Installing Git

Git is actually installed on MacOS, but we'll be reinstalling it so that we'll have the newest version:

1. go to https://git-scm.com/downloads
2. download the software for Mac
3. install Git choosing all of the default options

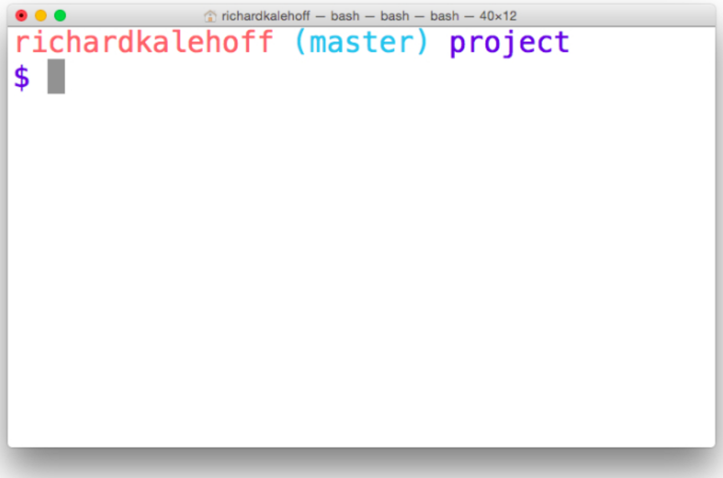
Once everything is installed, you should be able to run `git` on the command line. If it displays the usage information, then you're good to go!

If you run into any issues, please remember to take advantage of Knowledge and the community in your Study Group.

Configuring Mac's Terminal

We're about to configure the Terminal to display helpful information when in a directory that's under version control. *This is an optional step!* You do not need to re-configure your terminal for Git to work. You can complete the entire course without reconfiguring it. However, reconfiguring the Terminal makes it significantly easier to use.

If you choose to configure your Terminal, here's what it should look like when you're finished.



```
richardkalehoff (master) project
$
```

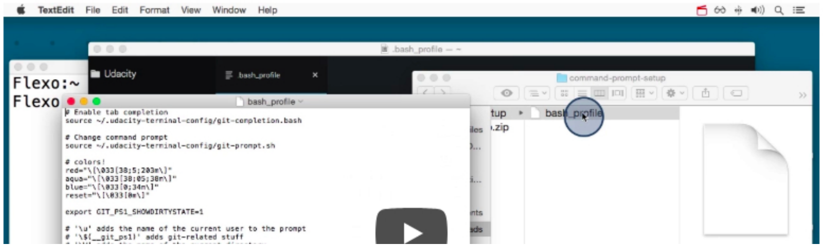
The terminal application on MacOS. The terminal has been configured to display version control information.

Configuration Steps

To configure the terminal, we'll perform the following steps:

1. download the zipped file
2. move the directory `udacity-terminal-config` to your home directory and name it `.udacity-terminal-config` (there's a dot at the front, now!)
3. move the `bash_profile` file to your home directory and name it `.bash_profile` (there's a dot at the front, now!)
   - if you *already* have a `.bash_profile` file in your home directory, transfer the content from the downloaded `bash_profile` to your existing `.bash_profile`

Download the zipped file in the Resources pane to get started.



```
Flexo:~
Flexo
```

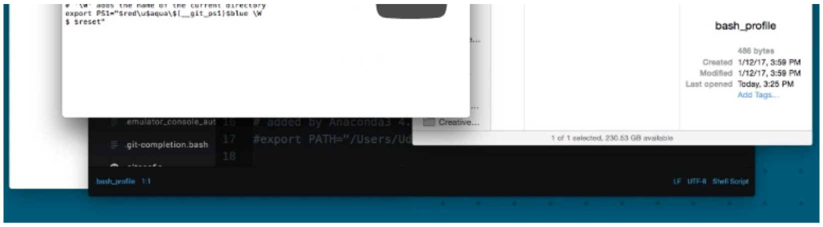
```
# Enable tab completion
source ~/.udacity-terminal-config/git-completion.bash

# Change command prompt
source ~/.udacity-terminal-config/git-prompt.sh

# colors!
red="\[\033[38;5;203m\]"
aqua="\[\033[38;05;38m\]"
blue="\[\033[0;34m\]"
reset="\[\033[0m\]"

export GIT_PS1_SHOWDIRTYSTATE=1

# '\u' adds the name of the current user to the prompt
# '\$(__git_ps1)' adds git-related stuff
```



```
# `\W` adds the name of the current directory
export PS1="$red\u$aqua\$(__git_ps1)$blue \W
$ $reset"
```

```
# added by Anaconda3 4.
#export PATH="/Users/Ud
```

First Time Git Configuration

Before you can start using Git, you need to configure it. Run each of the following lines on the command line to make sure everything is set up.

```
# sets up Git with your name
git config --global user.name "<Your-Full-Name>"

# sets up Git with your email
git config --global user.email "<your-email-address>"

# makes sure that Git output is colored
git config --global color.ui auto

# displays the original state in a conflict
git config --global merge.conflictstyle diff3

git config --list
```

Git & Code Editor

The last step of configuration is to get Git working with your code editor. Below are three of the most popular code editors. If you use a different editor, then do a quick search on Google for "associate X text editor with Git" (replace the X with the name of your code editor).

Atom Editor Setup

```
git config --global core.editor "atom --wait"
```

Sublime Text Setup

```
git config --global core.editor "'/Applications/Sublime Text 2.app/Contents/Sh
```

VSCode Setup

```
git config --global core.editor "code --wait"
```

If you have any questions, post them on the forum.

Everything Is All Set Up

☑ I've installed Git

☑ I've configured Git with my username

☑ I've configured Git with my email

☑ I've configured Git to use my chosen editor

Supporting Materials

udacity-terminal-config.zip

NEXT