**Development Automation and Linux Lab**

**Experiment-1**

A shell is a special user program that provides an interface to the user to use operating system services. Shell accepts human-readable commands from the user and converts them into something which the kernel can understand. It is a command language interpreter that executes commands read from input devices such as keyboards or from files. The shell gets started when the user logs in or starts the terminal.

* **Man**

The **man** command provides reference information on topics, such as commands, subroutines, and files. The **man** command provides one-line descriptions of commands specified by name. The **man** command also provides information on all commands whose descriptions contain a set of user-specified keywords.

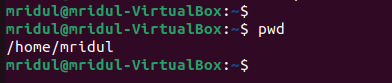
* **LS**

It is used to list information about files and directories within the file system. The ls utility is a part of the GNU core utilities package which is installed on all Linux distributions.



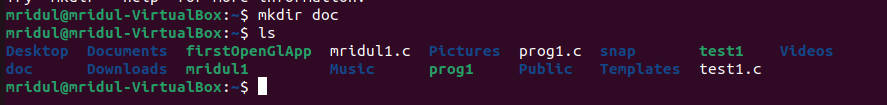
* **Pwd**

The **pwd** command writes to standard output the full path name of your current directory (from the root directory). All directories are separated by a / (slash). The root directory is represented by the first /, and the last directory named is your current directory.

****

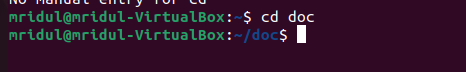
* **Mkdir**

Creates a directory or subdirectory. Command extensions, which are enabled by default, allow you to use a single mkdir command to create intermediate directories in a specified path.

****

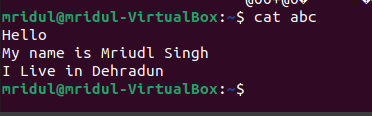
* **Cd**

Displays the name of the current directory or changes the current directory.

****

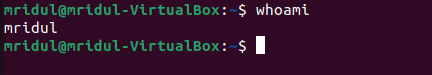
* **Cat**

The **cat** command reads each File parameter in sequence and writes it to standard output. If you do not specify a file name, the **cat** command reads from standard input. You can also specify a file name of **-** (dash) for standard input.



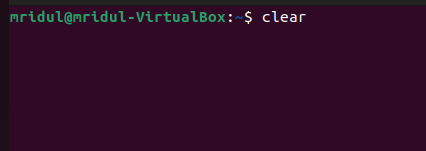
* **Whoami**

The whoami command allows Linux users to see the currently logged-in user.



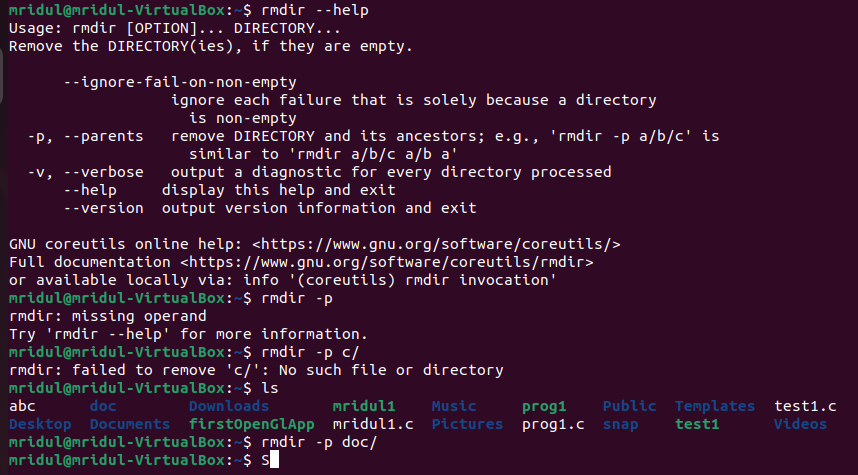
* Clear

clears the console if the console allows it.



* **Rmdir**

Removes empty directories

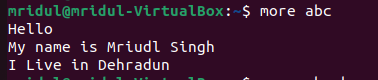


* **Touch**

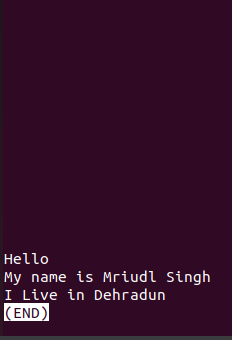
Change the timestamps

****

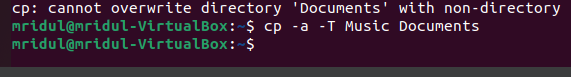
* **More**

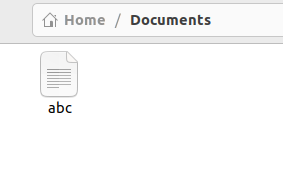
****

* **Less**

****

* **CP**

****

****

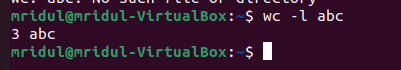
Say you want to copy a file to a new location making sure the original copy still exists in the main location. For that we do ctrl+c this is what cp does in Linux.

Now say you want to copy a file to a new location making sure the original copy is deleted from the main location. In that case you will use ctrl+x and this what mv does for you in Linux.

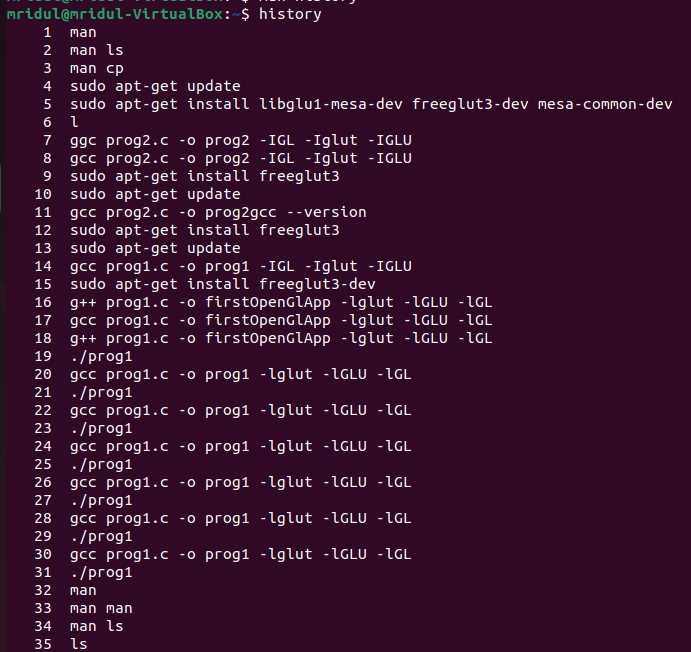
* **MV**

****

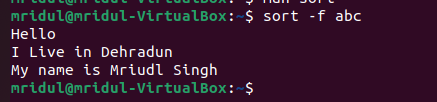
* **WC**

****

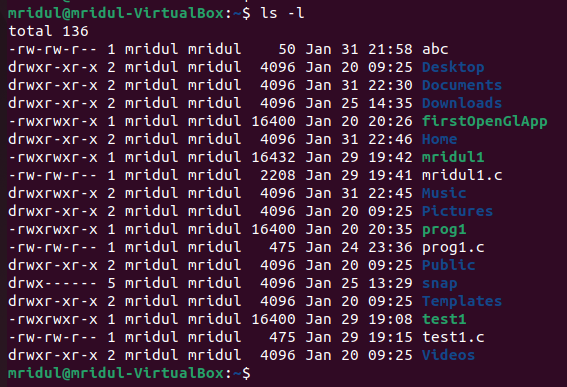
* **History**

****

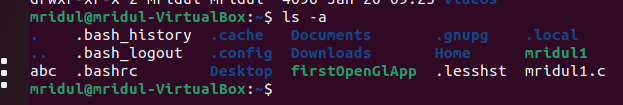
* **Sort**

****

* **ls -l**

****

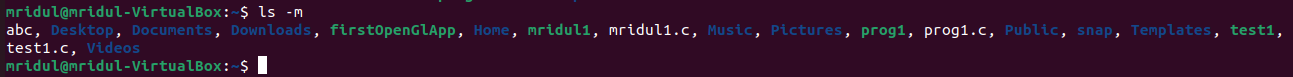
* **ls -a**

****

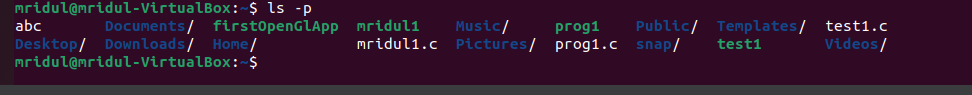
* **ls -c**

****

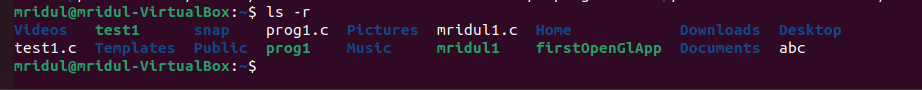
* **ls -m**

****

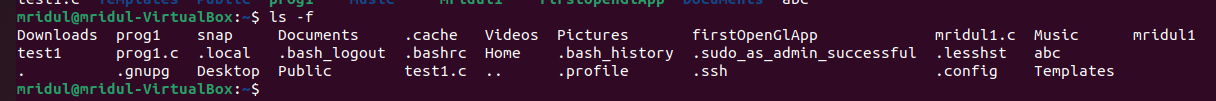
* **ls -p**

****

* **ls -r**

****

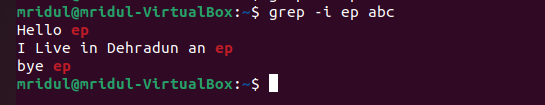
* **ls -f**

****

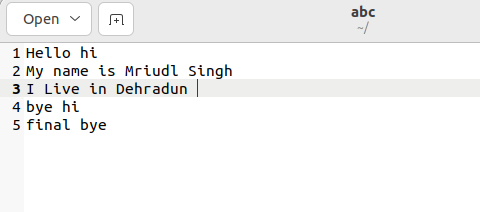
**Grep command in Unix/Linux**

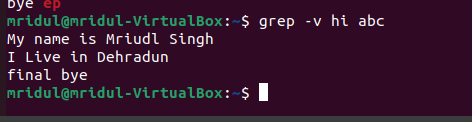
The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for global search for regular expression and print out).

i) find all the patterns that start with “ep” in a given file



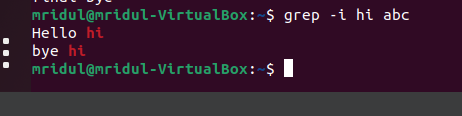
ii) Print those lines of a given text file which do not contain the word “hi”.



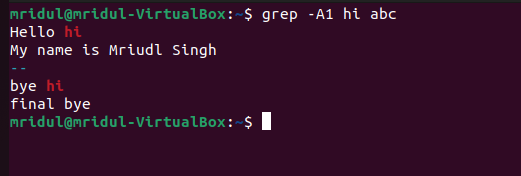


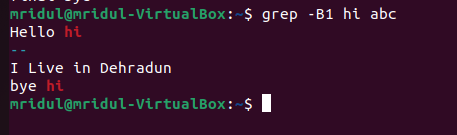
iii) Explain the usage of options -i, -A, -B, -C in grep command with examples

-A n : Prints searched line and nlines after the result.



-B n : Prints searched line and n line before the result.





-C n : Prints searched line and n lines after before the result.

