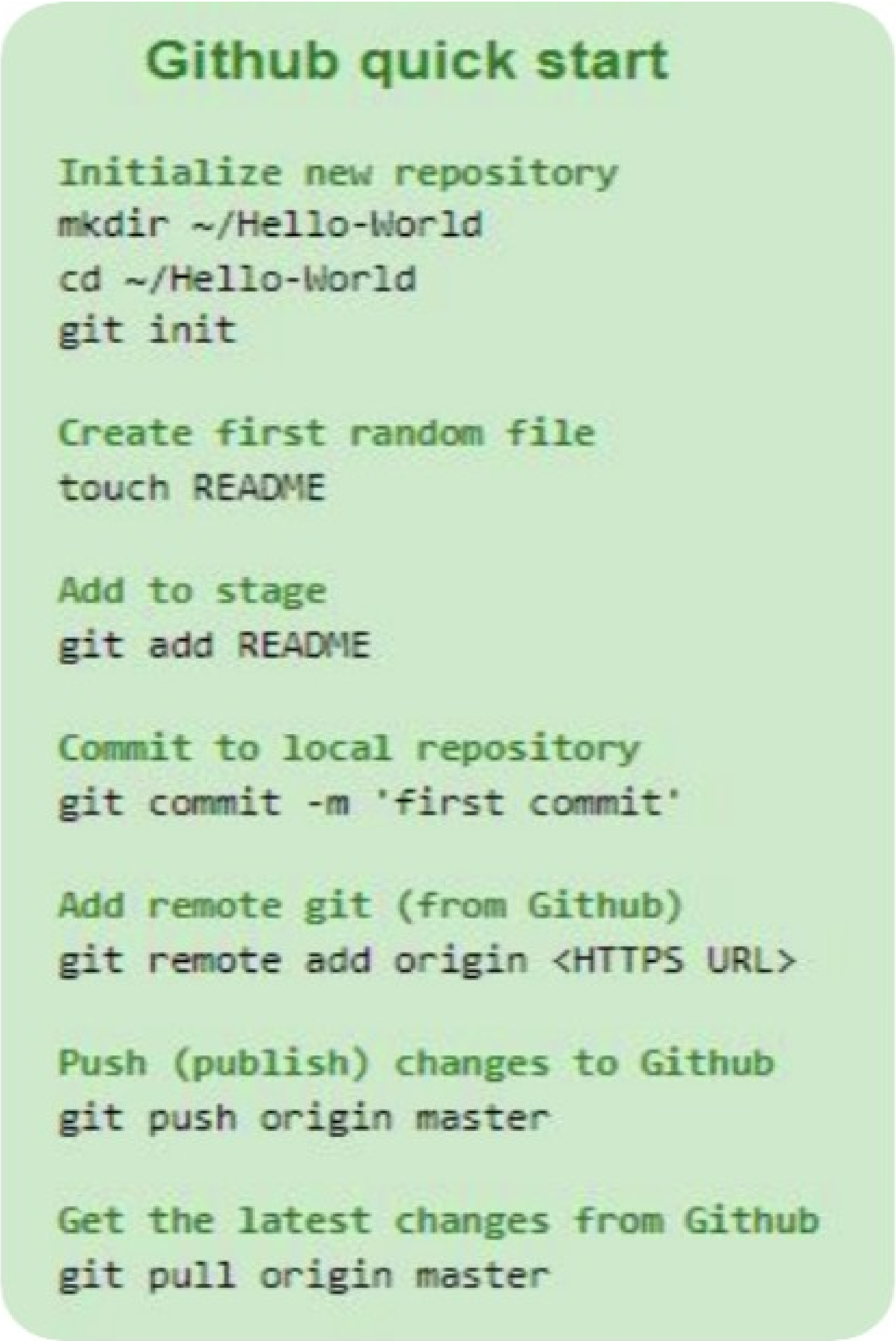


Github quick start

Initialize new repository

```
mkdir ~/Hello-World
cd ~/Hello-World
git init
```

Create first random file

```
touch README
```

Add to stage

```
git add README
```

Commit to local repository

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

Add remote git (from Github)

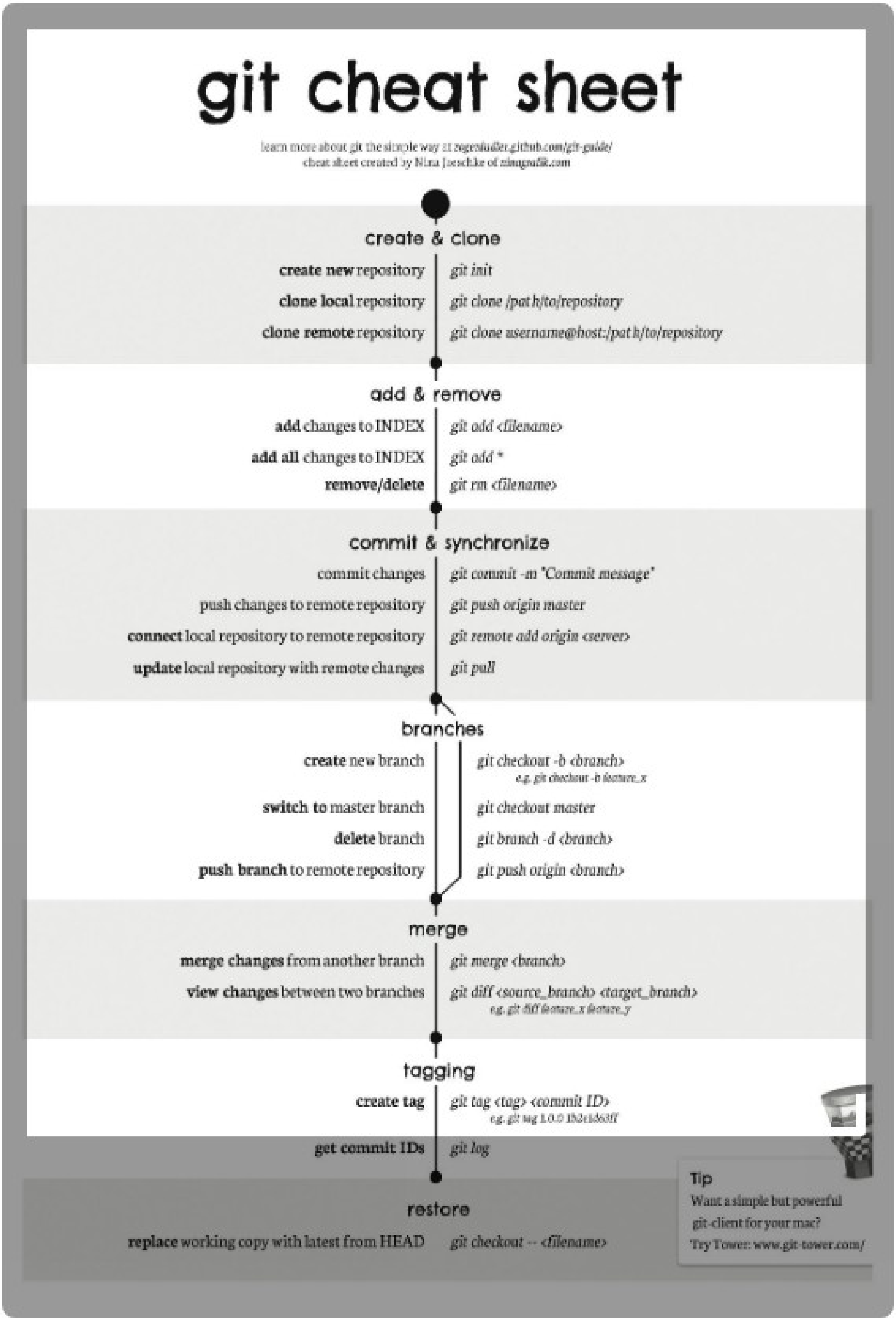
```
git remote add origin <HTTPS URL>
```

Push (publish) changes to Github

```
git push origin master
```

Get the latest changes from Github

```
git pull origin master
```



git cheat sheet

learn more about git the simple way at *rogerdudler.github.com/git-guide/*
cheat sheet created by Nina Jaeschke of *ninagrafik.com*

create & clone

create new repository	*git init*
clone local repository	*git clone /path/to/repository*
clone remote repository	*git clone username@host:/path/to/repository*

add & remove

add changes to INDEX	*git add <filename>*
add all changes to INDEX	*git add **
remove/delete	*git rm <filename>*

commit & synchronize

commit changes	*git commit -m "Commit message"*
push changes to remote repository	*git push origin master*
connect local repository to remote repository	*git remote add origin <server>*
update local repository with remote changes	*git pull*

branches

create new branch	*git checkout -b <branch>* *e.g. git checkout -b feature_x*
switch to master branch	*git checkout master*
delete branch	*git branch -d <branch>*
push branch to remote repository	*git push origin <branch>*

merge

merge changes from another branch	*git merge <branch>*
view changes between two branches	*git diff <source_branch> <target_branch>* *e.g. git diff feature_x feature_y*

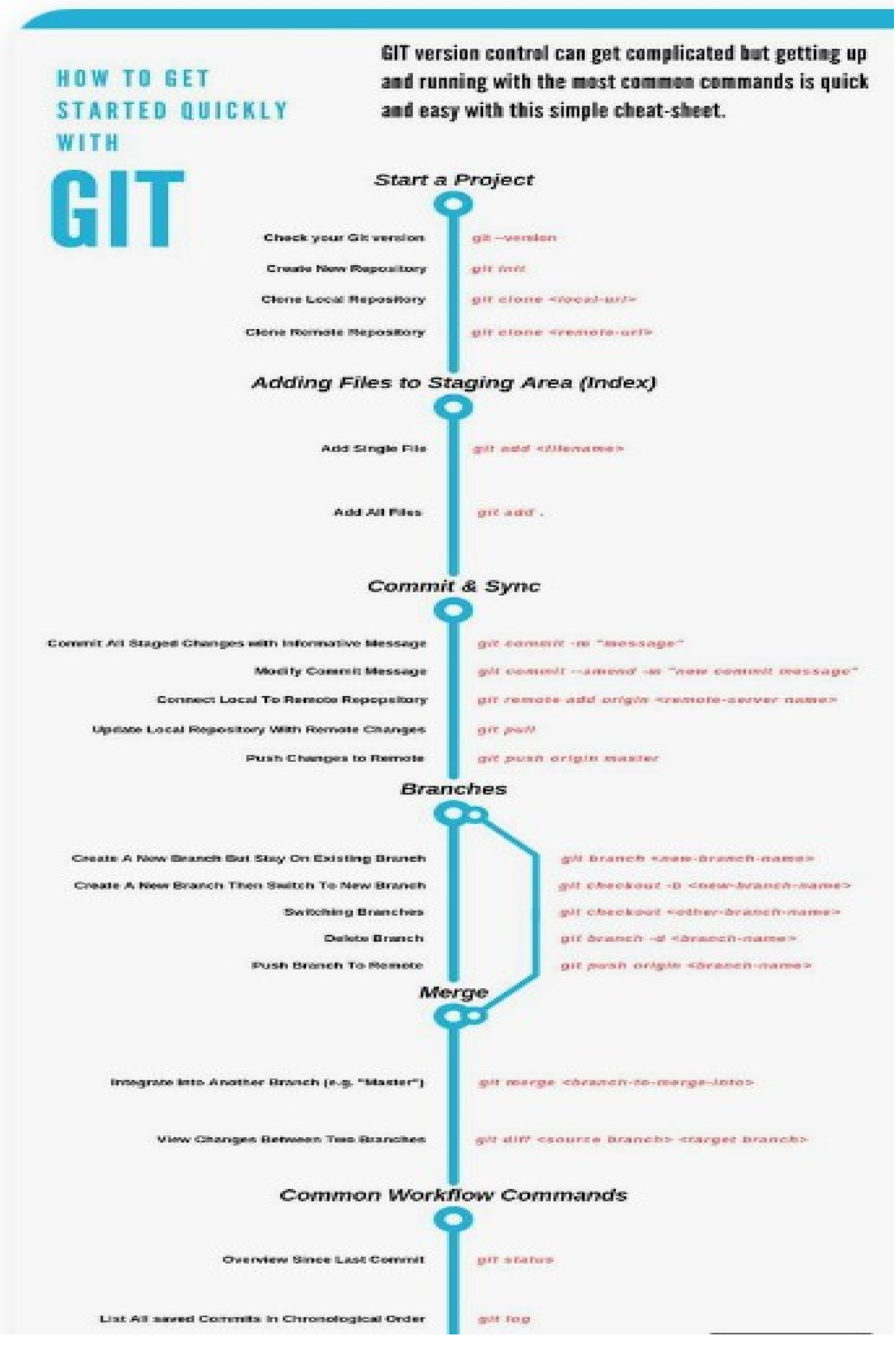
tagging

create tag	*git tag <tag> <commit ID>* *e.g. git tag 1.0.0 1b2e1d63ff*
get commit IDs	*git log*

restore

replace working copy with latest from HEAD	*git checkout -- <filename>*

Tip
Want a simple but powerful
git-client for your mac?
Try Tower: www.git-tower.com/



HOW TO GET STARTED QUICKLY WITH GIT

GIT version control can get complicated but getting up and running with the most common commands is quick and easy with this simple cheat-sheet.

Start a Project

Check your Git version	`git --version`
Create New Repository	`git init`
Clone Local Repository	`git clone <local-url/>`
Clone Remote Repository	`git clone <remote-url>`

Adding Files to Staging Area (Index)

Add Single File	`git add <filename>`
Add All Files	`git add .`

Commit & Sync

Commit All Staged Changes with Informative Message	`git commit -m "message"`
Modify Commit Message	`git commit --amend -m "new commit message"`
Connect Local To Remote Repopsitory	`git remote add origin <remote-server name>`
Update Local Repository With Remote Changes	`git pull`
Push Changes to Remote	`git push origin master`

Branches

Create A New Branch But Stay On Existing Branch	`git branch <new-branch-name>`
Create A New Branch Then Switch To New Branch	`git checkout -b <new-branch-name>`
Switching Branches	`git checkout <other-branch-name>`
Delete Branch	`git branch -d <branch-name>`
Push Branch To Remote	`git push origin <branch-name>`

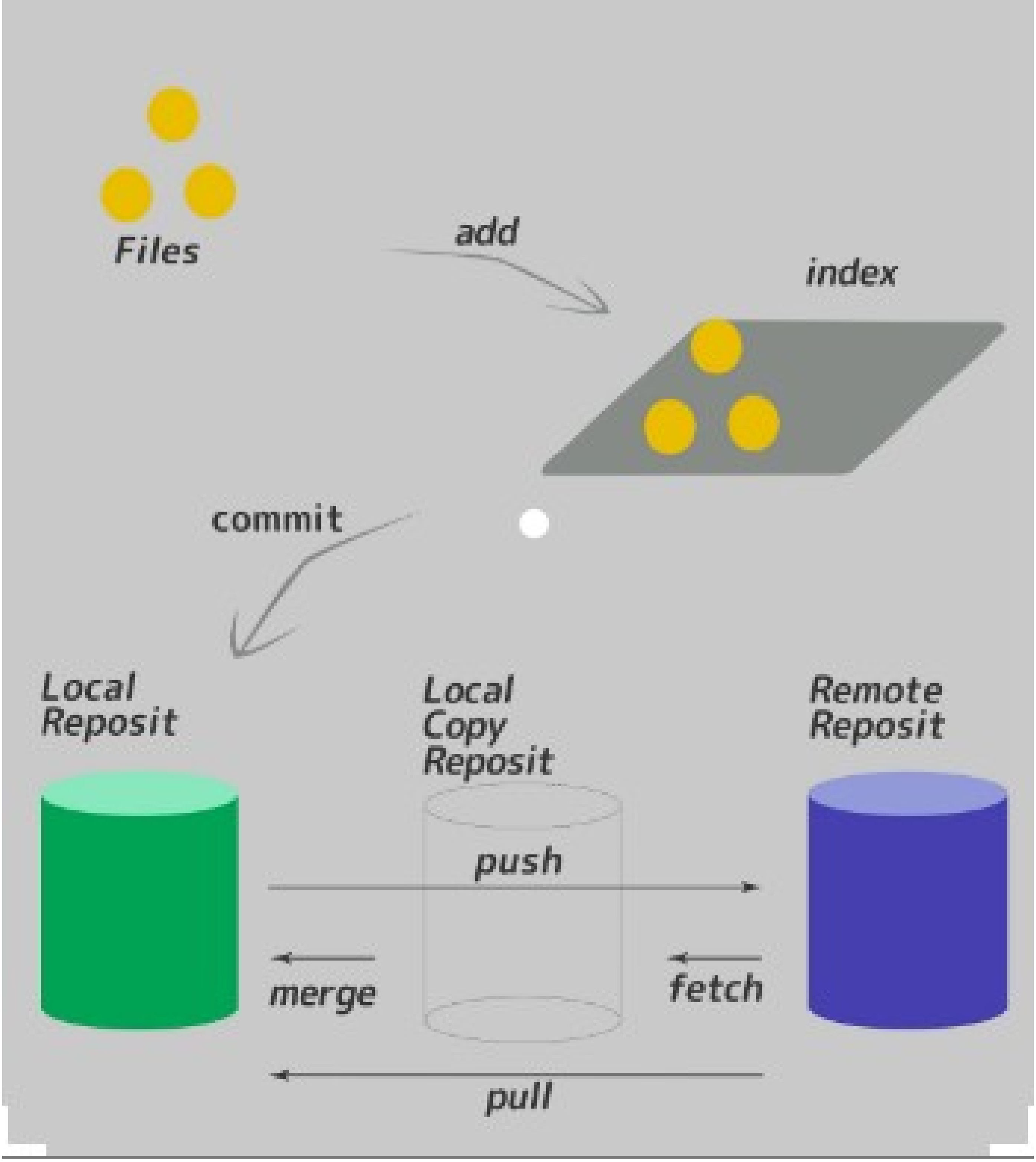
Merge

Integrate into Another Branch (e.g. "Master")	`git merge <branch-to-merge-into>`
View Changes Between Two Branches	`git diff <source branch> <target branch>`

Common Workflow Commands

Overview Since Last Commit	`git status`
List All saved Commits In Chronological Order	`git log`

Commit Cycle


Files
add
index
commit
Local
Reposit
Local
Copy
Reposit
Remote
Reposit
push
merge
fetch
pull