Attempt all Tasks

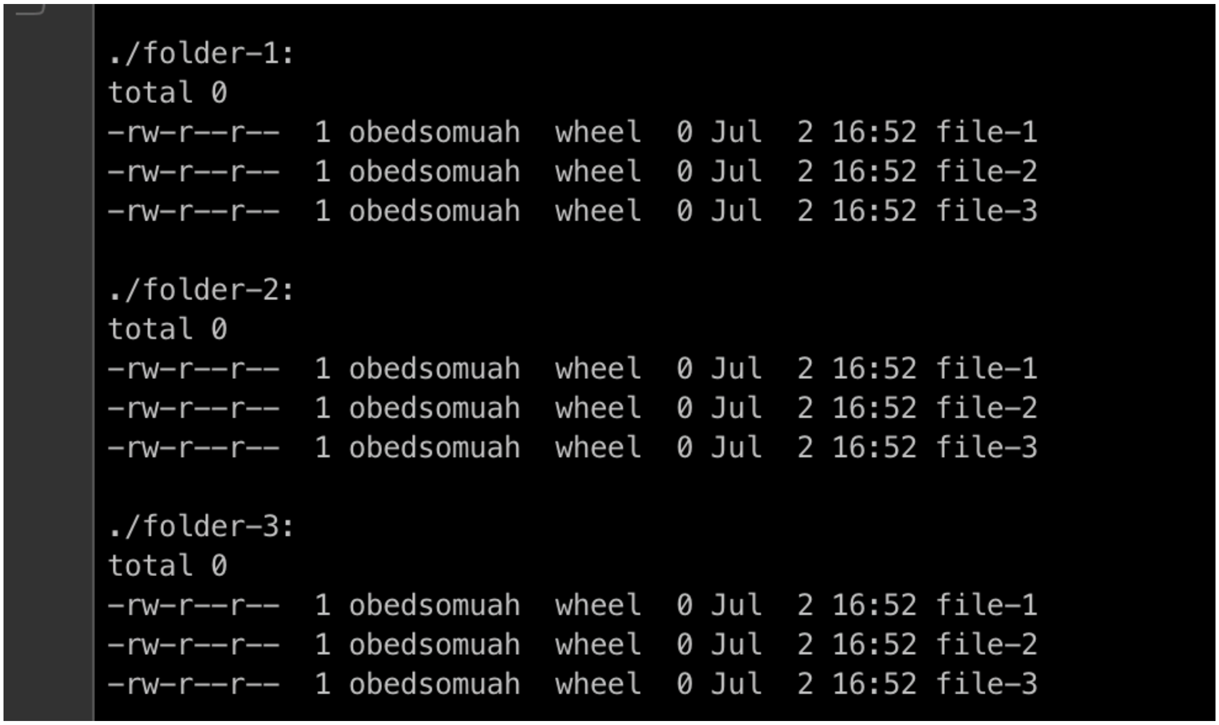
Task-1

Write a `Bash script` that:

- Prints a welcome message to the user.
- Asks the user for their name and stores it in a variable.
- Outputs the user's name and the current date
- Take a screenshot after running your script.

Task 2

- Create a script that accepts three arguments.
- Use these arguments as names of a directory
- The script then creates the directories and then creates three files in each directory.
- The script should then copy the created directories into /tmp/
- Take a screenshot after running `ls -lR .` in your `/tmp/`. Sample output below



```
./folder-1:
total 0
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-1
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-2
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-3

./folder-2:
total 0
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-1
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-2
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-3

./folder-3:
total 0
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-1
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-2
-rw-r--r--  1 obedsomuah  wheel  0 Jul  2 16:52 file-3
```

Task 3

- Write a script that asks the user to enter their name
- Check if the name the user entered is 'admin'
- Check if there is no name entered, ask the user for input again (*hint: use string tests*)
- Print out a message indicating the user is the admin
- Modify the script above to check for sudo. Print an error is script is run without `sudo` (meaning this script should only run successfully after using `sudo`) (*hint: use the environment variable $EUID*)
- Take a screenshot after executing your scripts

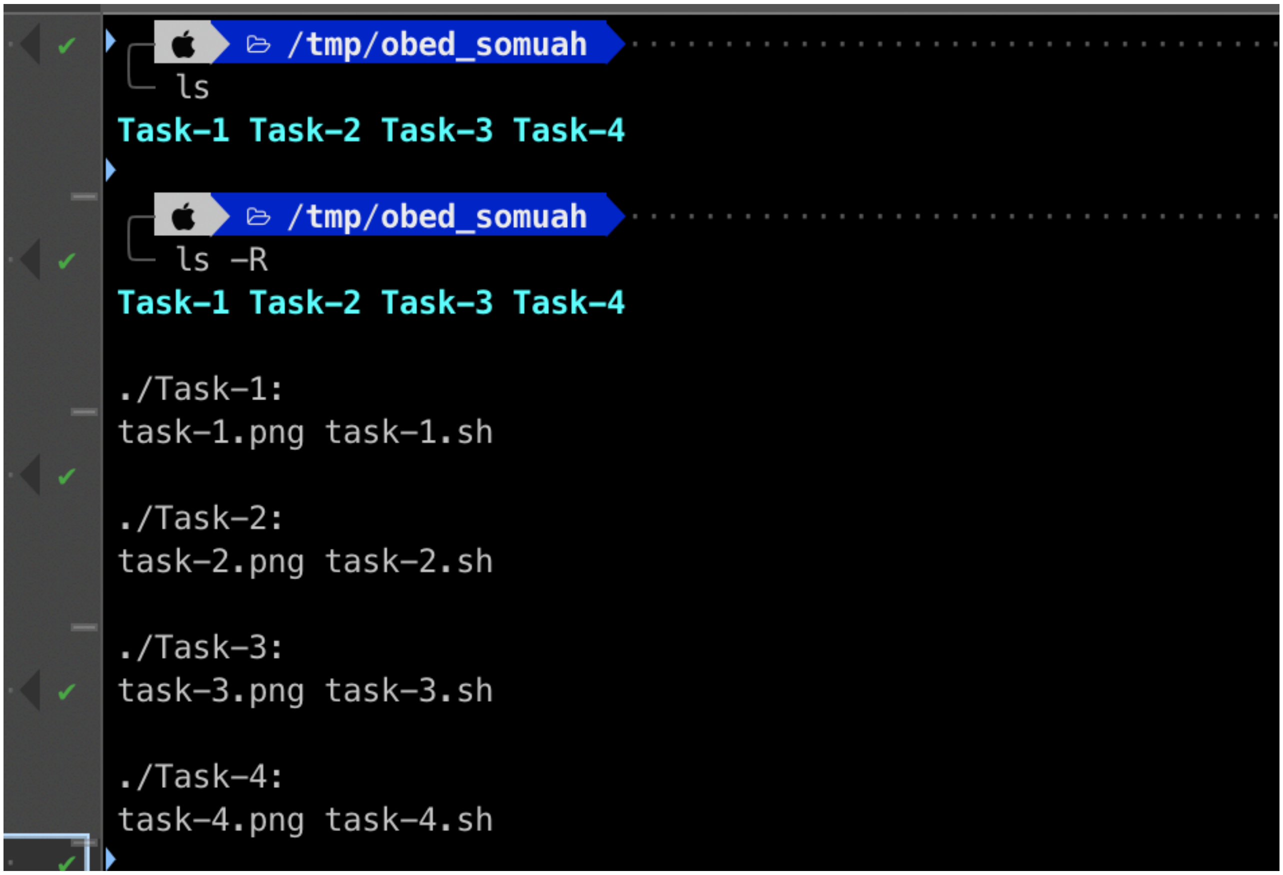
Task 4

- Write a script that checks if a file called 'users.txt' exists.
- If the file does not exist, print a message indicating the file is being created
- Ask the user for their name and email
- Save these details into the file 'users.txt' (*hint: use output redirection to a file*)
- Open the file above using cat showing the new name and email saved after the script was executed

NOTE

- Make sure all your scripts have execute permissions
- Name your scripts(.sh) and screenshots(.png or .jpg) according to their task names. Eg Task-1.sh, Task-1.png
- Put each corresponding task in a corresponding folder with the task name.
- Zip all the tasks as one folder and rename with your full name(No spaces in between names, use underscores).

Sample output



```
/tmp/obed_somuah
ls
Task-1 Task-2 Task-3 Task-4

/tmp/obed_somuah
ls -R
Task-1 Task-2 Task-3 Task-4

./Task-1:
task-1.png task-1.sh

./Task-2:
task-2.png task-2.sh

./Task-3:
task-3.png task-3.sh

./Task-4:
task-4.png task-4.sh
```

Reference scripts included in the '*Resources.zip*'