Good Afternoon CEG 2350!

Aiden Cox - CEG 2350 Lab Lead
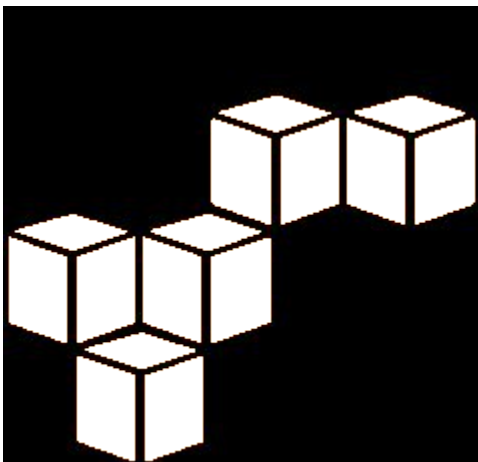Austin Kellough - CEG 2350 Lab Assistant
Quote of the week:
"An investment in knowledge always pays the best interest"

1:25pm - 2:20pm

September 8th - September 12th

Check in! How was Lab02?

Questions over anything up until now? Permissions, AWS, where things are located?


```
~ ❯ lls                                           11:43:57 AM
drwxr-xr-x@  -  aidencox 22 Apr 22:44 AndroidStudioProjects
drwxr-xr-x@  -  aidencox  7 May 21:19 Applications
drwx------@  -  aidencox 21 Jun 12:33 Desktop
drwx------@  -  aidencox 12 May 20:26 Documents
drwx------   -  aidencox 19 Jun 20:29 Downloads
drwxr-xr-x   -  aidencox 12 May 20:34 git
drwxr-xr-x@  -  aidencox 13 Mar 14:01 gradingScripts
drwxr-xr-x@  -  aidencox 29 Apr 12:59 IdeaProjects
drwxr-xr-x   -  aidencox  4 Apr 15:03 java
drwxr-xr-x@  -  aidencox 17 Feb 15:39 Keys
drwx------@  -  aidencox 23 Jun 11:15 Library
drwx------   -  aidencox 11 Feb 12:48 Movies
drwx------   -  aidencox 16 Feb 20:30 Music
.rw-r--r--@ 87 aidencox 24 Mar 12:07 package-lock.json
drwx------   -  aidencox  9 Apr 15:12 Pictures
drwxr-xr-x   -  aidencox 10 Feb 20:04 Public
drwxr-xr-x@  -  aidencox  8 Apr 22:10 pythonEnv312

~ ❯                                               11:43:58 AM
```
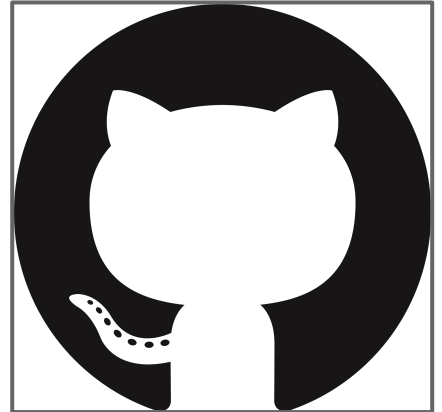
Beginning Lab03

Lab Instructions: https://pattonsgirl.github.io/CEG2350/Labs/Lab03/Instructions.html

Lab Template:
https://raw.githubusercontent.com/pattonsgirl/CEG2350/refs/heads/main/Labs/Lab03/LabTemplate.md

Discuss Git commands yet?

Git Guide - Part 1

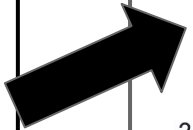| git command	Description
git clone repo_URI	
git status	
git add filename	
git commit	
git push	
git pull	

In your template, there are git commands that you will need to describe.

These commands will be used frequently, so make sure you understand what they are doing.

Clone Repo - Part 2

Professor Duncan's GitHub Setup Markdown: https://github.com/pattonsgirl/CEG2350/blob/main/GitHubSetup.md

Screenshot is here to show what the process looks like, we can do it together if wanted!

Setup SSH Authentication

1. Generate a new SSH key with the command:

- `ssh-keygen -t ed25519 -C "your_email@example.com"`
- Hit enter to use the default storage location and name
- Hit enter to **skip** entering a passphrase
- **NOTE** You need to create a key pair on each system you want to use.
    - AWS Ubuntu system = 1 key pair
    - Your system = 1 key pair
    - A system at wright state = 1 key pair

2. Go to the folder where the key pair was saved. Look for the file that ends with `.pub`
3. Open the file, and copy its contents
4. Add your public key to GitHub

- Go to "Settings" - "SSH and GPG keys", then select "Add new key"
- Give your key a useful name that ids which computer it belongs to
- Paste the contents of the `.pub` file in the box

5. To `clone` the repository:

- Click the green `Code` button
- Select `SSH` for SSH key authentication
- In a terminal use `git clone` followed by the corresponding URL to clone your repository

IO Redirection - Part 3

1. `printenv HOME > thishouse`

2. `cat doesnotexist 2>> hush.txt`

3. `cat nums.txt | sort -n >> all_nums.txt`
   - To run: create a file named `nums.txt` containing random numbers, each on a new line

4. `cat << "DONE" > here.txt`

5. `ls -lt ~ | head`

6. `history | grep ".md"`

```
~ ❯ history | grep "clear"                    09:57:09 AM
  1016  clear
  1019  clear
  1022  clear

~ ❯ █                                          09:57:18 AM
```

Roll the dice - Part 4

Assigning the text to variable $name

Tells the system to use the Bash shell to interpret the file

Assigning the arithmetic value to the variable $random. Might be useful later ;)

Printing what's stored in "name" to terminal!

```bash
1  #!/bin/bash
2
3  name="Aiden (the best 2350 TA)"
4
5  echo $name
6
7  random=$(($RANDOM % 20 + 1))
8
9  echo "you rolled a $random"
10
```

Printing out "you rolled a " and the value stored within $random

Retrospective - Part 5 and EC

🔗 Part 5 - Retrospective

A retrospective is a moment of refection after a task.

1. Where and when did it go wrong while working on your script tasks?
2. Was anything familiar working with a new language compared to one you are used to?
3. Assuming you `push` ed your `commit` s to GitHub, view your `commit` history. Did you write good `commit` messages that refer to what tasks were completed at each commit? What would you improve?

Extra Credit - Room for Improvement

Make a notable improvement to the `roll` script. This could be another feature, an additional boundary check on allowed conditions, experimenting with colored text for standard and error output... If you have an idea, you are encouraged to run it by the TAs first to make sure it is worthy.

Make sure you `commit` and `push` your script changes for grading and add a note of what you did in the answer template.

Review Of How to Submit

Submission

- In the Pilot Dropbox, paste the URL to the `Lab03` folder in your GitHub repo
  - URL should look like: https://github.com/WSU-kduncan/ceg2350-YOURGITHUBUSERNAME/tree/main/Lab03
- Your `Lab03` folder should contain (at minimum):
  - `README.md`
  - `roll`

Your Lab03 folder will now have 2 different files in it, one of the lab template with questions filled out, and the Roll script.

Reminder that you do not need to keep each part of the roll script on the final submission, but you **MUST commit** after each step is finished.

Any other questions? More scripting demos? Any feedback for me / professors?

Have a Good Weekend

Don't hesitate to reach out and ask questions!
Quote of the week:
"An investment in knowledge always pays the best interest "

1:25pm - 2:20pm

September 8th - September 12th