

Basic Commands

Completely Fresh

Command Listing

```
pwd
cd projects
pwd
git init fresh-project # replace with name of your project create new git repository
ls
cd fresh-project
ls
ls -al
cd .git
ls
cd ..
pwd
git status
mate hipster.txt      creating a txt file in the working directory (untracked file)
git status
git add hipster.txt   move the file from the working directory the staging area
git status
git commit            move the file from the the staging area to the local repository
git status
cd ..
pwd
ls
rm -rf fresh-projects force and recursive removing folder
```

Existing Source

Command Listing

```
pwd
cd projects
unzip ~/Downloads/initializr-verekia-4.0.zip
ls
mv initializr web-project  changing the folder name by using move command
ls
cd web-project
ls
git init      initiate git
ls -al
git status
git add .
git status
git commit -m "My first commit, inline"
git status
git status
ls -al
rm -rf .git
ls
pwd
ls
rm -rf web-project
```

Remote (GitHub)

Command Listing

```
pwd
cd projects
git clone github-https-url # my value: https://github.com/awesomejt/starter-web.git  get a remote repository
ls
cd starter-web
ls
ls -al
cd .git
ls
cd ..
clear
git status
```

Basic Git Workflow / First Commit

Command Listing

```
pwd
cd projects
ls
cd starter-web
ls
git status
mate hipster.txt
ls
git status
git add hipster.txt
git status
git commit
git status
git pull origin master  pull from the branch of the remote repository  origin: the name of the remote repository
clear  master: the name of the branch
git push origin master  push to the branch of the remote repository
```

Adjusting the Git Config

Resetting to author's name from demo in the Git Quick Start.

Command Listing

```
mate ~/.gitconfig
```

~/.gitconfig File

```
[user]
  name = Jason Taylor
  email = jason.git.training
[core]
  editor = mate -w
```

Tracked Files / Express Commit

Command Listing

```
pwd
cd projects
ls
```

```
cd starter-web
ls
git status
mate hipster.txt
git commit -am "Adding more ipsum text" add and commit in one command -am
git ls-files
mate newfile.txt
git status
git ls-files list all the tracked files \(the files that are added\)
git add newfile.txt
git status
git ls-files
```

Editing Files

Command Listing

```
pwd
cd projects/starter-web
clear
ls
mate hipster.txt
git status
git commit -m "Adding new file"
git status
git add hipster.txt
git status
mate hipster.txt
git status
git add hipster.txt
git status
git commit -m "More ipsum for hipsters"
git status
```

Recursive Add

Command Listing

```
pwd
cd projects/starter-web
git status
mkdir -p level1/level2/level3/level4 # last item later deleted (corrected from video)
ls
cd level1
pwd
mate level1-file.txt
ls
cd level2
mate level2-file.txt
ls
cd level3
mate level3-file.txt
ls
rm -rf level4
clear
cd ../../..
ls
git status
git add .
git status
git commit
```

Backing Out Changes

Command Listing

```
pwd
cd projects/starter-web
ls
git status
cd level1
ls
mate level1-file.txt
git status
git add level1-file.txt
git status
mate level1-file.txt
clear
git status
git reset HEAD level1-file.txt Back from stage to unstage \(from after adding —to before adding\)
mate level1-file.txt
clear
git status
git checkout -- level1-file.txt Back to the last commit
git status
mate level1-file.txt
```

Rename

Command Listing

```
pwd
cd projects/starter-web
ls
cd level1/level2/level3
pwd
clear
ls
git status
git mv level3--file.txt level3.txt using git with mv to move the file and add the changes
ls
git status
git commit -m "renaming level3 file"
cd ..
clear
pwd
ls
mv level2-file.txt level2.txt
ls
git status
git add -A Use -A to add all the files that are added and renamed
git status
git commit
clear
ls
git mv level2.txt 2.txt
ls
git status
git mv 2.txt level2.txt
ls
git status
clear
ls
git mv level2.txt level3
ls
```

```
cd level3
ls
git status
cd ..
git status
git commit
ls
cd level3
ls
mv level2.txt ..
ls
cd ..
pwd
ls
git status
git add -A
git status
git commit
clear
cd ..
pwd
# file renamed in Finder
git status
git add level1.txt
git add -u
git status
git commit
```

Deleting Files

Command Listing

```
pwd
cd projects/starter-web
ls
git status
clear
mate doomed.txt
ls
git status
git rm doomed.txt
rm doomed.txt
ls
git status
clear
git ls-files
ls
git rm newfile.txt
git status
git commit -m "Deleting new file"
git status
clear
git status
git ls-files
git rm hipster.txt
ls
git status
git reset HEAD hipster.txt
git status
git checkout -- hipster.txt
ls
git status
clear
ls
rm hipster.txt
```

```
ls
git status
git add -A
git status
git commit
git status
clear
ls
rm -rf level1
ls
git status
git add -A
git status
git commit -m "deleting level1 and all children"
git status
```

History

Command Listing

```
pwd
git help log
cd projects/starter-web
git status
clear
git log
git log --abbrev-commit
git log --oneline --graph --decorate
git log ae6f872...761b911
git log --since="3 days ago"
clear
ls
git log -- hipster.txt
clear
git log --follow -- level1/level2/level2.txt
clear
git show b1967804190e199d4c753e2c5e59d09ce65842d6
clear
```

Git Alias

Command Listing

```
pwd
cd projects/starter-web
ls
git status
clear
git log --all --graph --decorate --oneline
git hist
git config --global alias.hist "log --all --graph --decorate --oneline"
git hist
clear
mate ~/.gitconfig
git hist
mate ~/.gitconfig
git hist
clear
```

Git Configuration File (~/.gitconfig) snip

```
[alias]
```

```
hist = log --all --graph --decorate --oneline
```

Git Alias

Command Listing

```
pwd
cd projects/starter-web
ls
git status
ls -al
mate .gitignore
ls -al
git status
git add .gitignore
git status
git commit
git status
ls
mate access.log
ls
git status
git .gitignore
clear
git status
mkdir log
mv access.log log
ls
cd log
ll
cp access.log access.2014-11-04
ll
cd ..
git status
mate .gitignore
git status
git commit -am "Excluding log file directory"
git status
```

Git Basics Cleanup

Command Listing

```
pwd
cd projects/starter-web
ls
git status
clear
git pull origin master
git push origin master
```