



Git Basics



Git Setup

- Setup properties globally
 - `$git config --global user.name "Kalman Hazins"`
 - `$git config --global user.email myemail@whteva.com`
 - Verify that an option has been set
 - `$git config <option>`
 - For example, `$git config user.name`
- ```
C:\Users\HAZINK1>git config user.name
Kalman Hazins
```
- Getting help on any Git command
    - `$git help <command>`

# Initializing a repo

Where do I get a repo from?

## 1. Create a new repo

- `$cd some_dir`
- `$git init`
- (Possibly create a `.gitignore` file)
- `$git add .`  
(. Adds the entire current directory with subdirectories)
- `$git commit -m "Initial commit"`

# Cloning a repository

## 2. Clone an existing repo

```
$git clone https://repourl.git
```

Many transfer protocols available

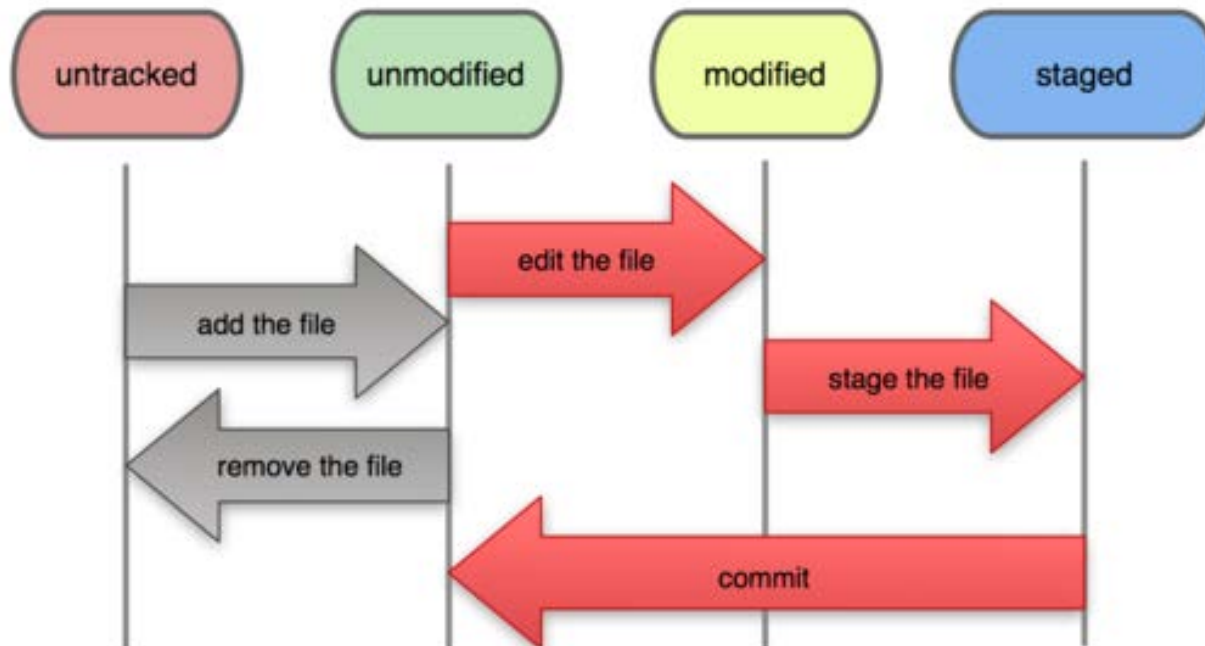
- https:
- git:
- file:

# git status

- `$git status`

– Provides the current status of your repo

```
$ git status
On branch master
nothing to commit (working directory clean)
```



# git add

- `$git add <file/dir>`
  - Add untracked file(s) to be tracked or
  - Add a tracked file to the staging area
  - **Mods made to the file after “git add” - need to be “git-added” again! (Even if you did not commit yet!)**

```
$ vim benchmarks.rb
$ git status
On branch master
Changes to be committed:
(use "git reset HEAD <file>..." to unstage)
#
new file: README
modified: benchmarks.rb
#
Changed but not updated:
(use "git add <file>..." to update what will be committed)
#
modified: benchmarks.rb
#
```

# git diff

- `$git diff`
  - Shows the difference between working directory and staging area
- `$git diff --staged`
  - Shows the changes between HEAD (latest commit on current branch) and staging area
- `$git diff HEAD`
  - Shows the deltas between HEAD and working dir

# git commit

- `$git commit`
  - Commits your changes to the repo
    - Prompts for a commit message in an editor
  - Better, just use the `-m` (message) option
    - `$git commit -m "Your msg here"`



# Skipping the staging area

- (The moment you've been waiting for!!! 😊)
- To skip the staging area - just use `-a` flag
  - After initially adding the file
- Either `-a` `-m` or `-am` will do the trick

```
$ git status
On branch master
#
Changed but not updated:
#
modified: benchmarks.rb
#
$ git commit -a -m 'added new benchmarks'
[master 83e38c7] added new benchmarks
1 files changed, 5 insertions(+), 0 deletions(-)
```

## git rm / git mv - Remove/Rename

- `$git rm <file>`
  - Removes the file from being tracked and your working directory
  - Stages the change for the next commit
- `$git mv <file1> <file2>`
  - Removes <file1> from being tracked
  - Renames it to <file2>
  - Stages the change for the next commit

# git log

- `$git log` – view commit history
- Many nice options – need to play with it to see what you like
  - `-<n>` – specify number of commits back to see
  - `-p` – generate patch
  - `--stat` – shows a diffstat
  - `--oneline` – one line history of commits
  - `--graph`

# Unstaging a staged file

```
$ git add .
$ git status
On branch master
Changes to be committed:
(use "git reset HEAD <file>..." to unstage)
#
modified: README.txt
modified: benchmarks.rb
#
```

```
$ git reset HEAD benchmarks.rb
benchmarks.rb: locally modified
$ git status
On branch master
Changes to be committed:
(use "git reset HEAD <file>..." to unstage)
#
modified: README.txt
#
Changed but not updated:
(use "git add <file>..." to update what will be committed)
(use "git checkout -- <file>..." to discard changes in working directory)
#
modified: benchmarks.rb
#
```

# Going back in time

- Before committing
  - `$git checkout .`
    - Re-checkout all tracked files overwriting local changes
  - `$git checkout -- <file>`
    - Re-checkout just one specific file
- After committing
  - `$git revert HEAD`
    - Reverts the most recent commit

# git tag - Bookmarks

```
$ git tag -a v1.4 -m 'my version 1.4'
$ git tag
v0.1
v1.3
v1.4
```

```
$ git show v1.4
tag v1.4
Tagger: Scott Chacon <schacon@gee-mail.com>
Date: Mon Feb 9 14:45:11 2009 -0800

my version 1.4
commit 15027957951b64cf874c3557a0f3547bd83b3ff6
Merge: 4a447f7... a6b4c97...
Author: Scott Chacon <schacon@gee-mail.com>
Date: Sun Feb 8 19:02:46 2009 -0800

Merge branch 'experiment'
```