Problem Set 1

Overview

Learning how to use git and GitHub can be intimidating. Problem sets this quarter will give you the opportunity to practice using git and GitHub both individually and in groups. In this problem set we are asking you to create a folder and file using your command line interface (Git Bash or terminal). These files you create on your local computer will later be pushed into a remote repository you create. These exercises will give you practice with the git/GitHub workflow (e.g. changing directories, creating files, staging those files, committing, and pushing to GitHub). Lastly, you will practice creating issues on our **rclass2** repository. We will be using the **issues** page in the **rclass** repository to ask questions.

Part I: Command line & R practice (7 pts)

1. Using your command line interface (CLI) (e.g. Git Bash, terminal), create a new folder called **scripts**. Change directories to where you want to save this folder. Be intentional about where you create this folder (hint: change directories first). Write the commands you used here (to change directories and create the folder):

```
cd Desktop
mkdir scripts
```

2. Change directory into the **scripts** folder and write the command you used here:

```
cd scripts
```

3. Inside the **scripts** folder, create a file called `problemset1.R` via the command line. Write the command you used here:

```
touch problemset1.R
```

4. Open `problemset1.R` in RStudio to edit the file and perform the following tasks:

- Load the `tidyverse` library
- Preview the first 5 rows of the `mpg` dataframe
- Filter the `mpg` dataframe to include only Ford Mustang's that were built after the year 2000

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts ------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
mpg %>% head(5)
```

```
## # A tibble: 5 x 11
##   manufacturer model displ  year   cyl trans      drv     cty   hwy fl    class
##   <chr>        <chr> <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
## 1 audi         a4      1.8  1999     4 auto(l5)   f        18    29 p     compa~
```

```
## 2 audi         a4      1.8  1999     4 manual(m5) f        21    29 p     compa~
## 3 audi         a4      2    2008     4 manual(m6) f        20    31 p     compa~
## 4 audi         a4      2    2008     4 auto(av)   f        21    30 p     compa~
## 5 audi         a4      2.8  1999     6 auto(l5)   f        16    26 p     compa~
```

```
mpg %>% filter(manufacturer == 'ford', model == 'mustang', year > 2000)
```

```
## # A tibble: 5 x 11
##   manufacturer model  displ  year   cyl trans    drv     cty   hwy fl    class
##   <chr>        <chr>  <dbl> <int> <int> <chr>    <chr> <int> <int> <chr> <chr>
## 1 ford         musta~   4    2008     6 manual(~ r        17    26 r     subcom~
## 2 ford         musta~   4    2008     6 auto(l5) r        16    24 r     subcom~
## 3 ford         musta~   4.6  2008     8 manual(~ r        15    23 r     subcom~
## 4 ford         musta~   4.6  2008     8 auto(l5) r        15    22 r     subcom~
## 5 ford         musta~   5.4  2008     8 manual(~ r        14    20 p     subcom~
```

Part II: GitHub (10 pts)

1. Log in to your GitHub account online and create a new private repository here: https://github.com/organizations/Rucla-ed/repositories/new

   Name it **lastname_ps1** and do not forget to initialize it with a `README.md` file. Paste the link to your repository here:

2. Clone the **lastname_ps1** repository to your local machine (again be intentional about where you save this folder):

   Write the git command you used here:

```
cd ../
git clone https://github.com/mpatricia01/martin_ps1.git
```

3. Change directory into **lastname_ps1**. Write the command you used here:

```
cd martin_ps1
```

4. What is the command to list all the directory contents in **lastname_ps1**, including hidden files and directories (ie. entries starting with `.`)?

   Write the command you used here and paste your output:

```
ls -a
. .. .git README.md
```

5. Since **lastname_ps1** is a git repository, you can run git commands in this directory. What is the command to check the current state of the repository?

   Write the git command you used here and paste your output:

```
git status
On branch master
Your branch is up to date with 'origin/master'.
```

6. Go back to the folder **scripts** you created in Part I and move this folder inside your **lastname_ps1** folder on your local machine. Now check the status of your repository (what you did in the question above).

   Write the git command you used here and paste your output:

```
git status
On branch master
Your branch is up to date with 'origin/master'.
Untracked files:
(use "git add <file>..." to include in what will be committed)
scripts/problemset1.R
nothing added to commit but untracked files present (use "git add" to track)
```

7. Add the new folder and file you've created (`scripts/problemset1.R`) to the staging area. Write the git command you used here:

```
git add scripts
```

8. Commit your changes with a message of your choice and write the git command you used here:

```
git commit -m "adding script folder + .R file to problem set 1 repo"
```

9. Push your changes to the remote repository and write the git command you used here:

```
git push
```

10. Finally, add this file you are working on (`problemset1.Rmd`) to the **scripts** folder and push to the remote repository as well.

Part III: GitHub issues (3 pts)

1. Navigate to the issues tab for the **rclass2** repository here: https://github.com/Rucla-ed/rclass2/issues

   Create a new issue titled "Problem Set 1 - YOUR NAME" and post a question of your choice (e.g favorite food, self-care practices, favorite shows). Add the "question" label to your issue and assign it to 3 students in the class who you do not know.

   Once your issue received 3 responses, close the issue. If other students assigned you to their issue, make sure to post your response as well. You should get an email notification if you were assigned to an issue.