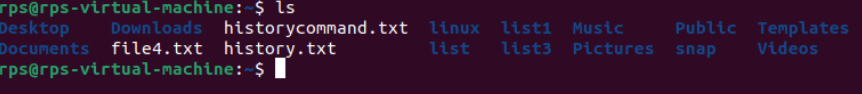
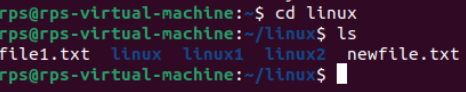
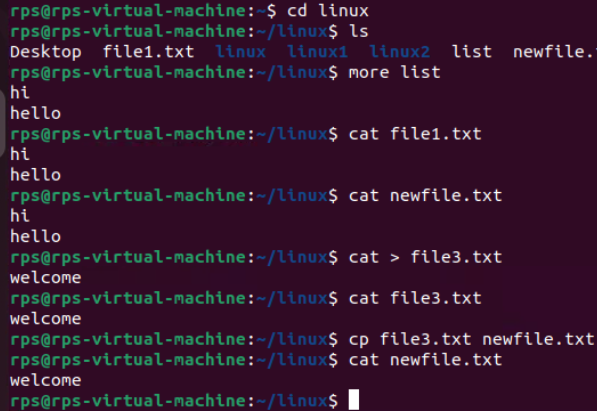
**ls-**show files in current position

****

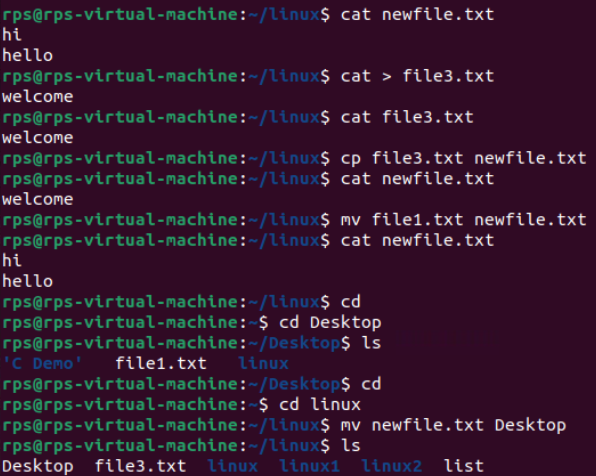
**cd-**change directory



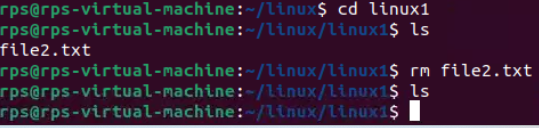
**cp-**copy file or directory



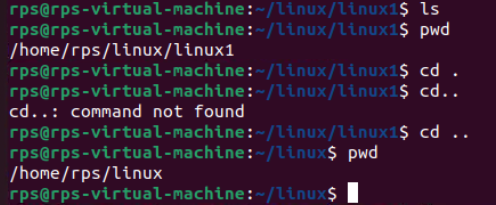
**mv-**move file or directory



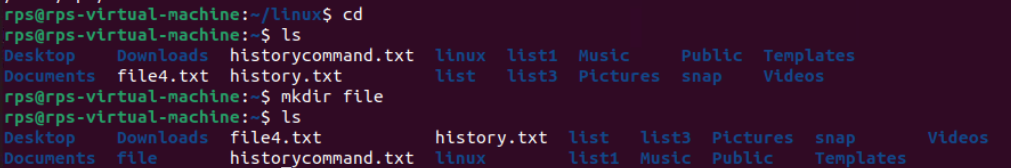
**rm-**remove file or directory



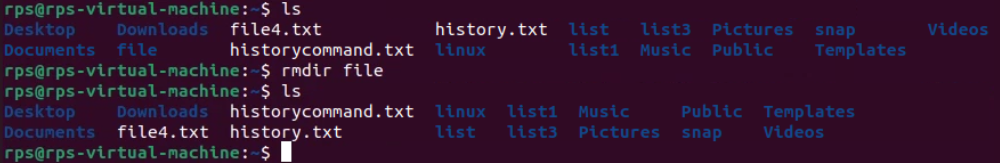
**Pwd-**show current postion



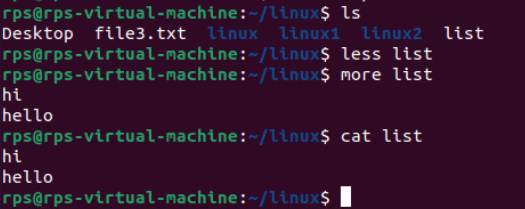
**mkdir-**create directory



**rmdir-**remove directory



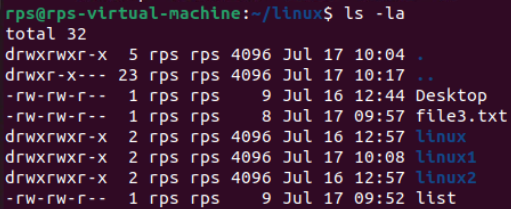
**less,more,cat-** display file contents



**ls-a-** list all files and directories, including hidden ones, in a directory



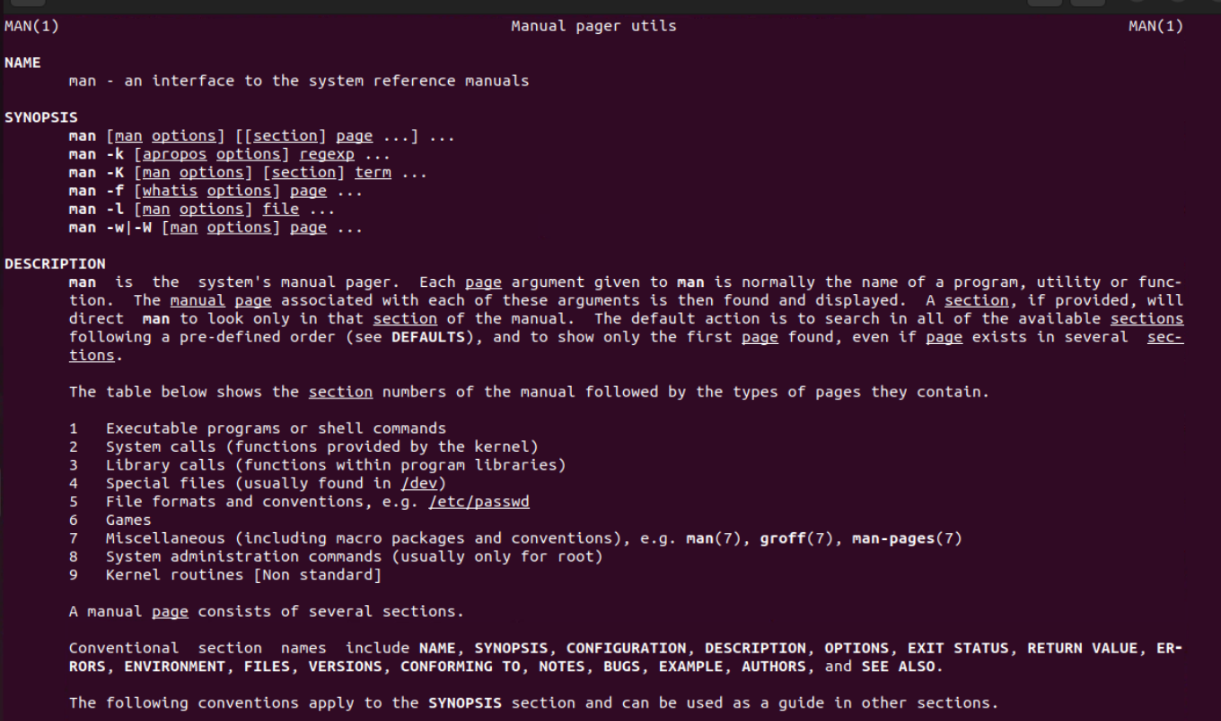
**ls-la-**list all files and directories, including hidden ones, in a long format with detailed information.



**ls-fa**- list all files and directories, including hidden ones, without following symbolic links.



**man:** Display the manual for a command or function, showing its usage, options, and documentation.



**pwd-** show current postion

**cd . -** No operation, stays in the current directory, can be used to refresh directory listing or reset shell state.

**cd .. -** Go up one level, to the parent directory.

**Relative Path:**

pwd

cd .

pwd

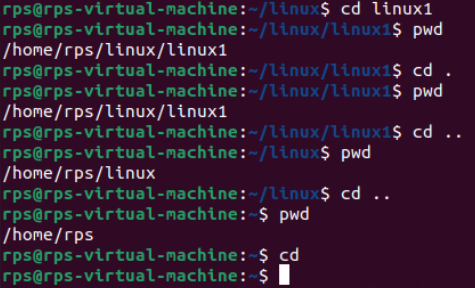
cd ..

pwd

cd ..

pwd

cd



**Ablsoute Path:**

cd

mkdir mydir

pwd

cd /Users/invite

pwd

cd /Users

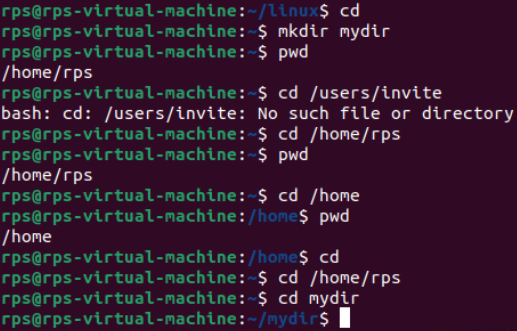
pwd

cd /

pwd

cd /Users/invite

cd ~/mydir

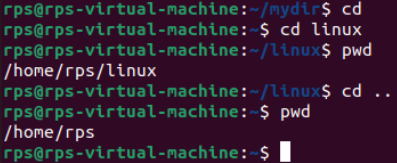


**Navigation:**

**cd (change directory): Moves you between directories.**

**Exercise: Navigate to your home directory (cd ~), then explore subdirectories like Documents (cd Documents). Use pwd (print working directory) to confirm your location. Try going back a directory with cd ... pwd (print working directory): Shows your current directory path.**

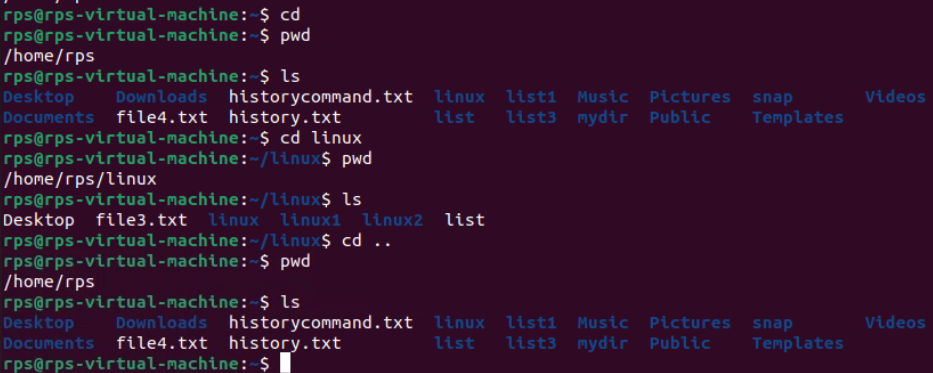
* cd-navigates to your home directory.
* pwd-shows your current directory path.
* Cd Documents -navigates to the Documents directory.
* cd ..-moves back to the parent directory.

****

**Exercise: After navigating using cd, use pwd to verify the path.**

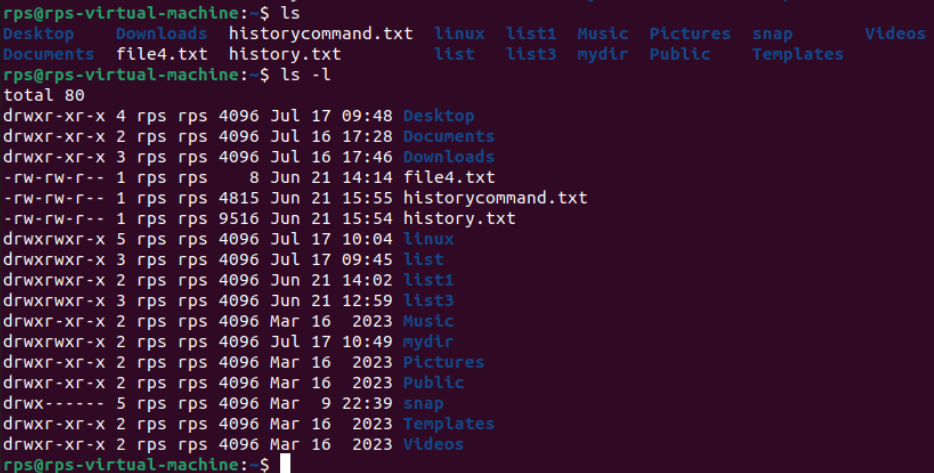
**ls (list): Lists files and directories in the current directory.**

* cd - Navigate to the home directory.
* pwd - Verify the current directory path.
* ls - List files and directories in the current directory.
* cd Documents - Navigate to the Documents directory.
* pwd - Verify the current directory path.
* ls - List files and directories in the Documents directory.
* cd .. - Go back to the parent directory.
* pwd - Verify the current directory path.
* ls - List files and directories in the current directory.



**Exercise: Use ls in your home directory and note the listed items. Try ls -l (long format) for detailed information like permissions, owner, and size.**

* ls-lists files and directories in the current directory.
* ls -l-lists files and directories in long format, showing detailed information like permissions, owner, and size.



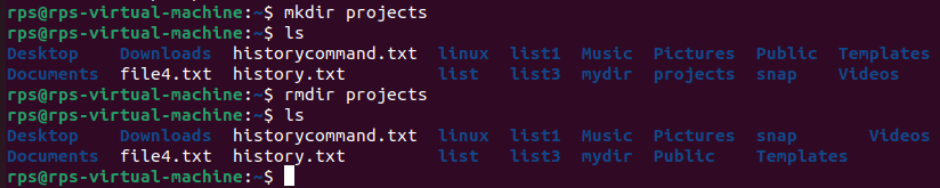
**File and Directory Management:**

**mkdir (make directory): Creates a new directory.**

**Exercise: Create a new directory called "Projects" (mkdir Projects). Use ls to confirm its existence.**

**rmdir (remove directory): Deletes an empty directory.**

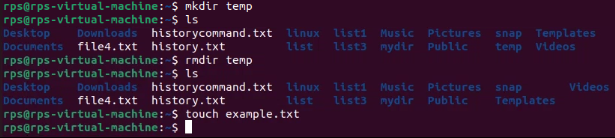
* mkdir-Projects creates a new directory named Projects.
* ls-lists files and directories in the current directory.
* rmdir- Projects removes the empty directory named Projects.



**Exercise: Make a directory named "temp" (mkdir temp). Delete it after verifying its existence with ls (rmdir temp).**

**touch (create file): Creates an empty file.**

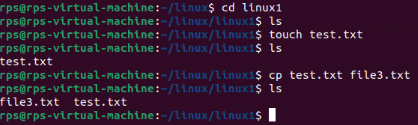
* mkdir temp-creates a new directory named temp.
* ls- lists files and directories in the current directory.
* rmdir temp-removes the empty directory named temp.
* touch-example.txt creates an empty file named example.txt.



**Exercise: Create a file called "test.txt" (touch test.txt). Use ls to see it listed.**

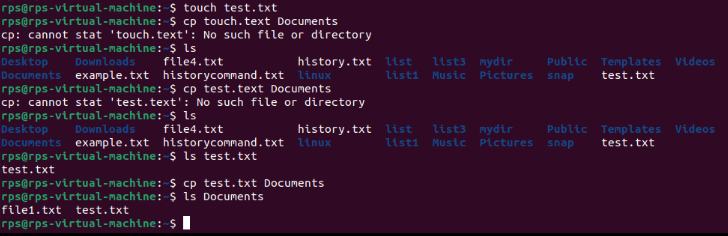
**cp (copy): Copies a file or directory to another location.**

* touch-test.txt creates an empty file named test.txt.
* ls-lists files and directories in the current directory, allowing you to see test.txt.
* cp-test.txt file3.txt copies the file test.txt to a new file named file3.txt.



**Exercise: Copy "test.txt" to your Documents directory (cp test.txt Documents). Verify the copy with ls Documents.**

* cp- test.txt Documents/ copies the file test.txt to the Documents directory.
* ls- Documents lists the files and directories in the Documents directory, allowing you to verify that test.txt has been copied there.

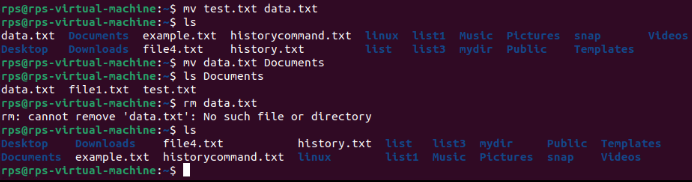


**mv (move/rename): Moves or renames a file or directory.**

**Exercise: Rename "test.txt" to "data.txt" (mv test.txt data.txt). Use ls to confirm the change. You can also move files to a different directory (e.g., mv data.txt Documents).**

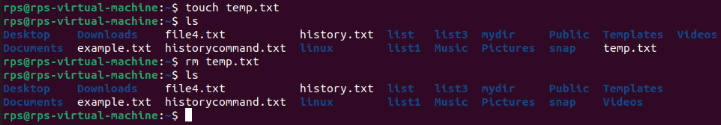
**rm (remove): Deletes files or directories (use with caution!).**

* mv test.txt data.txt: Renames test.txt to data.txt.
* ls: Lists files and directories in the current directory to confirm the rename.
* mv data.txt Documents/: Moves data.txt to the Documents directory.
* ls Documents: Lists files and directories in the Documents directory to confirm the move.
* rm data.txt: Deletes (removes) data.txt. Use with caution as this action is irreversible without backups.
* ls: Lists files and directories in the current directory to confirm the deletion.



**Exercise: Important: Only use this after creating a test file (e.g., touch temp.txt). Delete "temp.txt" with rm temp.txt. Never use rm -rf without understanding the risks!**

the rm command is powerful and irreversible. It permanently deletes files and directories without moving them to the trash or recycling bin.



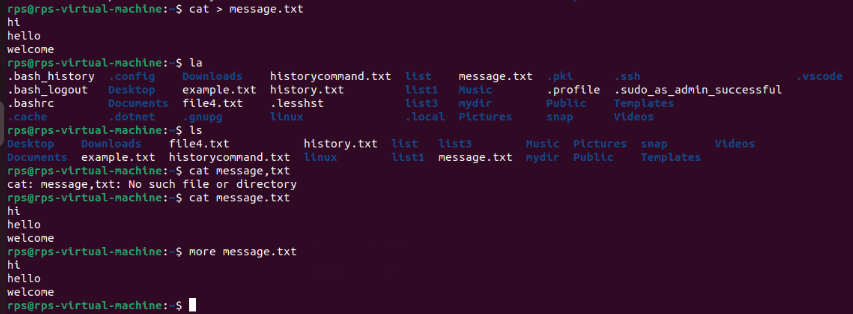
**File Viewing and Permissions:**

**cat (concatenate): Displays the contents of a text file.**

**Exercise: Create a text file named "message.txt" with some content (e.g., using a text editor). Then, use cat message.txt to view its contents.**

**more (pager): Displays a file's contents one screen at a time (useful for long files).**

* cat : Displays the contents of a text file.
* more : Displays a file's contents one screen at a time (useful for long files).

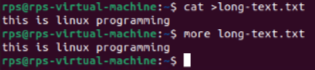


**Exercise: Create a larger text file (e.g., "long\_text.txt") and use more long\_text.txt to navigate through its content page by page.**

**less (pager): Similar to more, but allows you to move backward in the file.**

* Both more and less are essential Unix commands for viewing large text files page by page.
* less offers additional functionality like backward navigation, which can be useful for detailed review.

**More:**



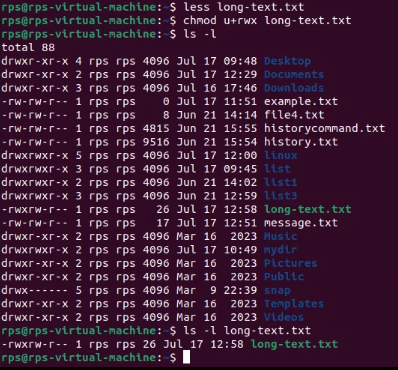
**Less:**



**Exercise: Use less with "long\_text.txt" to try moving backward using the Up arrow key.**

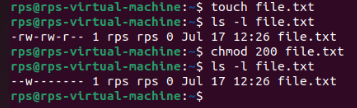
**chmod (change mode): Modifies file permissions (owner, group, others) for read, write, and execute access.**

* less is a versatile pager that allows interactive viewing and navigation through large text files.
* chmod to change file permissions based on specific requirements, ensuring proper access control for users and groups.



**Exercise: This requires understanding permissions. Refer to the man chmod page for details. Proceed with caution when modifying permissions.**

* Create a file touch file.txt
* View current permissions ls -l file.txt
* Chane permissions to read, write, and execute for the owner chmod 700 file.txt



**Getting Help and Information:**

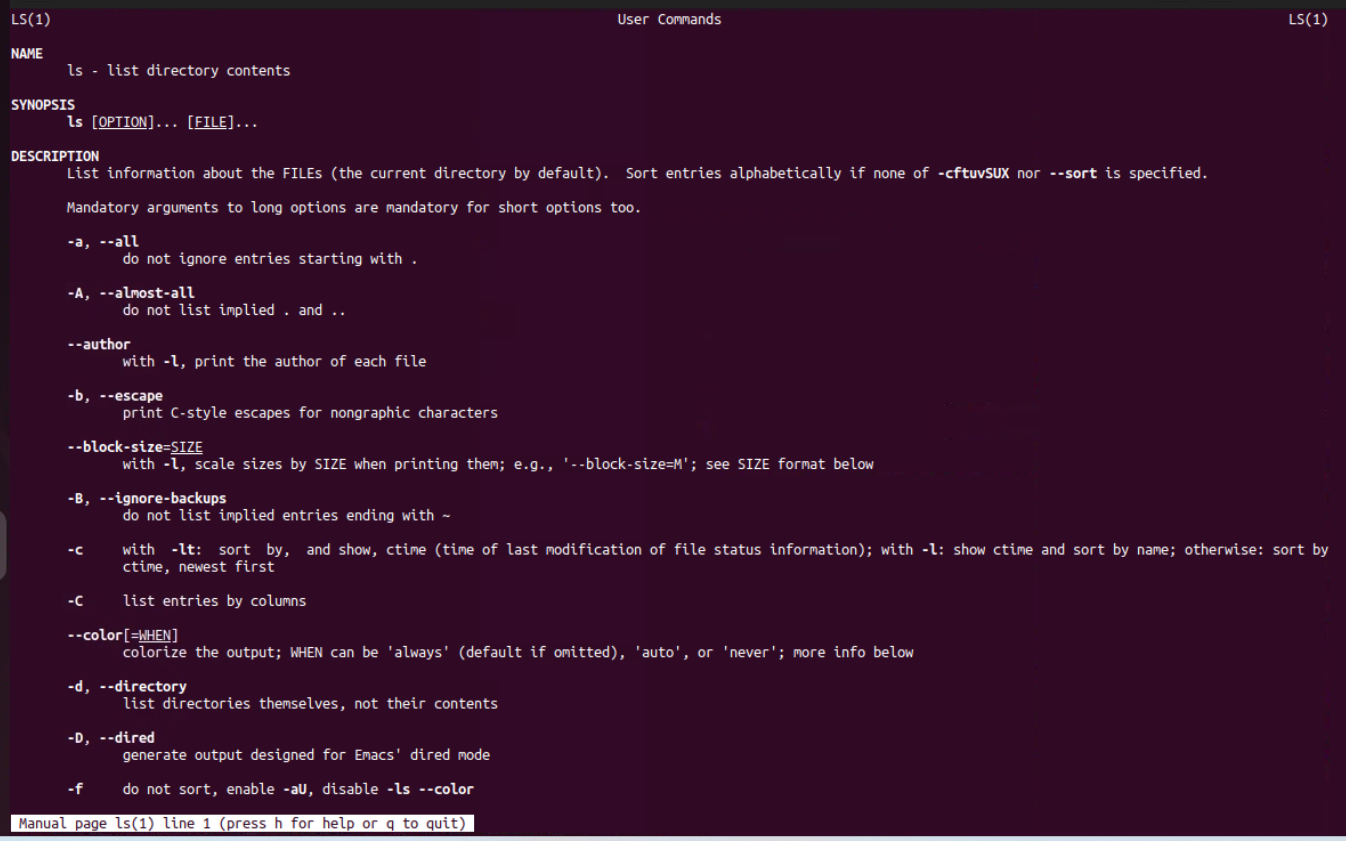
**man (manual): Provides detailed information about a command.**

**Exercise: Use man ls or man cd to learn more about these commands.**

**info (information): Another source of documentation for some commands, often more user-friendly than man.**

**man ls:** Displays the manual for the ls command, showing its usage, options, and documentation for listing files and directories**.**

**man ls:**

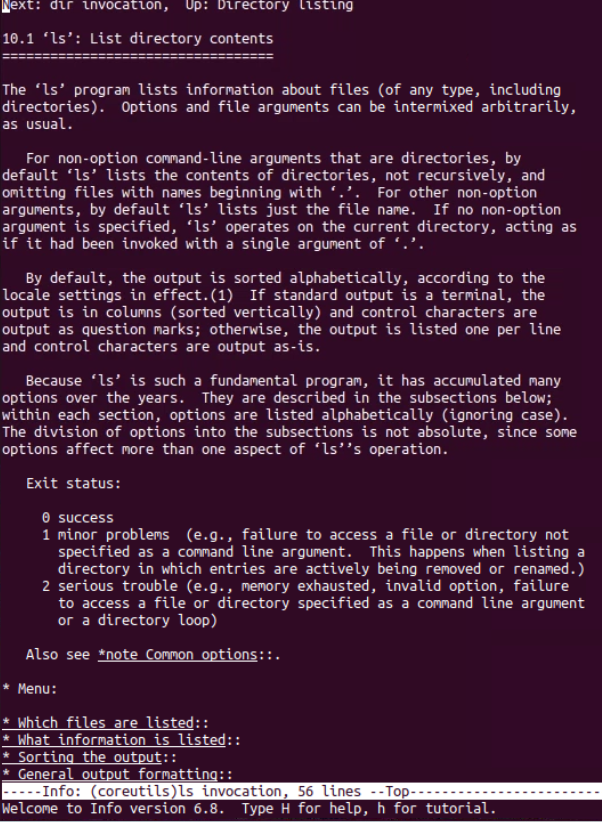
****

**Exercise: Try info ls or info cd if available on your system.**

**--help or -h (help flag): Provides a brief overview of a command's usage.**

**info ls:** Displays detailed information about the ls command, including its usage, options, and examples, in an interactive hypertext format.

**info ls:**

****

**Exercise: Use ls --help or ls -h to see the basic usage options for ls.**

**ls --help:** Displays a brief summary of the ls command's usage, options, and arguments, providing a quick reference for the command's syntax and features

**ls –help**



**Navigation and File Manipulation:**

**cd ~ && mkdir exercises && cd exercises: Navigate to your home directory, create a new directory named "exercises," and then move into it.**

* cd ~: Moves you to your home directory (~).
* mkdir exercises: Creates a new directory named exercises within your home directory.
* cd exercises: Changes your current directory to exercises that you just created.



**cp ../data.txt . (assuming "data.txt" exists in the parent directory): Copy a file named "data.txt" from the parent directory into your current "exercises" directory.**

* cp: Command to copy files.
* ../data.txt: Specifies the path to the file data.txt in the parent directory
* .: Represents the current directory, which is exercises in this case.

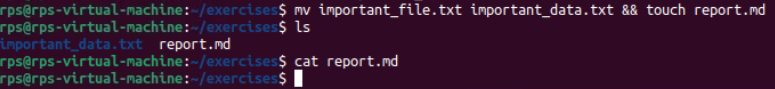
**mv important\_file.txt important\_data.txt && touch report.md: Rename a file named "important\_file.txt" to "important\_data.txt" and create a new Markdown file named "report.md" within the "exercises" directory.**

* mv important\_file.txt important\_data.txt: Renames the file "important\_file.txt" to "important\_data.txt". This assumes both files are in the current directory.
* &&: Combines multiple commands to run sequentially if the previous command succeeds.
* touch exercises/report.md: Creates a new file named "report.md" within the "exercises" directory using the touch command, which creates an empty file if it doesn't already exist.



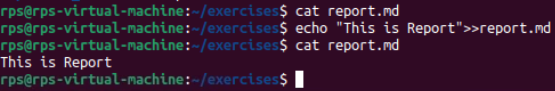
**cat report.md (assuming the file is empty): View the contents (which should be empty) of the "report.md" file using cat.**

cat report.md to view its contents, you won't see any output in the terminal because there is no content in the file. The command will execute without displaying anything, indicating that the file is indeed empty.



**echo "This is a report" >> report.md: Append a line of text "This is a report" to the "report.md" file using redirection ('>>').**

Appends the text "This is a report" to the "report.md" file using redirection ('>>').



**less report.md: Use less to view the contents of the "report.md" file, which now contains the appended text.**

Uses the less command to view the contents of the "report.md" file, which now contains the appended text.

****

**Finding and Organizing Files:**

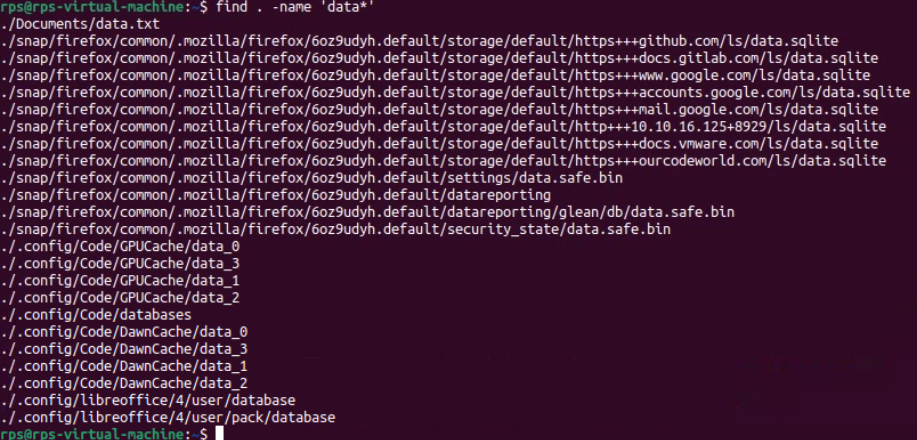
**ls -l | grep 'txt$': List all files in the current directory with the long format (-l) and filter the output using grep to only show files ending with the ".txt" extension.**

* ls -l: Lists files in long format, providing detailed information about each file.
* |: Pipe operator sends the output of ls -l to the input of grep.
* grep '\.txt$': Filters the output of ls -l to only show lines that end with ".txt".
* \.: Escapes the dot character to match a literal dot.
* txt$: Matches lines ending with "txt".



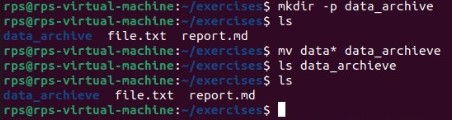
**find . -name 'data\*': Search for all files starting with "data" (including "data.txt", "data\_backup.csv", etc.) recursively within the current directory and its subdirectories using find.**

* find .: Starts the search from the current directory (.).
* -namedata\*: Specifies the pattern to match against the filenames. Here, data\* means filenames starting with data followed by any characters.

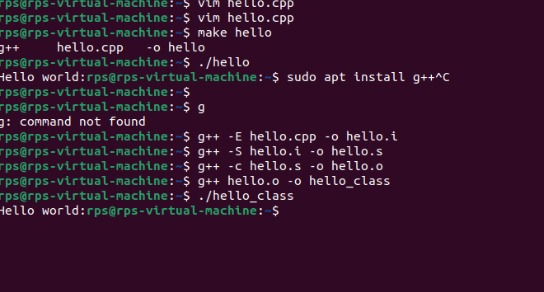


**mv data\* data\_archive/: Move all files starting with "data" into a new directory named "data\_archive" (create it if it doesn't exist).**

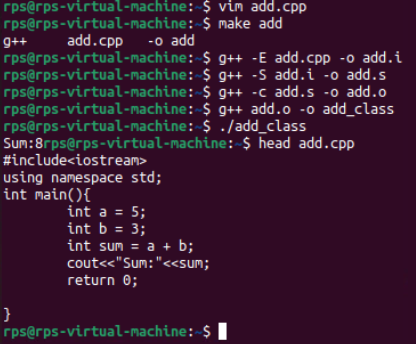
* mkdir -p data\_archive: Creates the directory "data\_archive". The -p flag ensures that the command does not throw an error if the directory already exists.
* mv data\* data\_archive/: Moves all files starting with "data" (data\*) into the "data\_archive" directory.

H

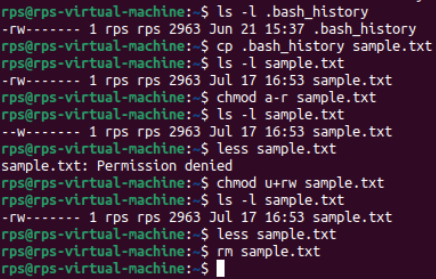
**Hello-world program:**

****

**Addition of two numbers:**

****

**Permissions:**

****