

### Setup

|                                     |  |
|-------------------------------------|--|
| Set a user name                     | <code>git config --global user.name "[firstname last name]"</code> |
| Set an email                        | <code>git config --global user.email "[valid email]"</code>        |
| Set automatic command line coloring | <code>git config --global color.ui auto</code>                     |

### Setup and INIT

|                                    |                              |
|------------------------------------|------------------------------|
| Initialize a directory             | <code>git init</code>        |
| Clone an entire repository via URL | <code>git clone [url]</code> |

### Inspect and compare

|                                    |   |
|------------------------------------|---|
| Show the commits in B but no in A  | <code>git log branchB..branchA</code>   |
| Show the commits that changed file | <code>git log --follow[file]</code>     |
| Show the diff of in A but no in B  | <code>git diff branchB...branchA</code> |
| Show any object in git             | <code>git show [SHA]</code>             |

### Tracking Path changes

|   |  |
|---|--|
| Delete the path from project and stage the removal for commit | <code>git rm [file]</code>                     |
| Change an existing file path and stage the move               | <code>git mv [existing-path] [new-path]</code> |
| Show all commits logs with movements in path                  | <code>git log --stat -M</code>                 |

### Ignoring patterns

|  |   |
|--|---|
| Save a file with desired patterns as <code>*.notes/.gitignore</code> |   |
| Configure the file   | <code>git config --global core.excludesfile [file]</code> |

### Stage and Snapshot

|  |                                |
|--|--------------------------------|
| Show modified files in working directory | <code>git status</code>        |
| Add a file to stage                      | <code>git add [file]</code>    |
| Unstage a file                           | <code>git reset [file]</code>  |
| Diff of what is changed but not staged   | <code>git diff</code>          |
| Diff of what is staged but not committed | <code>git diff --staged</code> |

### Stage and Snapshot (cont)

|                            |  |
|----------------------------|--|
| Commit your staged content | <code>git commit -m "[descriptive message]"</code> |
|----------------------------|--|

### Branch and Merge

|                                   |                                       |
|-----------------------------------|---------------------------------------|
| List your branches                | <code>git branch</code>               |
| Create a new branch               | <code>git branch [branch-name]</code> |
| Switch to another branch          | <code>git checkout</code>             |
| Merge the branch into the current | <code>git merge [branch]</code>       |
| Show all the commits              | <code>git log</code>                  |

### Share and Update

|  |   |
|--|---|
| Add a git URL as an alias                    | <code>git remote add [alias] [url]</code> |
| Fetch down all the branches from git remote  | <code>git fetch [alias]</code>            |
| Merge remote branch into your current branch | <code>git merge [alias]/[-branch]</code>  |
| Transmit local branch commits to remote      | <code>git push [alias][branch]</code>     |
| Fetch and merge all commits                  | <code>git pull</code>                     |

### Rewrite History

|   |  |
|---|--|
| Apply any commits of current branch ahead of specified one        | <code>git rebase [branch]</code>       |
| Clear staging area and rewrite working tree from specified commit | <code>git reset --hard [commit]</code> |

### Temporary commits

|   |                             |
|---|-----------------------------|
| Save modified and staged changes          | <code>git stash</code>      |
| List stack-order of stashed file changes  | <code>git stash list</code> |
| Write working from top of stash stack     | <code>git stash pop</code>  |
| Discard the changes from top of the stack | <code>git stash drop</code> |