

Bhavdeep Singh

A01314938

## Assignment 1 Shell Scripting

### NOTE:

I have added custom alias in my **.bashrc** to execute my scripts as commands.

This was done by following steps:-

```
vagrant@ubuntu2210:~$ vim .bashrc
vagrant@ubuntu2210:~$ |
```

Adding these aliases to **.bashrc**

```
#My custom aliases
alias rm='/home/vagrant/lgrm'
alias wrttask='/home/vagrant/wrttask'
alias rdtask='/home/vagrant/rdtask'
alias lnubr='/home/vagrant/lnubr'
```

```
vagrant@ubuntu2210:~$ vim .bashrc
vagrant@ubuntu2210:~$ source .bashrc
vagrant@ubuntu2210:~$ |
```

You can execute the script using './' before the script name for example:-

**./script\_name**

To make the files executable I added the executable permission on all the files.

```
vagrant@ubuntu2210:~$ chmod u+x lgrm
vagrant@ubuntu2210:~$ ls -l | grep lgrm
-rwxr--r-- 1 vagrant vagrant 1622 Mar  4 05:57 lgrm
vagrant@ubuntu2210:~$ |
```

## Script 1, tasks

This script is implemented by creating two scripts and an additional script that helps to normalize the date format to a given format.

### wrttask

This script will write the tasks with due date and tags provided by the user to the tasks.txt file.

Writing tasks using wrttask

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ wrttask -d '10 March' -t BCIT Complete assignment  
vagrant@ubuntu2210:~$ wrttask -d 02-03-23 Watch a movie  
vagrant@ubuntu2210:~$ wrttask -t Exercise Go for cycling  
vagrant@ubuntu2210:~$ wrttask  
Enter Task, Exit using ^D  
GROCERIES  
Milk  
Eggs  
Bread  
Butter  
vagrant@ubuntu2210:~$ wrttask -d 'March 20'-t Exercise Go to BCIT gym  
vagrant@ubuntu2210:~$ |
```

Here I'm writing different tasks with both due date and a tag as well as only a due date and a tag. There is a task that has task description on separate lines.

If the user wants help.

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ wrttask -h  
NAME  
    wrttask - used to add tasks with optional due date and tag  
SYNOPSIS  
/home/vagrant/wrttask: line 17: basenme: command not found  
[ options ] task...  
OPTIONS  
    -d, --due_date  
        This option adds a due date to the task  
    -t, --tags  
        This option adds tags to the task  
USAGE  
    wrttask -d '10 March' -t BCIT Complete assginment  
    wrttask -t URGENT Visit Doctor  
vagrant@ubuntu2210:~$
```

## rdtask

To read all the tasks:-

```
vagrant@ubuntu2210:~$ rdtask
vagrant@ubuntu2210:~$ |
```

```
-----
due: 03/10/2023 ##BCIT Complete assignment
-----
due: 03/23/2002 Watch a movie
-----
##Exercise Go for cycling
-----
GROCERIES
Milk
Eggs
Bread
Butter
-----
due: 03/20/2023 Exercise Go to BCIT gym
-----
(END)
```

To read a specific task with tag BCIT.

```
vagrant@ubuntu2210: ~
vagrant@ubuntu2210:~$ rdtask
vagrant@ubuntu2210:~$ rdtask -t BCIT
vagrant@ubuntu2210:~$ |
```

```
vagrant@ubuntu2210: ~  
  
-----  
due: 03/10/2023 ##BCIT Complete assignment  
-----  
(END)
```

To read a task with a due date

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ rdtask -d '10 March'  
vagrant@ubuntu2210:~$
```

```
vagrant@ubuntu2210: ~  
  
-----  
due: 03/10/2023 ##BCIT Complete assignment  
-----  
(END)
```

If the user wants help

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ rdtask -h  
NAME  
    rdtask - used to find tasks with optional due date and tag  
SYNOPSIS  
    rdtask [ options ] task...  
OPTIONS  
    -d, --due_date  
        This option finds tasks using a due date  
    -t, --tags  
        This option finds tasks using tags  
USAGE  
    rdtask -d '10 March'  
    rdtask -t Shopping  
    rdtask  
vagrant@ubuntu2210:~$ |
```

## formatdate

This script normalizes all the valid date formats provided to a single standard date format(mm/dd/yyyy).

For example:-

“10 March” => “03/10/2023”

This script is used in rdtask and wrttask to normalize the format of dates to a standard format.

## Script 2, remove log

### lgrm

To write an alias so that when you type **rm** the script executes.

```
vagrant@ubuntu2210:~$ vim .bashrc
vagrant@ubuntu2210:~$ |
```

```
#My custom aliases
alias rm='/home/vagrant/lgrm'
```

```
vagrant@ubuntu2210:~$ vim .bashrc
vagrant@ubuntu2210:~$ source .bashrc
vagrant@ubuntu2210:~$ |
```

Select the files you want to delete; you can delete multiple files in single command.

I want to delete two files test1 and test2 I can use the command rm which will delete the files and store the log of the deleted files in ~/.remove\_log file.

```
vagrant@ubuntu2210: ~
vagrant@ubuntu2210:~$ rm test1 test2
vagrant@ubuntu2210:~$ cat ~/.remove_log
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test1
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test2
vagrant@ubuntu2210:~$ |
```

If the user wants to delete the file without saving the log using **-s** option.

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ rm -s test3  
vagrant@ubuntu2210:~$ cat ~/.remove_log  
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test1  
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test2  
vagrant@ubuntu2210:~$ |
```

If the user wants help

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ rm -h  
NAME  
    lgrm - used to delete the files  
SYNOPSIS  
    lgrm [ options ] file...  
OPTIONS  
    -s, --silent  
        This option deletes the file without updating the information to the log  
    -h, --help  
        This option is used to provide the user help for using the command  
USAGE  
    lgrm file1 file2  
    lgrm -s file1  
vagrant@ubuntu2210:~$ |
```

If the file doesn't exist it will show an error.

Here fake.txt doesn't exist.

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ rm notes.txt sample.txt fake.txt  
[2023-03-05T07:35:35+0000]: ERROR: fake.txt Doesn't exist  
vagrant@ubuntu2210:~$ cat ~/.remove_log  
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test1  
Sun Mar  5 07:27:11 AM UTC 2023: vagrant: test2  
Sun Mar  5 07:35:35 AM UTC 2023: vagrant: notes.txt  
Sun Mar  5 07:35:35 AM UTC 2023: vagrant: sample.txt  
vagrant@ubuntu2210:~$ |
```

## Script 3, print a file with line numbers.

### Lnmbbr

If the user wants help.

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ lnmbbr -h  
NAME  
    lnmbbr - used to print content in the files line by line  
SYNOPSIS  
    lnmbbr [ option ] file...  
OPTIONS  
    -h, --help  
        This option is used to provide the user help for using the command  
USAGE  
    lnmbbr -h  
    lnmbbr file1 file2 file3  
vagrant@ubuntu2210:~$ |
```

If the user wants to print a file line by line.

And if the file doesn't exist there will be an error.

Here fake.txt is not a real file.

```
vagrant@ubuntu2210: ~  
vagrant@ubuntu2210:~$ lnmbbr history-of-linux.txt shell-scripting.txt fake.txt  
history-of-linux.txt  
file owner is: vagrant  
-----  
1: Linux, computer operating system created in the early 1990s by Finnish software engineer Linus Torvalds and the Free Software Foundation (FSF).  
2: While still a student at the University of Helsinki, Torvalds started developing Linux to create a system similar to MINIX, a UNIX operating system.  
3: In 1991 he released version 0.02; Version 1.0 of the Linux kernel, the core of the operating system, was released in 1994.  
-----  
shell-scripting.txt  
file owner is: vagrant  
-----  
1: A shell script is a text file that contains a sequence of commands for a UNIX-based operating system.  
2: It is called a shell script because it combines a sequence of commands, that would otherwise have to be typed into the keyboard one at a time.  
3: The shell is the operating system's command-line interface (CLI) and interpreter for the set of commands that are used to communicate with the system.  
4: A shell script is usually created for command sequences in which a user has a need to use repeatedly in order to save time.  
5: Like other programs, the shell script can contain parameters, comments and subcommands that the shell must follow.  
-----  
[2023-03-05T07:50:09+0000]: ERROR: fake.txt is not a file  
vagrant@ubuntu2210:~$ |
```