JMS 302: HACKING THE MEDIA

CLASS 2: 8/21/2014

TODAY'S TASKS

- Your First Programs

- Web/Program Planning

- HTML Structure & Text

Facebook group
First, introductions (ball)

REVIEW

- Norms

- Hacking

- Getting around the command line

1. Name a norm for this class.
2. Define hacking (two definitions).
3. Give an example relevant to this class.
4. T/F: It's important to spend as much time as possible planning so that you only have to write your code once.
5. What does DRY mean?
6. Who do you get documentation on a command (while in the CLI, or terminal)?
7. How do you list files in a directory?
8. How do you determine what directory you're in?
9. Describe what "`cd ../`" would do.

GIT & GITHUB

- Create an account on GitHub(.com)

- Launch GitHub app and sign-in.

- Rename "code" folder (directory) on lab machine to "JMS 302" and lowercase exercises.

- Now follow along.

See "Set Up Git"

We'll create a "day_1" directory and a file or two and commit them.

Get used to creating files without spaces in them, especially important for the web (you can get away with spaces, but they can cause problems).

- Simple Ruby

- Simple HTML

- Simple CSS

- Simple JavaScript

1.   Launch Brackets
2. File > Open Folder: the JMS 302 folder you just renamed.
3. Go to (or create) "day_1" folder in "exercises".
4. Create new file, "hello_world.rb" and type 'puts "Hello, world!"'.
5. Run from Brackets and command line. From CLI, you'll need to type "ruby" before the filename.
6. Create "hello_world.html" and add "Hello, world!" in a paragraph.
7. Next, we'll add CSS to make the whole paragraph (and every paragraph) bold. The temptation is to add it inline, but we're going to create an external file.
8. Now, we'll add a simple JavaScript alert inline. This should really go in an external file as a matter of habit.

See sample code in class_2 folder in GitHub.