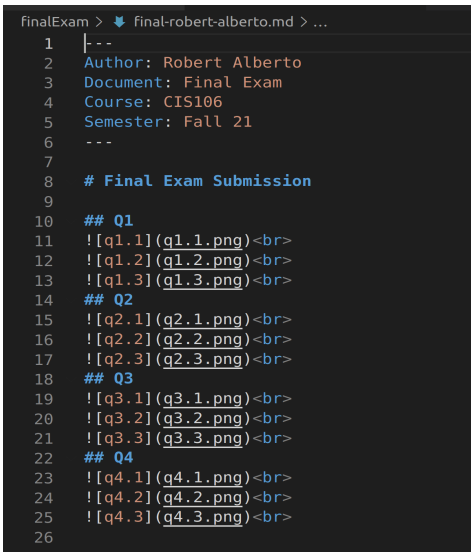
# Spring 23 Final

---

## Complete these steps before starting the exam.

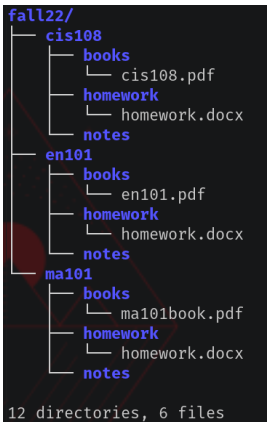



- In your git repo (cis106 directory), create a directory called: **`finalExam`.** This is going to be the directory where you will place your screenshots and the markdown file that you will use to answer all the questions.
- Inside the **`finalExam`** directory, create a file named: **`final-firstname-lastname.md`**
- Use the following naming convention for your images: **`q#.1.png`** for example, **`q1.1.png`** will be for question 1 part one if more than one screenshot is required.
- In the file **`final-firstname-lastname.md`** write the text in the image. *Obviously, replace my name with yours.*
- Each question is worth **25 points**. For a total of 75.
- Before you start working on any question, make sure to be in your home directory (cd)
- All questions are independent of each other. You can work in any order you like.
- Make sure that you have enough space in your virtual machine so clear your trash and delete everything form your Downloads folder.

# Question 1

**Scenario**

Kathy is a computer science major at PCCC. Every semester she creates a directory structure for the courses she is taking. Here is how last semester's directory tree looked like: →




This Spring semester Kathy took the following courses:

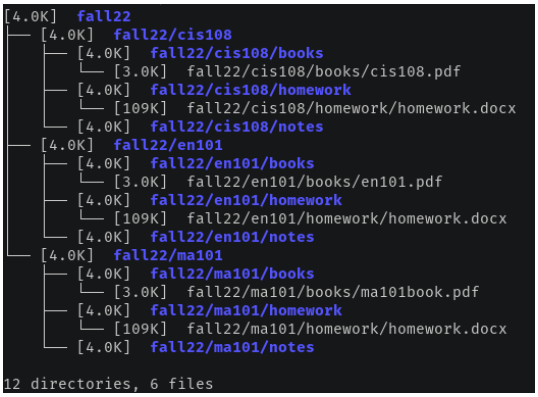
- CIS 160 Fundamentals of Computer Science
- EN 102 Composition II
- MA 121 Calculus II

Kathy's best friend, Rebecca gave her the book for each course in PDF format, the homework list for each course and her study notes. All of Rebecca's files can be found here:
https://github.com/linuxworkshop67/rebeccasfiles

**Instructions:**

1. Clone the repository that contains all of Rebecca's files in your home directory. You must do this in the terminal. You are **not allowed** to open the URL and download the zip file.
2. In the Documents directory, create a directory for Kathy's Spring semester and all the necessary subdirectories.
3. Rename the Rebecca's files using the following naming convention: →
4. Move all Rebecca's files to their corresponding directory in the Spring directory you created earlier.
5. Display a tree of the Spring directory that looks exactly like this one: →

| | |
|---|---|
| 1.docx | cis160.docx |
| 2.docx | en102.docx |
| 3.docx | ma121.docx |
| notes1.md | cis160.md |
| notes2.md | en102.md |
| notes3.md | ma121.md |
| 1.pdf | cis160.pdf |
| 2.pdf | en102.pdf |
| 3.pdf | ma121.pdf |




***Take a screenshot of all the commands you used to complete this question.***

# Question 2

Complete the following instructions:

1. Run the following command: `curl https://cis106.com/assets/final_q2.sh | bash` This command will create a directory in your home directory called `final_q2` You will use the files in this directory to complete the rest of the instructions.
2. Change your current working directory to `/usr/share`. You will complete the rest of the instructions in this directory. Until you complete this question, you are not allowed to use cd.
3. Display a long list **without owner and group**, **with human-readable file size**, and **sorted by file extensions** of all the files that contain the number 1 and 2 in their name inside the `final_q2 directory`.
4. Create a folder in your home directory called `all_files_organized`. Inside the `all_files_organized` directory, create 1 directory for each file type. *The programs files should be in a single directory because they are all scripts written in different languages*.
5. Move all the files inside `final_q2` directory to their respective new directories. You are **NOT** allowed to move 1 single file at the time. *You must use wildcards!*
6. List all the files inside the `all_files_organized` directory recursively and sorted by filesize and save the output to a file called `report.txt`
7. Display the content of the `report.txt` file
8. How many characters and lines does the file `report.txt` have?
9. Long list all the files inside the `all_files_organized` directory **without the files permissions**, and **without group** and **owner**
10. Append the output of the previous step to the file: `report.txt`

***Take a screenshot of all the commands you used to complete this question.***

# Question 3

**Tips:**

- Inspecting a file means to look inside the file using a command that can search for words in every line of a given file.
- Do not get intimidated by the HTML. What you are looking for is a path that shows you which folders you need to make.

**Scenario Description:**

Fred is a first year web development student, he downloaded an example website but when he opened it in Firefox he noticed that the page looked broken. All the assets (images, scripts, etc.) are not in their right place. Help Fred fix this site. You do not need to know HTML to do this. Also, you don't need to modify the HTML. Inspect the HTML file so that you can place each file in their respective folder.

**Instructions:**

1. Download the broken site by cloning this repository: `https://github.com/linuxworkshop67/broken_website`
2. Inspect the **index.html** to find the directory structure you need to create to fix the site. Here is what you need to know about HTML and CSS.
    a. The location of files is referenced with the attribute `src="file/location/here"` or with the `href="file/location/here"` or with `url("../location/of/images")`. You can ignore the ../ in the url().
2. Create the necessary directories in a single command.
3. Move all the files to their respective directories using a wildcard when needed
4. Open the website in Firefox. You can use the command **`firefox index.html &`**

***Take a screenshot of the terminal showing all the commands you used to fix the website. Take a screenshot of Firefox showing the website fixed.***

# Good Luck!