**ls:**

**[hdfs dfs -ls <path>]**

**List the contents of a directory.**

**cp:**

**[hdfs dfs -copyFromLocal <source> <destination>]**

**[hdfs dfs -copyFromLocal test.txt mlds/this\_is\_the\_new\_test.txt ]**

**[hdfs dfs -copyToLocal wc/output\_spark/part\_00000]**

**Copies files or directories**

**rm:**

**[hdfs dfs -rm test.txt]**

**[hdfs dfs -rm -r mlds]**

**Removes files or directories, -r for removing directory.**

**mkdir:**

**[hdfs dfs -mkdir msia]**

**Creates a new directory**

**emacs:**

**[emacs test.txt]**

**Creates a file writer in the local server (wolf in this case)**

**more:**

**[more test.txt]**

**[more part\_00000]**

**Displays text files one screen at a time, used in wolf local, not in hdfs.**

**find:**

**Searches for files and directories based on criteria.**

**touch:**

**[hdfs dfs -touchz <path>]**

**Creates a new empty file or updates the timestamps of an existing file.**

**tail:**

**[hdfs dfs -tail <file>]**

**Displays the last part of a file.**

**chown:**

**[hdfs dfs -chown <OWNER> [:<GROUP>] <path>]**

**Changes the owner of a file or directory.**

**chmod:**

**[hdfs dfs -chmod <permissions> <path>]**

**Changes the permissions of a file or directory.**

ps

kill

grep

uptime, top

export environment variables

python and env variables

print(os.environ)

os.environ.get('Nonexistent-variable')

os.environ['USER'] = 'Bob'

git

[remote "origin"]

url = ssh://github.com/qGZQFjLYXGa4bEPaz/videspan.git

fetch = +refs/heads/\*:refs/remotes/origin/\*

[branch "main"]

remote = origin

merge = refs/heads/main

Host hc-demo github.com

HostName github.com

User git

IdentityFile ~/.ssh/hc-demo

git init -b main (older don’t support -b)

Initializes a new Git repository locally in the current directory and sets the default branch name to **main**. The **-b** option is used with newer versions of Git to specify the branch name directly during initialization.

git status

touch REAMDE.md

Creates a new file named **README.md** in the current directory.

git add README.md

Stages the **README.md** file for the next commit, indicating that its changes should be included. 工作区转移到暂存区

git commit -m “initial”

Commits the staged changes to the repository with a commit message "initial".

git branch -a

Lists all branches in the repository, including both local and remote branches.

git show-ref

Displays all references (branches, tags) in the repository, along with their corresponding commit hashes.

create ssh pair

Refers to generating an SSH key pair (a private key and a public key). This is typically done with the command **ssh-keygen**. The private key remains with the user and should be kept secure, while the public key can be shared.

upload public to github

Instructs to add the generated public SSH key to your GitHub account under the "SSH and GPG keys" section in account settings. This allows GitHub to recognize and authenticate your computer.

create config (chmod 400)

Suggests creating an SSH configuration file (usually named **config** in the **~/.ssh** directory) to specify details about the SSH connection to GitHub and sets the file's permissions to **read-only** for the owner with **chmod 400**. This file includes:

* **Host hc-demo github.com**: Defines an SSH configuration block for GitHub, giving it an alias **hc-demo**. This makes it easier to manage SSH keys and connections, especially if you use SSH for multiple services.
* **HostName github.com**: Specifies the actual hostname for the SSH connection; in this case, GitHub's domain.
* **User git**: The default username for Git operations over SSH.
* **IdentityFile ~/.ssh/hc-demo**: Points to the private key file used for the SSH connection.

git push origin main