INTRO TO VERSION CONTROL USING GIT

JEREMY FLORES & LINDSEY BIEDA

INTRODUCTIONS

PLEASE ASK QUESTIONS.

GOALS FOR THE CLASS

- Basic understanding of what version control is
- Basic understanding of git is and how to use it
- Knowledge of github and what it provides us

VERSION CONTROL SYSTEM

A tool that assists in the management of changes to documents.

WHAT DOES THAT MEAN?

- You can make changes to documents and save those changes.
- You can undo changes
- You can include changes from other people*
- You can multiple change-sets simultaniously*

NOT JUST FOR PROGRAMMING

Anytime you are working on any set of documents that you are making changes to and want to save the state of version control is extremely useful.

GIT

A **distributed** version control system.

Distributed, in this case, basically means that we can have the same thing existing in multiple locations

GITHUB

A web service for programming projects that use Git.

github
SOCIAL CODING

TOOL CHECK

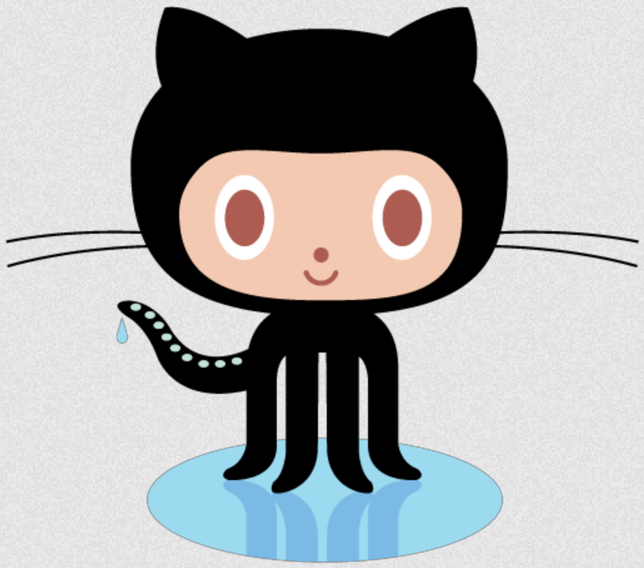
GITHUB

- If you haven't already create a GitHub account at github.com
- Login to your account
- Set your avatar at gravatar.com
- View your GitHub profile page: http://github.com/**your_username**

note: for gravatar use the same email your signed up for github with.

GITHUB FOR MAC/WINDOWS

A program that lets us **easily** use git and github.

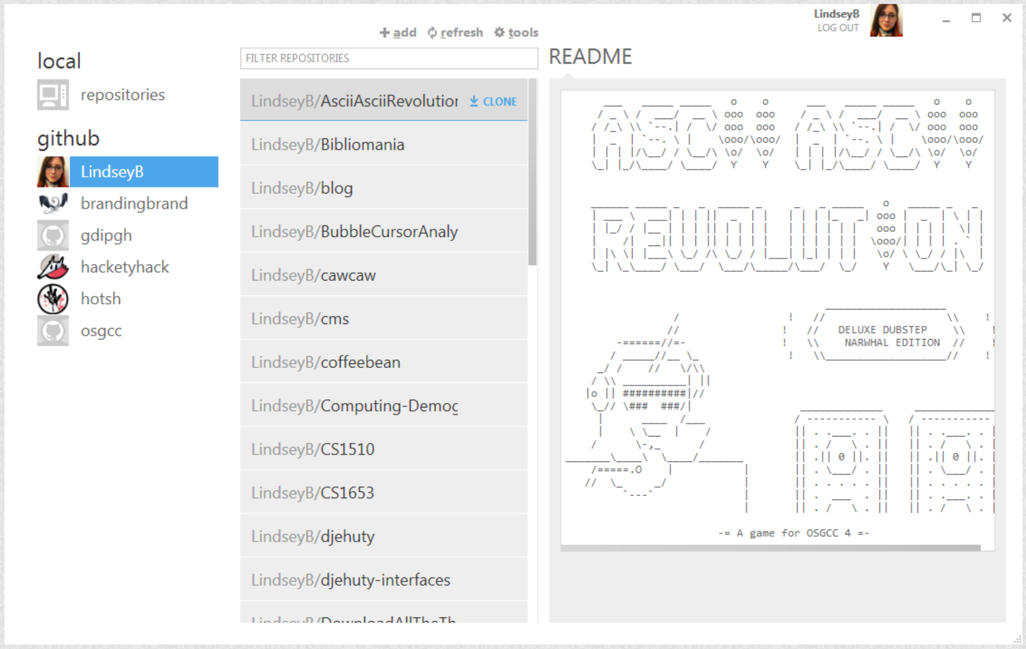


TOOL CHECK

GITHUB FOR MAC/WINDOWS

- If you haven't already download GitHub for Mac / Windows and install
- Open GitHub for Mac/Windows
- Login to your GitHub Account

GITHUB APP ANATOMY



add refresh tools

LindseyB
LOG OUT

local
- repositories

github
- LindseyB
- brandingbrand
- gdipgh
- hacketyhack
- hotsh
- osgcc

FILTER REPOSITORIES

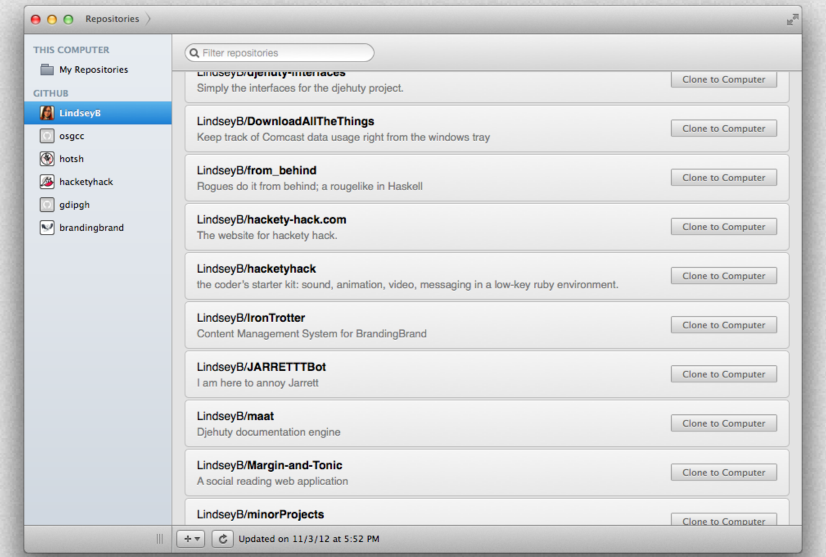
- LindseyB/AsciiAsciiRevolutior CLONE
- LindseyB/Bibliomania
- LindseyB/blog
- LindseyB/BubbleCursorAnaly
- LindseyB/cawcaw
- LindseyB/cms
- LindseyB/coffeebean
- LindseyB/Computing-Democ
- LindseyB/CS1510
- LindseyB/CS1653
- LindseyB/djehuty
- LindseyB/djehuty-interfaces

README

```
DELUXE DUBSTEP
NARWHAL EDITION

-= A game for OSGCC 4 =-
```

GITHUB APP ANATOMY



LindseyB/**djehuty-interfaces**
Simply the interfaces for the djehuty project.

LindseyB/**DownloadAllTheThings**
Keep track of Comcast data usage right from the windows tray

LindseyB/**from_behind**
Rogues do it from behind; a rougelike in Haskell

LindseyB/**hackety-hack.com**
The website for hackety hack.

LindseyB/**hacketyhack**
the coder's starter kit: sound, animation, video, messaging in a low-key ruby environment.

LindseyB/**IronTrotter**
Content Management System for BrandingBrand

LindseyB/**JARRETTTBot**
I am here to annoy Jarrett

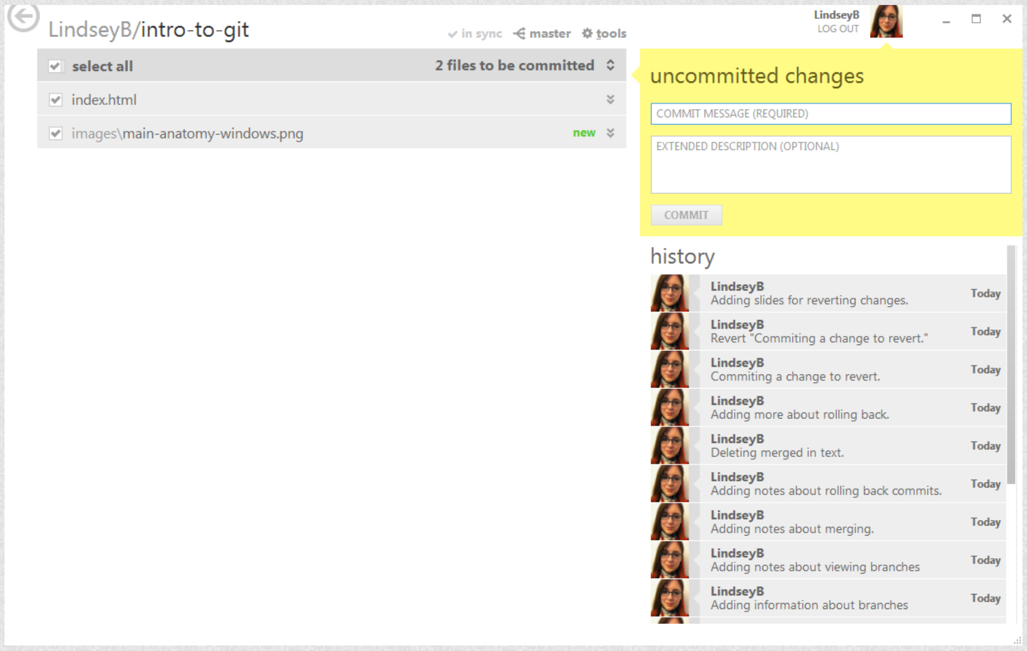
LindseyB/**maat**
Djehuty documentation engine

LindseyB/**Margin-and-Tonic**
A social reading web application

LindseyB/**minorProjects**

Updated on 11/3/12 at 5:52 PM

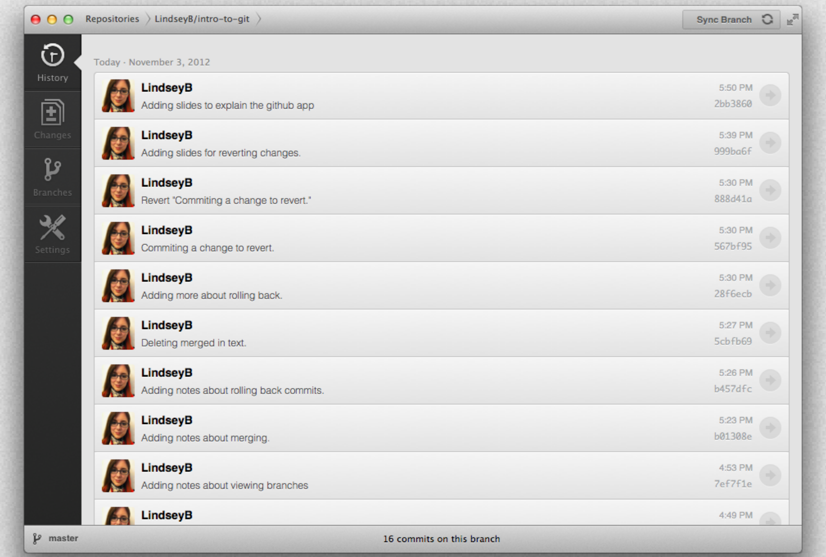
GITHUB APP ANATOMY



LindseyB/intro-to-git

LindseyB	Adding slides for reverting changes.	Today
LindseyB	Revert "Commiting a change to revert."	Today
LindseyB	Commiting a change to revert.	Today
LindseyB	Adding more about rolling back.	Today
LindseyB	Deleting merged in text.	Today
LindseyB	Adding notes about rolling back commits.	Today
LindseyB	Adding notes about merging.	Today
LindseyB	Adding notes about viewing branches	Today
LindseyB	Adding information about branches	Today

GITHUB APP ANATOMY



Today · November 3, 2012

LindseyB
Adding slides to explain the github app
5:50 PM
2bb3860

LindseyB
Adding slides for reverting changes.
5:39 PM
999ba6f

LindseyB
Revert "Commiting a change to revert."
5:30 PM
888d41a

LindseyB
Commiting a change to revert.
5:30 PM
567bf95

LindseyB
Adding more about rolling back.
5:30 PM
28f6ecb

LindseyB
Deleting merged in text.
5:27 PM
5cbfb69

LindseyB
Adding notes about rolling back commits.
5:26 PM
b457dfc

LindseyB
Adding notes about merging.
5:23 PM
b01308e

LindseyB
Adding notes about viewing branches
4:53 PM
7ef7f1e

LindseyB
4:49 PM

master

16 commits on this branch

REPOSITORY

WHERE OUR FILES LIVE

Also contains:

- A record of the changes made to those files
- Who made those changes
- When those changes were made

Also referred to as a repo for short.

ACTIVITY

OUR FIRST REPOSITORY

- Open the GitHub application
- Click "add" to create a new repo
- Give it a name and a description (e.g. "My first repo")

COMMIT

To save the state of a single or collection of changes.

HOW TO MAKE A COMMIT WITH GIT

Whenever you make a change inside the folder where your repo is the GitHub app notices and will ask for a commit message in order to save the changes.

COMMIT MESSAGE

A note that describes what the commit is about.

BAD COMMIT MESSAGES

- Whoops
- Fixed a thing
- I have no idea
- Everything works now
- I am so angry at the universe right now

BETTER COMMIT MESSAGES

- Added the missing read-me file
- Fix: Extranous semi-colon removed
- Moved the cat-pix folder inside the images folder
- Resolved the problem where content would not display
- Added a hack in the CSS to fix the bug with IE

THINGS TO KEEP IN MIND

- Avoid being vague
- Realize that not only are you communicating with future-you, but potientially others
- When you can focus on why you did something not what you did

ACTIVITY

FIRST COMMIT

- Open the folder for your repo from the GitHub app
- Create a new file named "README.md" inside the folder
- Commit your changes from the GitHub app
  Make sure to provide a good description for your commit

SYNCING

In the command-line world known as pushing

Takes the changes you made on your computer and sends them off to the GitHub website

ACTIVITY

SYNC YOUR CHANGES

- Open the GitHub app
- Click on sync to send your changes to the GitHub website
- View the changes on GitHub: https://github.com/**[username]**/**[repo name]**

A QUICK NOTE ON READMES

README files are generally provided with software or other files that provides information for the person opening it.

They can contain:

- How to use a piece of software
- Who made the software
- Any bugs in a chunk of software

According to the jargon file the name comes from Lewis Carroll's *Alice's Adventures In Wonderland* when she encounters "Eat Me" and "Drink Me"

READMES ON GITHUB

- Always will display on the main page on the repo
- Are important to those viewing the project so they can see what it is about and how to use it
- Supports the ability to make the content appear with styling using MarkDown
- Example: reveal.js README

MARKDOWN

An extremely lightweight markup language.

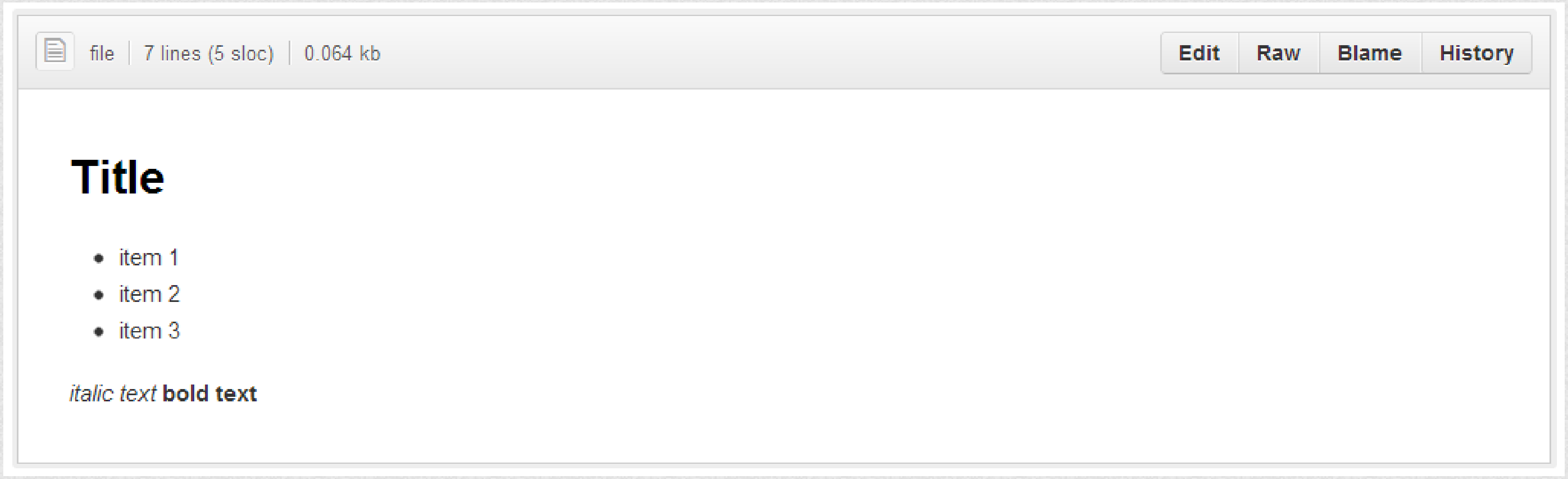
Reference

MARKDOWN EXAMPLE

```
# Title

* item 1
* item 2
* item 3

*italic text* __bold text__
```



Title

- item 1
- item 2
- item 3

italic text **bold text**

YOU DON'T HAVE TO USE MARKDOWN

Plain text will still display perfectly fine

ACTIVITY

ADD SOME CONTENT TO YOUR README

- Open "README.md" in a text editor
- Add any content that you desire with or without MarkDown
- Commit your changes from the GitHub app
  Make sure to provide a good description for your commit
- Sync your changes
- View your new README on the GitHub website
  https://github.com/**[username]**/**[repo name]**

COMMITS DON'T NEED ALL FILES

We can tell git that we only want to commit changes to some files rather than all.

WHEN DO WE WANT TO ONLY COMMIT SOME FILES?

- If we don't want to save our changes in other files.
- If the other files are in some broken state.
- If the other files are still "works in progress".

HOW DO WE COMMIT ONLY FILES WE WANT IN THE GITHUB APP?

The GitHub app uses checkboxes to let us select which files we want to include as part of our commit.

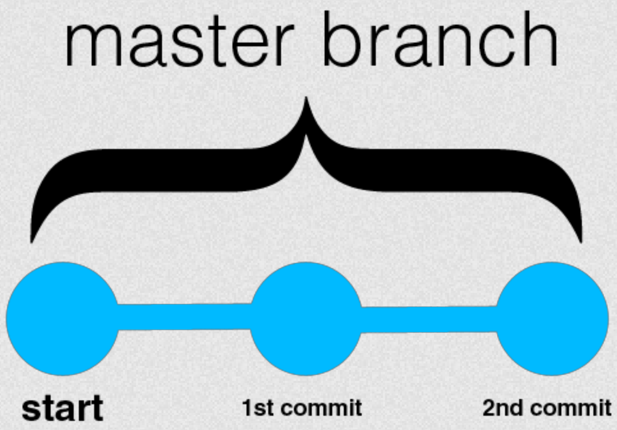
ACTIVITY

COMMIT ONLY ONE FILE

- Create "NewFile.md"
- Commit your changes from the GitHub app and be sure to only select this file.
- When you click on the commit only this file will show up in the list of changes.

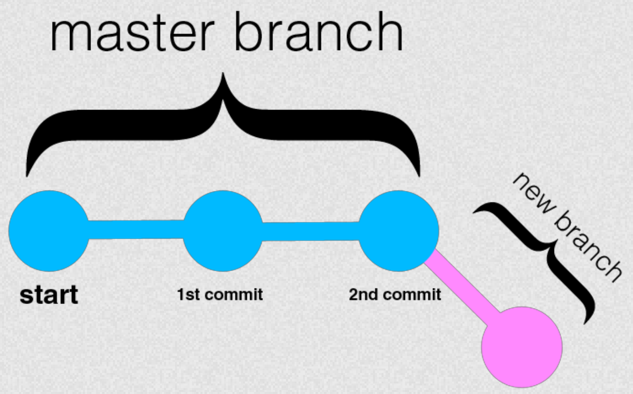
BRANCH

- Where your **changes** live.
- By deafult you start out in the **master** branch
- Difference branches are like parallel universes: they may be very different from one another or very similar and they exist at the same time.



CREATING A NEW BRANCH

When we create a new branch we keep all the old files as is, but are able to make changes that don't affect the **master** branch.



WHY WOULD WE DO THIS?

- Working on code we are unsure is correct.
  (e.g. bug fixes)
- Working on a completely new version of something.
- Working on a specific feature we want to add.

HOW DO WE CREATE A NEW BRANCH IN THE GITHUB APP?

HOW DO WE VIEW A DIFFERENT BRANCH ON GITHUB.COM?

ACTIVITY

CREATING A NEW BRANCH

- Create a new branch named "song"
- Add any song lyrics to your "README.md" file
- Commit your changes
- Sync your changes
- View your README on the GitHub website https://github.com/**[username]**/**[repo name]**
  note: it will still look the same
- Change branches on the github website and now look at your README.md file.

MERGING

When we take the changes of one branch and pull them into a second branch.

WHEN DO WE WANT TO MERGE?

- When we have tested our new bug fixes and want to use it.
- When we are ready to go live with new features.
- When we are ready to go live with a new version.
- When we want to combine two new features together.

HOW DO WE MERGE IN THE GITHUB APP

ACTIVITY

CREATING A NEW BRANCH

- Merge the **song** branch into the **master** branch
- Commit your changes
- Sync your changes
- View your README on the GitHub website
  https://github.com/**[username]**/**[repo name]**
  note: it should now have your song lyrics

ROLLING BACK COMMITS

Sometimes you want to undo a commit, but still **keep your changes**.

HOW DO WE ROLL BACK IN THE GITHUB APP?

REVERTING A COMMIT

Sometimes you want to completely undo a commit.

HOW DO WE REVERT IN THE GITHUB APP?

ACTIVITY

REVERTING SOME CHANGES

- Open "README.md" and add "Changes I don't want" to it.
- Open the GitHub app and commit your changes.
- Now click on the commit from the list and click the revert button.
- Notice you will now have a new commit that says "Revert..."

FURTHER LEARNING

1. Code School - Try Git (command-line based tutorial)
2. Git's official documentation
3. Git Immersion a git walkthrough from the command-line

THE END

THANK YOU!