

1. What unix-based OS you are using (Ubuntu, Fedora, OSX, etc...)?

Command:

`uname -a`

Linux ISIAHJOHNSO02B6 4.4.0-43-Microsoft #1-Microsoft Wed Dec 31 14:42:53  
PST 2014 x86\_64 x86\_64 x86\_64 GNU/Linux

2. What are command line options (also called parameters)?

Single letters preceded by an hyphen used to include the following option as appropriate with its commands.

The parameter:

TMRJIJ@ISIAHJOHNSO02B6:~\$

3. What are ALL the files in your home directory (the directory you start in) whose name contains the word 'bash'? What command did you use?

`.bash_logout`

`.bashrc`

Command:

`ls -a`

4. Create a new directory in your home and call it Lab1.

Command:

`mkdir Lab1`

5. Create a directory inside of Lab1 named Lab1Sub while still in the home directory.

Command:

`mkdir Lab1/Lab1Sub`

6. Create a file inside Lab1Sub named readme.please.txt, the file can be empty

Command:

`cat > Lab1/Lab1Sub/readme.please.txt`

7. What is the full (absolute) path of Lab1Sub? What command did you use?

Command:

`ls -ld Lab1/Lab1Sub*`

Path:

Lab1/Lab1Sub

8. If your present working directory is Lab1, name three 1 line commands that will return you to your home directory?

**Command:**

```
cd  
cd ..  
cd ~
```

9. Copy the `readme.please.txt` (that is in the current working directory) file into your home directory (not the directory named `home`, the top level directory of your personal files)

**Command:**

```
sudo cp readme.please.txt /
```

10. Rename this new file to be `useless.txt`

**Command:**

```
cd /  
sudo mv readme.please.txt useless.txt
```

11. Explain what `|`, `<`, `>`, and `>>` do when used on the command prompt.

`<` and `>` redirect I/Os, while `|` redirects the output of one program to another

12. What is the difference between `*` and `?` in filename wildcards?

`*` - all possible combinations; `?` must match the exact position

13. What command would you type in order to determine the number of lines in `somefile.txt` that do NOT contain `HI hi Hi` or `hi`? Hint, `grep` is your friend.

**Command:**

```
grep -vci "hi" Somefile.txt
```

14. What 1 line command would you type to generate a file `names.txt` that contains a list of users currently on the system sorted lexicographically (default order of sort) (hint: who is on the system)?

**Command:**

```
users sort > names.txt
```

15. What does the `-r` option do for the `rm` command? How about `-f`? Hint: use `man`.

**-r** - Attempt to remove the file hierarchy rooted in each file argument. The `-R` option implies the `-d` option. If the `-i` option is specified, the user is prompted for confirmation before each directory's contents are processed (as well as before the attempt is made to remove the directory). If the user does not respond affirmatively, the file hierarchy rooted in that directory is skipped.

**-f** - Attempt to remove the files without prompting for confirmation, regardless of the file's permissions. If the file does not exist, do not display a diagnostic message or modify the exit status to reflect an error. The `-f` option overrides any previous `-i` options.

16. Is executing the command `rm -r *` from your home directory a good idea? Why/why not (what would it do)?

Nope. The result would be the loss of everything in the Home directory. That may be a very bad idea.