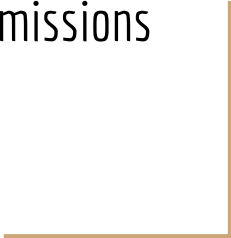




Assignment 0: Office Hour

Github / W266 homework submissions
Jan. 12, 2018



Office hour agenda:

- Grading policy -- a re-statement from live class #1
- How to submit (submit.sh script!!)
- What to do if things don't work.
- Announcing: Next TA OH

Assignments

- Start early... ..and submit often.
- Start early.... ..and submit often.
- Start early.... ..and submit often.
- Also... start early.... ..and submit often.

This course is a ***lot*** of work.

Submitting

Submitting

- **Do!** Use the submit script.
- **Don't!** Pull requests, merge to master, etc
- Early and often. We grade the last submit before the hard deadline.

What is the “hard deadline”? 48 hours after the assignment deadline.

If you don't submit by hard deadline => You will get a zero.

Don't be this person!

Late Days

Don't submit by soft deadline?

Costs you a late day (or 10%age points) for any part of each 24 hours.

Hard deadline

Still 48 hours after the regular deadline, regardless of late days left.

“Work is busy, I need an extension”

Give it to yourself! Don't waste time telling us.

We grade the last submit before the hard deadline and calculate late days.

Submitting homework -- use the submit.sh script

...suppose you're in the homework directory:

```
$ ../submit.sh -a 0 -u drewplant-MIDS
```

What does this do??

the submit.sh script

...suppose you're in the homework directory:

```
$ ../submit.sh -u <gituser> -a <x> # where x = NumAssignment
```

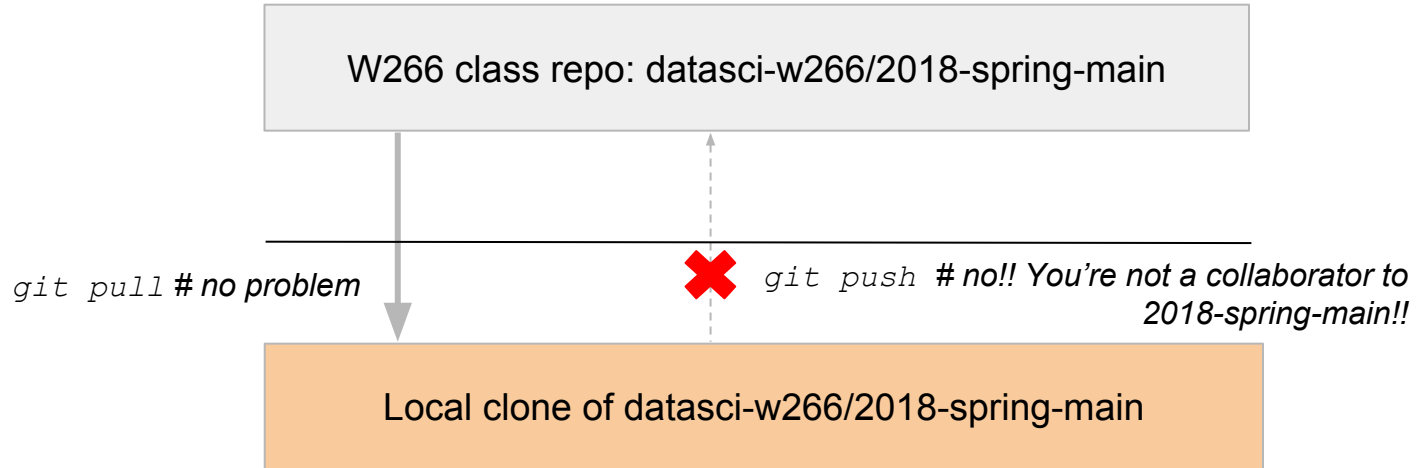
Example: `submit.sh -a 0 -u drewplant-MIDS # for assignment a0`

What does this do??

the submit.sh script - motivation

```
$ ../submit.sh -a 0
```

Often, you're doing work in your own clone of `datasci-w266/2018-spring-main`



the submit.sh script - what's it doing?

```
$ ../submit.sh -a 0 -u username
```

W266 class repo:
datasci-w266/2018-spring-main

Your personal remote submission repo:
datasci-w266/2018-spring-assignment-<username>

git pull # no problem

You are the owner of 2018-spring-assignment-username !!

Local clone of datasci-w266/2018-spring-main

3.git push your entire repo
including committed files to branch:
a0-submit.

1. Check if any files are different from HEAD and commit

```
Note: this will submit your whole repository to branch  
'a0-submit'  
== Warning! You have uncommitted changes in this repository:  
assignment/a0/username | 2 +-  
 1 file changed, 1 insertion(+), 1 deletion(-)  
Commit before submitting? (submit.sh will only push committed  
changes)  
1) Yes  
2) No  
3) (cancel)  
#? 1
```

Files in local dir:

a0.ipynb

Username

other files in repo

2.You choose method of pushing:

```
== Select GitHub access protocol ==  
HTTPS is default, but SSH may be needed if you  
use two-factor auth.  
1) HTTPS  
2) SSH  
3) (cancel)  
#? 2
```

What if you encounter a merge problem?

```
$ ../submit.sh -a 0 -u username
```

Your remote class repo:
datasci-w266/2018-spring-main

Local clone of datasci-w266/2018-spring-main

Files in local dir:

a0.ipynb

Username

other files in repo

uh-oh:



! [rejected] HEAD -> a0-submit (non-fast-forward)
error: failed to push some refs to
'git@github.com:datasci-w266/2018-spring-assignment-drewplant-MIDS.git'
hint: Updates were rejected because a pushed branch tip is behind its remote
hint: counterpart. Check out this branch and integrate the remote changes
hint: (e.g. 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

Your remote submission repo:
datasci-w266/2018-spring-assignment-<username>

Local clone of
datasci-w266/2018-spring-assignment-<username>

Files in local dir:

a0.ipynb

Username

other files in repo

4. **../submit -a 0 -u username**

3. **Copy your work to this clone**

1. **Clone your submission repo:**

```
git clone https://github.com/datasci-w266/2018-spring-assignment-<uname>.git
```

2. **Checkout submission branch:**

```
git checkout a0-submit
```

Next Office hour:

- “Regular time”: Tuesdays at noon (12pm) PST
- Most days. We’ll announce if OH is skipped for a particular week.
(such as Spring Break, Immersion, etc.)