Git Cheat Sheet

Init existing repository

```
mkdir NewDir && cd NewDir && git init
```

```
git commit -m 'first commit'
```

```
git remote add origin git@github.com:Treehopper/Git-Cheat-Sheet.git
```

```
git push --set-upstream origin master
```

Clone existing repository

```
git clone git@github.com:Treehopper/Git-Cheat-Sheet.git
```

Import from SVN into existing repository

```
git svn clone --shared SVN_REPO_URL LOCAL_DIR && cd LOCAL_DIR
```

```
git remote add origin git@github.com:GITHUB_USERNAME/REPO_NAME.git
```

```
git push origin master
```

```
git svn --authors-file=path/to/authors_file clone SVN_REPO_URL LOCAL_DIR
```

Basics

add file	`git add foobar.txt`
show changes	`git log && git diff`
commit changes	`git commit <file> -m "message"`
commit all changes	`git commit --all -m "message`
transmit changes	`git push`
receive changes	`git pull #(or git fetch)`

~/.gitconfig

```
[alias]
```

st	`= status . --untracked-files=no --short`
ci	`= commit`
ls	`= ls-files`
lg	`= --oneline --decorate=no --stat --graph --date-order`
lol	`= log --graph --decorate --oneline`

Modifications

real short status	`git status -uno -s`
concise logging info	`git log -9 --oneline --color=always --decorate=no --stat --graph --date-order`
what happened during my vacation	`git log --since=2weeks`
push local changes on a stack and return to last committed state for now	`git stash && ... && git pop`

Undo/Change Commits

revert committed changes	`git revert HEAD "foobar.txt"` `git checkout HEAD foobar.txt`
check out latest version	`git checkout HEAD foobar.txt`
change latest commit	`git commit --amend`

Looking in the Past

what did a file look like 9 commits ago	`git show HEAD~9:.muttrc`
what changed between two commits	`git diff HEAD~2 HEAD~3`

Branching and Merging

Branch off new Release Branch	`git checkout -b release mainline`
Merge Release Branch Changes back	`git checkout mainline` `git merge release_x`
replay commits to Feature Branch onto the Mainline	`git rebase featurex mainline`
mark the nearest commit whose message begins with a keyword as stable	`git tag "stable" :/"fixed"`