Lecture 1: Course overview + the shell

Navigating in the shell

```
missing:~$ ls -l /home
drwxr-xr-x 1 missing  users  4096 Jun 15  2019 missing
```

First, the `d` at the beginning of the line tells us that missing is a directory.

Then follow three groups of three characters (`rwx`).These indicate what permissions the owner of the file (`missing`), the owning group (`users`), and everyone else respectively have on the relevant item. A `-` indicates that the given principal does not have the given permission.

> Above, only the owner is allowed to modify (`w`) the missing directory (i.e., add/remove files in it). To enter a directory, a user must have “search” (represented by “execute”: `x`) permissions on that directory (and its parents). To list its contents, a user must have read (`r`) permissions on that directory.

If you ever want more information about a program’s arguments, inputs, outputs, or how it works in general, give the `man` program a try. It takes as an argument the name of a program, and shows you its *manual page*.

Connecting programs

The simplest form of redirection is `<` file and `>` file. These let you rewire the input and output streams of a program to a file respectively:

```
missing:~$ echo hello > hello.txt
missing:~$ cat hello.txt
hello
missing:~$ cat < hello.txt
hello
missing:~$ cat < hello.txt > hello2.txt
missing:~$ cat hello2.txt
hello
```

 You can also use `>>` to append to a file.

The `|` operator lets you “chain” programs such that the output of one is the input of another:

```
missing:~$ ls -l / | tail -n1
drwxr-xr-x 1 root  root  4096 Jun 20  2019 var
missing:~$ curl --head --silent google.com | grep --ignore-case content-length | cut --delimiter=' ' -f2
219
```

Exercise

Why does bash test.sh work but not ./test.sh?

Running bash test.sh works. But running ./test.sh doesn't work. Why is this? $PATH Adams-MacBook-Pro:~ adamzerner$ echo $PATH /Users/adamzerner/.rvm/gems/ruby-2.3.0/bin:/Users/adamzerner/.rvm/gem...

https://stackoverflow.com/questions/37358543/why-does-bash-test-sh-work-but-not-test-sh

Permissions

The Unix-like operating systems, such as Linux differ from other computing systems in that they are not only multitasking but also multi-user. What exactly does this mean? It means that more than one user can be operating the computer at the same time.

http://linuxcommand.org/lc3_lts0090.php

Shebang

The #! syntax used in scripts to indicate an interpreter for execution under UNIX / Linux operating systems. Most Linux shell and perl / python script starts with the following line: OR OR OR OR It is called a shebang or a "bang" line. It is nothing but the absolute path to the Bash interpreter.

https://bash.cyberciti.biz/guide/Shebang

Quoting (Bash Reference Manual)

3.1.2 Quoting Quoting is used to remove the special meaning of certain characters or words to the shell. Quoting can be used to disable special treatment for special characters, to prevent reserved words from being recognized as such, and to prevent parameter expansion.

https://www.gnu.org/software/bash/manual/html_node/Quoting.html

Single quotes and double quotes