**Terminal Paths**

class

cd ~/Sites/WDI/wdidc2/w12/d02/students/Martin

my rails

cd ~/dev/wdi/w11/d02

next-books

cd ~/dev/wdi/w06/next\_books

mini-project

cd ~/dev/wdi/w09/MiniProject/Top-Tomato

app\_etizers

cd ~/dev/wdi/w10/app\_etizer/mappetizer

git\_coffee

cd ~/dev/wdi/w11/git\_coffee

**Submit Homework**

{

“completeness”:”4”,

“comfortability”:”4”

}

**rSpec**

I created a .rspec file with:

echo ‘—format documentation’ > ~/.rspec

—color

—warning

—require spec\_helper

put in .rspec in Rails app

--color

--require spec\_helper

to run rSpec, cd into the app folder (above the spec folder) and

$ rspec

**seed**

require\_relative “files”

**LESS**

control D: page down

control U: page up

**SQL**

psql -d jobbed\_out -f schema.sql

cd ~/dev/wdi/w09/d02

SELECT name, (points/games) AS points\_per\_game FROM players

ORDER BY points\_per\_games;

c\ nba\_db

**Git Clone**

to pull another repository, go to my ~Sites/dev/

go to web page with what I want to fork

select the SSH under the HTTPS, so it does not ask for my key. this will update what is on my clipboard. Then copy the SSH

$ git clone (paste which looks like git@github.com:mobjohnson/coffee.git)

cd into the app

**Git Adding a Remote as my own**

go to a github repository

download zip

create the folder in dev/w05/d03

move the .zip into dev/w05/d03

click it to unzip

change into the new app folder

$ git init

$ git add .

$ git commit -m ‘Initial commit’

now by master branch shows up!

go to my github

select the + button top right to create a new repository

give it a name and description

create new repo

copy ssh

$ git remote add origin (url from new github repo)

$ git push -u origin master

now can just git push!

**Git starting app as my own**

rails new name\_app -d postgresql -T

cd into name\_app

$ git init

$ git add .

$ git commit -m ‘Initial commit’

now my master branch shows up!

go to my github

select the + button top right to create a new repository

give it a name and description

create new repo

copy ssh

$ git remote add origin (url from new github repo)

$ git push -u origin master

now can just git push!

**Deploying to Heroku**

$ heroku create

$ heroku create heroku master

$ heroku run rake db:migrate

$ heroku run rake db:seed

go to <http://intense-sierra-2061.herokuapp.com/>

in irb: reload!

**JavaScript**

can run in the console, after you copy an html file into the browser

can also use $ node, but can not use it for browser interaction like prompts

**Git: to submit a clean pull request**

1. $ git push origin w08d01E
2. go to github, select a pull request. is at top of the page

**Git: to pull each feature branch into my master branch**

1. $ git checkout master
2. $ git merge w08d01C

repeat for each time I had to do a git fetch

**Git: When instructor says git pull upstream master**

1. Make sure I am in a clean working directory.
2. $ git fetch upstream master

this does not affect my working directory, it only updates the local reference to the remote branch

3) $ git checkout -b w12d02A upstream/master

upstream/master is the class repository

master is now my dirty branch

4) copy folder from d01 down to my students/Martin/ directory

5) $ git add .

6) $ git commit -m ‘Start new assignment’

5) cd to directory and then: subl .

5) work on the assignment

**Mini Project**

**Git: using Rebase instead of pull requests**

1. create a git repository
2. clone repository:

**Git Clone**

to pull another repository, go to my local folder ~Sites/dev/

go to GitHub web page with repository I want to fork

select the SSH under the HTTPS, so it does not ask for my key. this will update what is on my clipboard. Then copy the SSH

$ git clone (paste which looks like git@github.com:mobjohnson/coffee.git)

cd into the app

1. create a feature branch:
2. $ git checkout -b branch\_name master
3. make commits and want to merge the feature branch
4. $ git checkout master
5. $ git pull origin master
6. $ git checkout branch\_name
7. $ git rebase master
8. $ git checkout master
9. $ git merge branch\_name
10. $ git push
11. notify team members that they need to: $ git rebase master