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls
bluestyle.css  index.html  readme.txt
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout -b emergency-fix
Switched to a new branch 'emergency-fix'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch emergency-fix
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add index.html
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch emergency-fix
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   index.html
```

```
parnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -m "emergency-fix"
emergency-fix f42dc1e] emergency-fix
1 file changed, 16 insertions(+), 23 deletions(-)
parnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout master
witched to branch 'master'
```

```
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout master
Switched to branch 'master'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git merge emergency-fix
Updating 4f58453..f42dc1e
Fast-forward
 index.html | 39 ++++++-----+
 1 file changed, 16 insertions(+), 23 deletions(-)
```

```
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout hello-world-images
Switched to branch 'hello-world-images'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add --all
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -m "added a new image"
[hello-world-images 563736b] added a new image
 3 files changed, 2 insertions(+), 2 deletions(-)
 create mode 100644 .DS_Store
 create mode 100644 img_hello_git.jpg
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git checkout master
Switched to branch 'master'
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git merge hello-world-image
merge: hello-world-image - not something we can merge
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git merge hello-world-images
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch master
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Changes to be committed:
  new file:  .DS_Store
  new file:  img_hello_git.jpg
  new file:  img_hello_world.jpg

Unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:  index.html

aparnashedigumme@Aparnas-MacBook-Pro git_learning % git add index.html
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git status
On branch master
All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)

Changes to be committed:
  new file:  .DS_Store
  new file:  img_hello_git.jpg
  new file:  img_hello_world.jpg
  modified:  index.html

aparnashedigumme@Aparnas-MacBook-Pro git_learning % git commit -m "merged wit hello-world-images after fixing conflicts"
[master 32dbb1a] merged wit hello-world-images after fixing conflicts
aparnashedigumme@Aparnas-MacBook-Pro git_learning % ls
bluestyle.css      img_hello_git.jpg      img_hello_world.jpg      index.html      readme.txt
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git branch
  hello-world-images
* master
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git branch -d hello-world-images
Deleted branch hello-world-images (was 563736b).
aparnashedigumme@Aparnas-MacBook-Pro git_learning % git branch
* master
```