

CS 1XC3 Lab 4: Shell Scripting

Feb. 13th – 17th, 2023

Contents

1	Introduction: Shell Scripting	1
2	Activity: Simple Shell Script	1
3	Objectives	2
4	Grading	2

1 Introduction: Shell Scripting

Ever need to run multiple commands in the same sequence several times? Tired of all that typing? Then shell scripting is for you! Shell scripts can be used to run batches of commands, give your users a standard way to install your program, automate common tasks and simplify your workflow, and much more besides!

A simple *Hello World* shell script for Bash can be written as:

```
#!/bin/bash
echo "Hello World"
```

To execute this (or any) script:

- Save your script to a *.sh file, such as `hello.sh`
- Make the file executable:
 - `chmod +x hello.sh`
- Execute the file by specifying it's absolute or relative path:
 - `./hello.sh`

2 Activity: Simple Shell Script

Consider the following shell script:

```
#!/bin/bash

# Use First Argument As Working Directory
WDIR="$1"
cd "$WDIR"

if [ "$2" -eq 1 ] ; then
    echo "TODO Objective 1"
elif [ "$2" -eq 2 ] ; then
    echo "TODO Objective 2"
fi

echo "Finished"
```

- Download the file `lab4.txt` from Avenue. Its contents are given above.
- In your CS1XC3 repository, make a directory for “L04”, and add this file to it.

- Change the file extension to *.sh
- Ensure the file is executable, and give it a test run! Just to make sure everything is working of course. You'll notice it complains about not being given arguments!

3 Objectives

1. Remove the line `echo "TODO Objective 1"` and replace it with code that:
 - Checks if a file `file1.txt` exists in the current working directory
 - If it doesn't, create it and fill it with *nothing except* the username of whoever is running the script. (hint: use a special variable!)
 - If it does, do nothing.
2. Remove the line `echo "TODO Objective 2"` and replace it with code that:
 - If and only if `file1.txt` exists in the current working directory:
 - create a directory called `backup` in the current working directory (if it doesn't already exist) and copy `file1.txt` into it.
3. Extend the script to check if the second parameter has a value of 3. If so:
 - if and only if the file `backup/file1.txt` exists...
 - copy the file `backup/file1.txt` back into the current working directory (overwriting any `file1.txt` that may already exist there)
 - delete the file `backup/file1.txt` (but not the directory `backup`)
4. Extend the script to check if it was given a third parameter. If so:
 - Alter the script to use the third parameter as the name of the file to which all other operations are applied.
 - This would replace `file1.txt` in the previous steps.
 - make sure the script retains the ability to work as previously described if no third parameter is given (i.e., the name should default to `file1.txt`)
5. Extend the script to check if more than three parameters were given. If so:
 - Create a file `excess.txt` in the current working directory (overwrite if it already exists)
 - output each parameter after the third parameter on its own line in `excess.txt`.

4 Grading

- You'll be assigned a single mark for each step completed.
- In order to submit your work, add all of the files and directories you've created to your CS1XC3 git repository.
- Commit and push your lab to finish this laboratory exercise. Remember, work not committed will have grades omitted!