



# **GIT COMMANDS FOR DEVOPS**

**#30daysofdevops**



**Mohammed Ismail**  
@mohammedismaildevops

# **SETUP COMMANDS**

**Configuring user information used across all local repositories.**

1. **git config --global user.name "[firstname lastname]"** - set a name that is identifiable for credit when review version history
2. **git config --global user.email "[valid-email]"** - set an email address that will be associated with each history marker
3. **git config --global color.ui auto** - set automatic command line coloring for Git for easy reviewing

# **CREATING AND CLONING REPOS**

4. **git init** - initialize an existing directory as a Git repository
5. **git clone [url]** - retrieve an entire repository from a hosted location (git remote repo) via URL



# **STAGE & SNAPSHOT**

6. **git status** - show modified files in working directory, staged for your next commit
7. **git add [file]** - add a file to the staging area
8. **git add .** - add all new and changed files to the staging area
9. **git reset [file]** - unstage a file while retaining the changes in working directory
10. **git diff** - diff of what is changed but not staged
11. **git diff --staged** - diff of what is staged but not yet committed
12. **git commit -m "[descriptive message]"** - commit your staged content as a new commit snapshot

# **BRANCH & MERGE**

- 13. **git branch** - list your branches. '\*' will appear next to the currently active branch
- 14. **git branch [branch-name]** - create a new branch at the current commit
- 15. **git checkout [branch-name]** - switch to another branch and check it out into your working directory
- 16. **git merge [branch]** - merge the specified branch's history into the current one
- 17. **git log** - show all commits in the current branch's history

# **SHARE AND UPDATE**

- 18. **git remote add origin [remote git link]** - Add a remote repository
- 19. **git push origin [branch name]** - push a branch to your remote repository
- 20. **git push -u origin [branch name]** - push changes to remote repository and remember the branch
- 21. **git push** - Push changes to remote repository (remembered branch)
- 22. **git push origin --delete [branch name]** - Delete a remote branch
- 23. **git pull** - Update local repository to the newest commit
- 24. **git pull origin [branch name]** - Pull changes from remote repository



# **TEMPORARY COMMITS**

**Temporarily store modified, tracked files in order to change branches**

- 25. git stash** - Save modified and staged changes
- 26. git stash list** - list stack-order of stashed file changes
- 27. git stash pop** - write working from top of stash stack
- 28. git stash drop** - discard the changes from top of stash stack

# **IGNORING PATTERNS**

**Preventing unintentional staging or committing of files**

**.gitignore -**

**logs/**

**\*.notes**

**pattern\*/**

**Save a file with desired patterns as .gitignore with either direct string matches or wildcard globs.**



# Learned something?

Follow and hit the 



**Mohammed Ismail**  
**@mohammedismaildevops**