

CS 279 - Week 12 Lab Exercise

Deadline

Due by the end of lab on 2022-11-10.

How to submit

Submit the files specified below on <https://canvas.humboldt.edu>.

Purpose

To practice a bit with `find`, `tar`, `gzip` and `gunzip`, and `sed`.

Important notes

- This exercise assumes that whoever is serving as navigator has a `bin` directory in their home directory, and that this `bin` directory has been added to their `PATH` environment variable (as was set up during the Week 7 Lab Exercise).
- Work in PAIRS for this lab exercise:
 - two people at one computer,
 - one typing (driver),
 - one saying what to type (navigator),
 - both discussing along the way!

When done, the driver should e-mail the files to the navigator, so BOTH of you can EACH submit them.

- Assume, for all `bash` scripts in this course, that the following are required:
 - Start each script with the line that is considered good style (and is a CS 279 course requirement), that specifies that this script should be executed using the `bash` shell
 - After a blank line, put in one or more **comments** including at least the name of the shell script, your names, and its last modified date
 - And follow these comments with a blank line.

Lab Exercise setup

- use `ssh` to connect to the one of your accounts on `nrs-projects.humboldt.edu`
- make and protect a directory `279lab12` using the commands:

```
mkdir 279lab12
chmod 700 279lab12
```
- go into that directory using:

```
cd 279lab12
```

Problem 1

In a file named `lab12-1.txt`, put:

- your names
- your answers to the following

1 part a

Somewhere in the publicly-accessible part of the instructor's nrs-projects account, there is a file named `important-script.sh`.

Write a `find` command that will locate this file, such that:

- its result is to print its absolute pathname
- it starts searching in `~st10`
- it redirects any error messages (for directories you are not allowed to access) to `/dev/null`

1 part b

Paste in the absolute pathname your `find` command found for `important-script.sh`.

1 part c

You should be able to execute `important-script.sh -- paste` what it prints to the screen when you do so.

1 part d

Choose a common file suffix for which the navigator is pretty sure there are multiple instances of that suffix in their nrs-projects directory (for example, `.sh` or `.cpp` or `.sql` or `.txt`)

Write a `find` command that will start searching in your pair's navigator's home directory on nrs-projects, looking for all such instances of files with this suffix within the navigator's account, such that:

- its result is to print its absolute pathname
- it starts searching in the navigator's home directory
- it redirects any error messages (for directories you are not allowed to access) to `/dev/null`
- *after* that error message redirection, it pipes its result to `wc -l` so that its result to the screen is the number of such files found

HINT: Remember that you CAN use file globbing wildcards in the files you want to look for within a `find` command, BUT you need to escape them or quote them (single and double quotes both SEEM to work) so they won't be expanded too soon (so they won't be expanded before "giving" them to the `find` command).

1 part e

Paste in the number of files your `find` command found.

Submit your resulting file `lab12-1.txt`.

Problem 2

In a Bash script named `lab12-2` or `lab12-2.sh`, put commands for each of the following:

- Echo that this is the starting state (followed by a border, blank line, etc. if you would like).
- Do a `pwd` command to note where you are when the script starts.
- Choose a file, and put an `ls -l` command showing the initial state of this file.
- Choose a directory BESIDES this directory, and put two commands, `ls -ld` and then `ls -l`, showing the initial state of this directory
- Echo that you are about to use `gzip` and `gunzip` (followed by a border, blank line, etc. if you would like).
- Use `gzip` to compress your chosen file, followed by an `ls -l` command to show that you now have a gzipped result
- Use `gunzip` to uncompress that file you just compressed, followed by an `ls -l` command to show that it is now gunzipped
- Echo that you are about to use `tar` (followed by a border, blank line, etc. if you would like).
- Use `tar` to create a `.tar` file from the chosen directory, followed by an `ls -l` showing the tarred file resulting
- Change to another directory of your choice, and run `pwd` to show you are elsewhere
- Copy your `.tar` file into this current directory
- Use `tar` to unpack this tar file copy back into directory form in this different directory, followed by `pwd`, an `ls -ld`, and a `ls -l` for the now-untarred directory copy, showing that a copy of it is now in this different directory.

Run your `lab12-2` or `lab12-2.sh`, redirecting its output into `lab12-2-test.txt`, and submit your `lab12-2` or `lab12-2.sh` and the resulting `lab12-2-test.txt`

Problem 3

In a file `lab12-3.txt`:

- put your names
- put your answers for each of the following

3 part a

Write a `sed` command whose output will be the contents of `play.txt` except with the first instance of `moO` on each line replaced by `MOO`

3 part b

Write a `sed` command whose output will be the contents of `play.txt` except with `'MOO> '` added to the beginning of each line.

3 part c

Write a `sed` command whose output will be the contents of `play.txt` except with `/**/` added to the end

of lines containing a `grep` command.

3 part d

Write a `sed` command whose output will be just the non-blank lines of file `play.txt`.

3 part e

Write a `sed` command whose output will be everything except lines 8-22 of file `play.txt`.

3 part f

FUN FACT: you can have more than one command within a `sed` statement -- you just need to separate them with semicolons.

So, for example, the following will output everything except the first three lines of file `play.txt`, adding an exclamation point and a `/` to the end to each of those lines:

```
sed -e '1,3 d;s/$/!\//' play.txt
```

Write a single `sed` command whose output will be the contents of `play.txt` except with the first instance of `red` on each line replaced by `RED!` and all instances of `black` replaced by `DARKNESS`

Submit your resulting `lab12-3.txt`

Submit these files to Canvas:

- `lab12-1.txt`
- `lab12-2` or `lab12-2.sh`
- `lab12-2-test.txt`
- `lab12-3.txt`