Shell Scripting - Module 2

Basics of Linux commands

Mkdir module

▼ Mkdir in verbose

> Run mkdir in verbose

```
mkdir -v /data
```

- [v] refers verbose
- Running any cmd with [-v] is going to print the results of that particular cmd

> Output

```
mkdir: created directory /data
```

▼ Mkdir with parent directory

> Create parent directory and sub directory

```
mkdir -v -p /example/file
```

- [p] refers parent
- [v] refers verbose
- This cmd will create parent dir and sub dir underneath.

> Output

```
mkdir: created directory /example
mkdir: created directory /example/file
```

Less modules

▼ Using less to read files

```
less ansible.cfg
```

- less is very helpful to read big files

▼ View file with line numbers using less

```
less -N ansible.cfg
```

▼ Use less to view multiple files

```
less playbooks*
```

- Eg, how much ever files available under playbook folder can be seen using less.
    - To view next file, you can use [:n]
    - To view previews file, you can use [:p]

File module

▼ Use File to identify symbolic link

> ### Command

```
file /usr/bin/python
```

- File module can be used to determine the type of a file.

> ### Output

```
/usr/bin/python3: symbolic link to /etc/alternatives/python3
```

▼ Use File to identify the directory

> ### Command

```
file /data/ansible
```

> ### Output

```
/data/ansible/: directory
```

▼ Use File to identify the executable files

> Command

```
file /data/ansible/bin/ansible
```

- You can also use file to identify the executable of particular application.

> Output

```
/data/ansible/bin/ansible: Python script, ASCII text executable
```

Remove module

▼ Remove file

```
rm filename
```

- Removes file with prompt

```
rm -i filename
```

- [i] Prompt for every removal [user confirmation]

▼ Remove directory

```
rm dirname
```

- Remove the directory.

```
rm -r dirname
```

- Removes the directory and all the available files underneath.

▼ Remove directory but prompt for confirmation if the directory has more the 3 files.

```
rm -r -i dirname
```

  - Removes directory but prompt confirmation if the directory has more then 3 files

▼ Combine all arguments

```
rm -rfiv dirname
```

  - [r] - recursive remove
  - [f] - force removal
  - [i] - prompt if directory has more then 3 files.
  - [v] - print results in verbose

▼ Well know removal

```
rm -rf directory
```

Cd module

```
cd .
```

- [.] refers current working dir

```
cd ..
```

- [..] refers parent dir of current working dir

```
cd ../..
```

- Navigates 2 step ahead of current working dir
- For instance [pwd is /data/ansible/playbook] & I want to switch to data from here. You can use this command.

```
cd ../../../..
```

- Switch four steps ahead.
- [pwd /data/ansible/playbooks/hello/world] & I want to switch to data from here.

```
cd ~prathang
```

- The notation tilder username can be used to refer the home directory associated with the specified username
- For instance if you are in dir [/data/ansible/playbook] and you just want to go to your home dir [prathang]

Cat module

```
cat ../../register.yml
```

- pwd is [/data/ansible/playbooks/hello/world] from there I want to view a file which is under folder playbook which is two steps ahead

List module

▼ Navigate between directories and files

```
ls..
```

  - [..] refers parent dir
  - you can list parent directory file from its subdirectory.

```
ls../..
```

  - list files under playbook when you are located under [/data/ansible/playbooks/hello/world]

```
ls../playbooks
```

  - Lets say you are in directory [/data/ansible/lib] but you want to [ls] file [playbook] which is under [/data/ansible]

▼ List files & folders

Long listing

```
ls -l /usr/bin/python
```

- [l] refers long listing
- [-l] argument provides more details about each files such as file type, permissions, owner, group etc..
- Using this argument passed along with [ls] you can also view symbolic links created for files if any..

List including hidden files

```
ls -a
```

- By using [-a] it do not ignore files starting with [.] which eventually get all the hidden file details as well.

Human readable

```
ls -lh
```

- [h] human readable
- Using h with ls command gives the file size in more human readable formate [KB, MB & GB]

Combine all arguments

```
ls -alh
```

- This cmd retrive hidden files, long listing details of each file and file size in human readable.

Link module

> ## Link a file with new file

```
ln -s [target_file] [new_link_file]
```

- [s] refers symbolic link
- Using this cmd you can link a [target file] to one new file which will be created.

> ## Create link file with same name

```
ln -s [target_file]
```

- Using this cmd we are just creating a link with the same name of target file.
- Eg : ln -s /data/ansible/hello [Link will be created with the same name]
  - hello --> /data/ansible/hello
  - world --> /data/ansible/hello

Copy module

> ## Copy one file to other

```
cp [file] [file]
```

> ## Copy one dir to other

```
cp -r [dir] [dir]
```

Difference module

▼ Show the difference between 2 files

```
diff [file 1] [file 2]
```

▼ Show which file is available where?

> ### Command

```
diff [dir 1] [dir 2]
```

> ### Output

```
Only in bin: activate
Only in playbooks: hello
```

Echo module

> ### Using Escape sequences in echo with [-e]

```
echo -e "hello\nworld"
```

- [\n] refers new line
- [-e] is required to interpretate escape characters

```
echo -e "hello\tworld"
```

- [\t] giving a tab space between 2 characters.