

# Bash Programming Assignment

---

**Due** Wednesday by 11:59pm      **Points** 80      **Submitting** a file upload  
**Available** until Dec 6 at 11:59pm

---

Make sure you read all the directions and understand how to use the different commands.

(Plan before you type)

*(I will attach a demo folder with files soon)*

Bash Programming Assignment

Pre-reqs: **You need to install lamp – sudo apt-get install lamp-server^ (^ is important)**

You will make a program that will take in several different options:

1. `-v` : Verbose – you will echo a message every step
2. `-f` : The very next argument will be the directory where the user information is stored
  1. Default is `./pers`
3. `-l (L)`: All output will be directed to a file called “Log.txt”
4. `-e` : Extensions – The next argument will specify what extensions you are looking for
  1. Default is `.peop`
5. `-noLamp` : You will not create a `public_html` for the user nor will you link anything to the `/var/www/html` folder

This program will read in all the above arguments, can be in any order (except for `-f` and `-e`). All arguments must be checked and verified. Once all arguments are validated, you will then find all the files with the appropriate extension underneath the appropriate directory.

You will then read the name from each file and invoke `adduser` (use `--quiet` and `--force-badname`) and pipe the particular file into it. (The files will be formatted to what `adduser` is expecting). After the user is created, but before you move to the next name (if `noLamp` is false), you will create a `public_html` folder underneath the users home folder, change the owner to that particular user, and make a symbolic link from that folder to `/var/www/html`. You will also create an HTML file containing “Hello World” inside the `public_html` folder (Once again setting the ownership to the correct user.)



You should be able to switch user to any of the new users, their passwords will be their logins.

**HINT and WARNING: echo the command you think you are going to run first, as if you make a mistake bad things can happen!**

**echo "adduser --quiet --force-badname \$N < \$i " # This may NOT be what you think it is.**

**! If any errors occur, you must break the script and give a descriptive message as to where the error occurred!**

Files:

Here is the pers [directory](#)  with three user information. Once you get the adduser working you may need to rerun it. I have created a [script](#)  that will read a specified directory and delete those users. (Invoke: ./removeUser ./pers)

### **Pseudo Code**

Validate Arguments

For each file.<ext> under <dir>

    Read Name

    addUser \$Name

    if(! noLamp)

{

    Create public\_html under /home/\$Name

        Create index.html containing “Hello World” underneath public\_html

        Change owner for directory and file

        Create symbolic link

}

End for

File Upload	<a href="#">Google Doc</a>
Upload a file, or choose a file you've already uploaded.	
File:	<input type="button" value="Browse..."/> No file selected.
<a href="#">+ Add Another File</a>	

[Click here to find a file you've already uploaded](#)

Comments...

Cancel

Submit Assignment



