

Lecture 7

Intermediate Git: Stash, Revert, and Searching History

**Sign in on the
attendance
sheet!**



Git'stache

Scenario: you want to switch branches, but you have uncommitted changes

```
:( Andrew@Gengar ~/temp
04:08 PM (coolfeature)$ git checkout master
error: Your local changes to the following files would be overwritten by checkout:
    A
Please, commit your changes or stash them before you can switch branches.
Aborting
```

What if you don't want to commit?

git stash

Example use:

```
git stash
```

- Makes a “pseudo-commit” and puts it on a stack of stashed pseudo-commit.
- Message for stash is “WIP on <*branchname*>...”
- Use `git stash save <message>` to store stashes with better messages

git stash list

Example use:

```
git stash list
```

- Lists all stashed changes on the stash stack.

```
git stash apply (stash@{<depth>})
```

Example use:

```
git stash apply stash@{2}
```

- Reapplies the stashed change at the specified depth, if given.
- Depth is just another way of choosing from a list of saved stashes

git stash pop

Example use:

```
git stash pop
```

- Reapplies the top stashed change and removes it from the stash stack.

```
git stash drop (stash@{<depth>})
```

Example use:

```
git stash drop stash@{2}
```

- Removes the stashed change at the specified depth, if given.

```
git stash show (-p) (stash@{<depth>})
```

Example use:

```
git stash show stash@{2}
```

- Show details about the stashed change at the specified depth, if given.


```
git revert <commit hash> ...
```

Example use:

```
git revert HEAD (putting HEAD there is optional)
```


- Makes a new commit that “undoes” the commit at HEAD
- For when you made a mistake and want to completely reverse a commit
- Can cause a conflict (just like a merge) that you have to manually resolve
- Cause use with a list of commits


Spelunking with Git (and Github)






File History on Github

Branch: master ▾ Autolab / app / models / annotation.rb Find file Copy path

 icanberk Annotations for PDF submission ce723aa on Jul 8, 2015

2 contributors 


Executable File | 26 lines (23 sloc) | 564 Bytes Raw Blame **History**   

```
1 ##
2 # Annotations are Submission and Problem specific.
- - - - -
```











- Same as `git log <filename>`

git blame <filename>

- Shows which commit last modified a specific line

Autolab / app / models / annotation.rb 

100755 | 26 lines (23 sloc) | 564 Bytes

 Rubocop'd A models	2 years ago		1 ##
			2 # Annotations are Submission and Problem specific.
			3 # Currently, they are just text, but it would be nice if they could be used in
			4 # score calculations in the future.
			5 #
 Initial commit for open source Autolab	2 years ago		6 class Annotation < ActiveRecord::Base
			7 belongs_to :submission
			8 belongs_to :problem
			9
 Annotations for PDF submission	2 years ago		10 validates :comment, :filename, :submission_id, presence: true
 Initial commit for open source Autolab	2 years ago		11
 Fixes for viewFeedback	2 years ago		12 def as_text
 Ran rubocop --autocorrect	2 years ago		13 if value
			14 if problem

git log -G "regex to search for"

Searches for commits whose diff contains the text

```
$ git log --pretty=short -G "validates :comment"  
commit ce723aad8a5ab59994f8f55110d7de9c404e2fc8  
Author: Ilder Canberk <iltercanberk@gmail.com>  
  
    Annotations for PDF submission  
  
commit 1494872c7f780a406d5aad20087b2b9840fef6  
Author: Daniel Bucci <dlbucci@superftc.com>  
  
    Rubocop'd A models
```



(called pickaxe diff in the git documentation)

Activity!

- Go to the Autolab project on Github
- Take a look at line #20 of the file `app/views/devise/sessions/new.html.erb`
- In which commit was the text on that line initially added to the repository?
- To which file was the text moved to in April 2015?
- When was the text moved back to the original file? By whom? Why?

