Exercise 6 - Collaboration with GitHub

Work with a partner to complete the tasks below. After completing the collaborative activity, complete the exercise questions below. Submit your results for the exercise questions via a pull request on GitHub by the beginning of tutorial next Friday.

Activity

1. Pick a partner and decide who is the repo **owner** and who is the repo **collaborator**.
2. **Owner** should create a new repo on Github named `fav_movies`
3. **Owner** should clone this repository (be sure to get the SSH version) to their local machine. Use the command `git clone <repo url copied from GitHub>` to do this.
4. **Owner** should use `nano` to create a text file that includes their three favorite movies as the first three lines of the text file. Save this with an informative file name then add and commit a version of your Git repo. In order to update the GitHub repo the owner must now push their latest commit to GitHub. Use the command `git push` to accomplish this. Git may ask you for your GitHub username and password at this point.
5. **Collaborator** should now fork **Owner's** Github repository on the Github webpage. Then clone that repository to their local machine. Now, add your favorite three movies to the fourth through sixth lines of the text file created by your collaborator. Add and commit your changes to your local Git repo and push those changes to GitHub.
6. To synchronize the repositories, **Collaborator** must submit a pull request on the GitHub webpage. **Owner** must check and accept this pull request. The **Owner** then must use `git pull` to sync their local git repo with the upated Github repo.
7. Now change roles and repeat the process above, but rank your favorite foods instead of movies.

Assignment

Armed with your new GitHub skills, fork the Exercise 6 repo from your TA's Github (link available on Canvas), clone that repo on your local machine, and develop a single shell script that accomplishes the three tasks below. These tasks will require the file "wages.csv", which you will have in your local computer once you clone the repo you forked from the TA. Don't worry about an orthogonal script or your script taking arguments this time. The script you will write for this exercise definitely falls in the category of a specific data analysis.

1. Write a file containing the unique gender-yearsExperience combinations contained in the file "wages.csv". The file you create should contain gender in the first column and yearsExperience in a second column with a space separating the two columns. The rows should be sorted first by gender and then by yearsExperience, but remember to keep the pairings in a given row intact. Don't worry about column names in the output file.
2. Return the following information to `stdout` when the shell script is executed: 1) the gender, yearsExperience, and wage for the highest earner, 2) the gender, yearsExperience, and wage for the lowest earner,

and 3) the number of females in the top ten earners in this data set. Be sure to indicate, which output is which when returning them to `stdout`.

3. Return one more piece of information to `stdout`: the effect of graduating college (12 vs. 16 years of school) on the minimum wage for earners in this dataset. Two hints: 1) you can assign the output of a pipeline to a variable with this code `variable_name=$(code)`, where variable_name can be any name of your choosing and "code" represents a Linux pipeline, and 2) you can assign numeric values to variables and then us the command `bc` to do simple arithmetic. If you've defined two shell variables (`val1` and `val2`) that are decimal values you can subtract them with the following code: `echo "$val1 - $val2" | bc` . If using `bc` doesn't work on your system, test your code up to this last step and then add the line necessary to calculate the difference on a system that can use the `bc` method given above to your script.

Turning in your assignment via GitHub

Once you have committed all changes to your local Git repo and pushed all of those commits to the forked repo on GitHub, you can "turn in" your assignment using a `pull request`. This can be done from the GitHub repo website. When viewing the forked repo, select "Pull requests" in the upper middle of the screen, then click the green "New pull request" button in the upper right. You'll then see a screen with a history of commits for you, select the green "Create pull request button". In the text box next to your user icon near the top of the page, remove whatever text is there and add "last name -submission", but obviously substitute your last names. Then click the green "Create pull request" button.