https://github.com/NobodySleep/inet_4031_adduser_script

Submission 2: The code here is not that complex and can be somewhat figured out by reading the code. However, it is bad practice not to include comments in the code for other users/developers. There are design choices and semantics that may not be easily clear. Comments as well indicate the intended outcome, so if it does not operate correctly, it can be easier to find the issue. If I was handed this code in a professional setting, I would tell them to go back and add comments.

Submission 3: Because of the check for match and len(fields), if the input was not sufficient. It would halt the script and the user would have to run it again. If they want to bypass putting a real input, they would use a '#' to make a comment instead, which would allow the script to continue.

Submission 4:

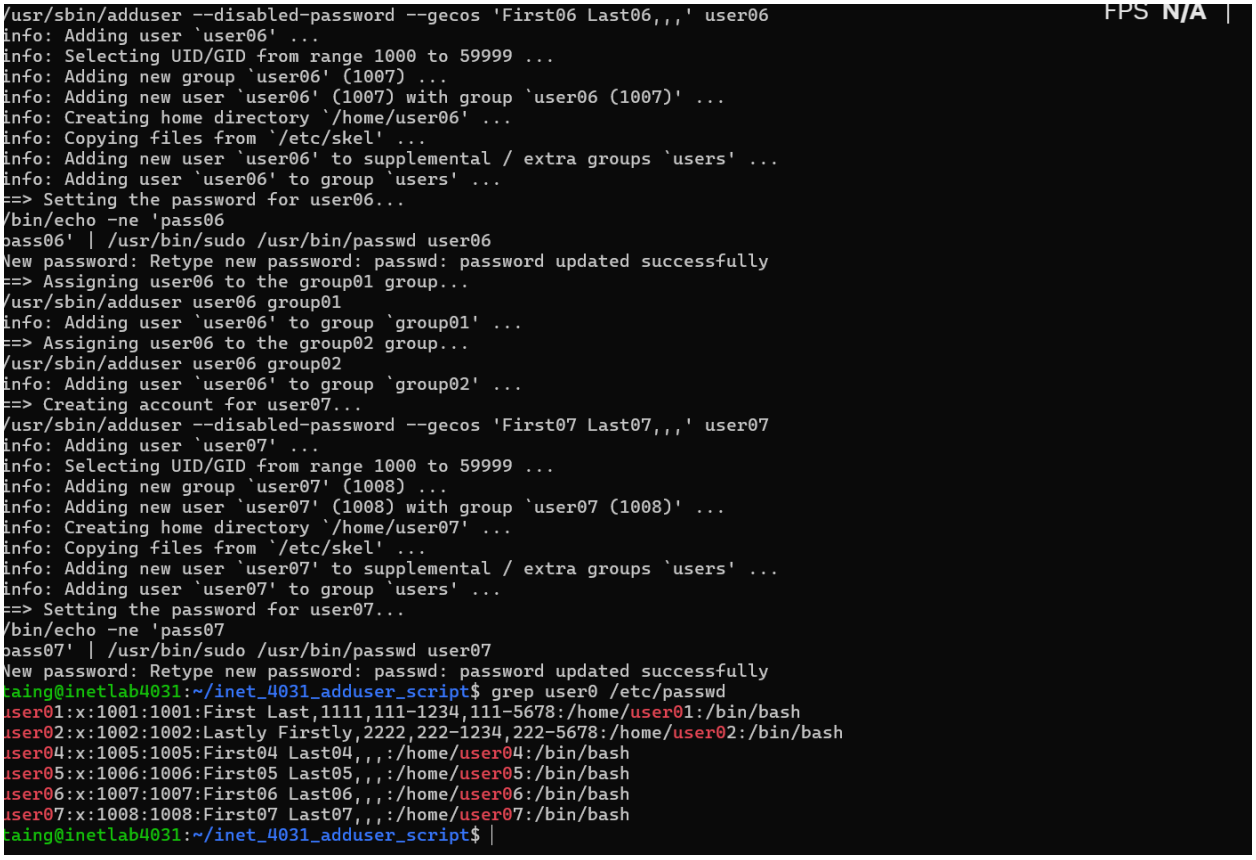
```
taing@inetlab4031:~/inet_4031_adduser_script$ cat create-users.input
user11:pass11:Last11:First11:group11
user22:pass22:Last22:First22:group22
user33:pass33:Last33:First33:group22, group11
#user12:pass12:Last12:First12:group12
user44:pass44:Last44:First44:-
user21:Last21:First21:group21

taing@inetlab4031:~/inet_4031_adduser_script$ ./create-users.py < create-users.input
==> Creating account for user11...
==> Setting the password for user11...
==> Assigning user11 to the group11 group...
==> Creating account for user22...
==> Setting the password for user22...
==> Assigning user22 to the group22 group...
==> Creating account for user33...
==> Setting the password for user33...
==> Assigning user33 to the group22 group...
==> Assigning user33 to the  group11 group...
==> Creating account for user44...
==> Setting the password for user44…
```

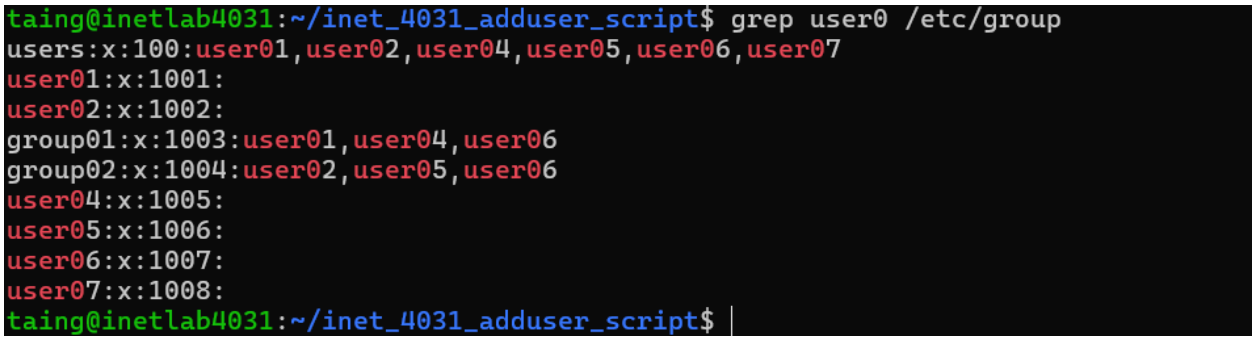
Submission 5:

1. This is required in Unix based OS to tell the interpreter to run the script through python. This also allows it to run ./create-user.py without python3 in front of it.

2. When I attempted to do it, it could not locate the file. An additional step was needed prior to using sudo. With the chmod command, chmod +x create-users.py, the program can then run.
3. < is the file that is used inside of the script. It is basically telling the file on the right to go into the file of the left.
4. The line in the python file has for line in sys.stdin, which reads each line and runs the whole script with it. After the first line is done, the next one is passed.

Submission 6:





Submission 7:

grep finds all of the instances of the given string and outputs all of the lines that contain that string.

Submission 8-10: I tried adding the dry-run option, but it just did not work and I could not figure it out. I still have create-users2.py to look over.