

Once the bash shell script is running, it will first enter a while loop.

-p tells the command read for a prompt. It will then store what the user entered to the variable folder\_name.

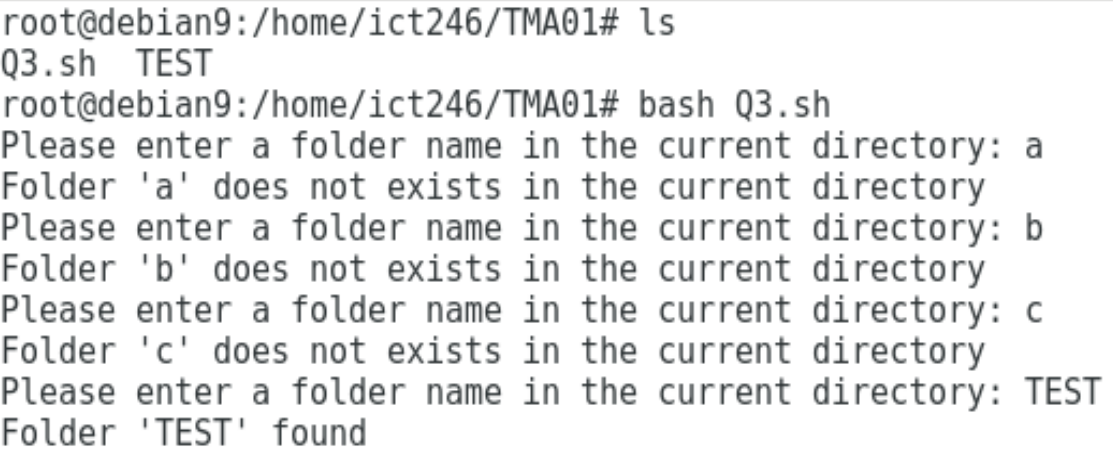
[] tells the if statement to test the expression in the brackets.

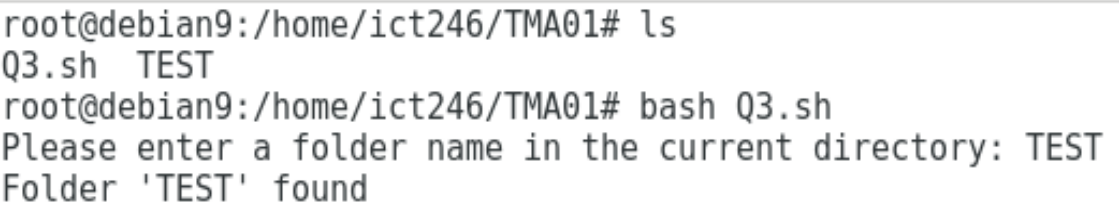
! means not.

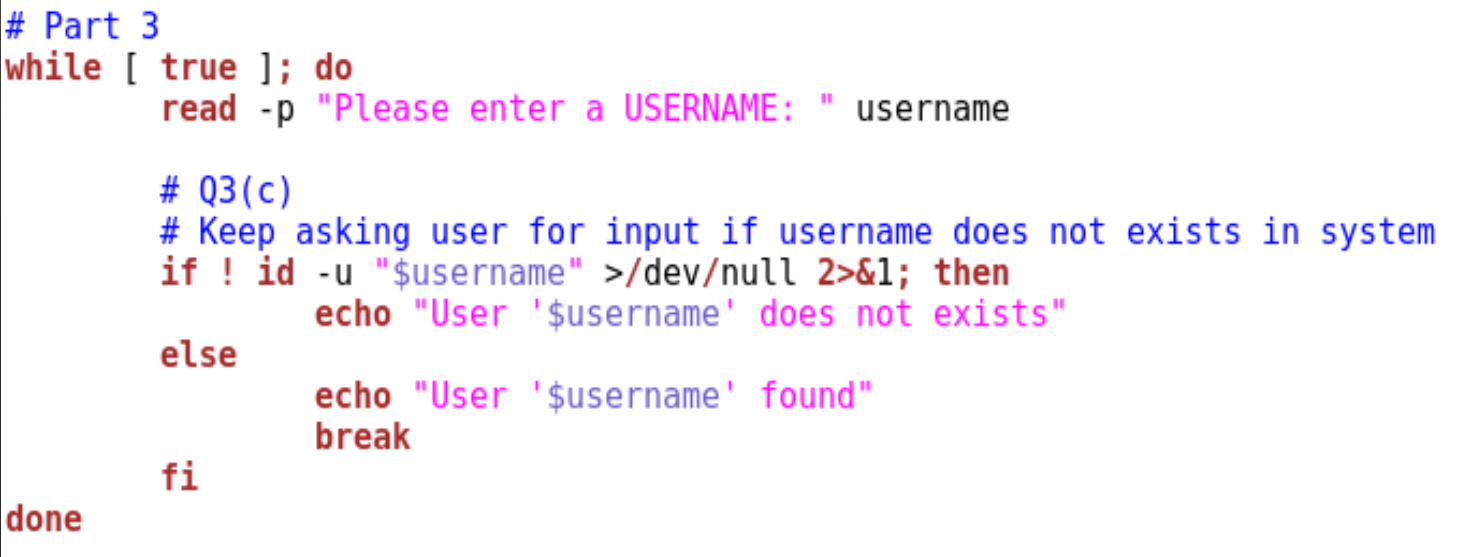
-d means directory. In this case, it refers to the folder.

Hence, the if statement is checking if the folder folder\_name exist in the current directory. If it does not exist, because of the while loop, user will continuously be prompted to enter a folder name until an existing folder name is entered. If it does exist, it will break out of the while loop.

**Result**







It will then enter a second while loop.

Similarly, the user will enter username and it will be stored in the variable username.

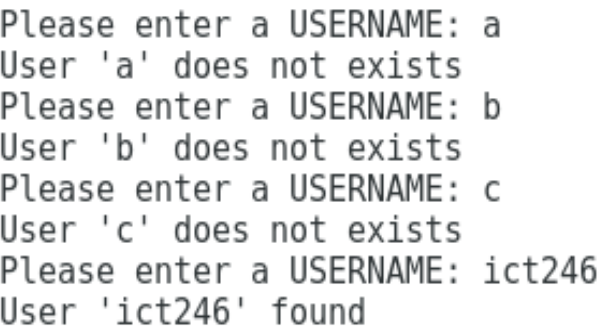
-u tells the command id to search for user.

>/dev/null means the output of id will then be redirected to /dev/null.

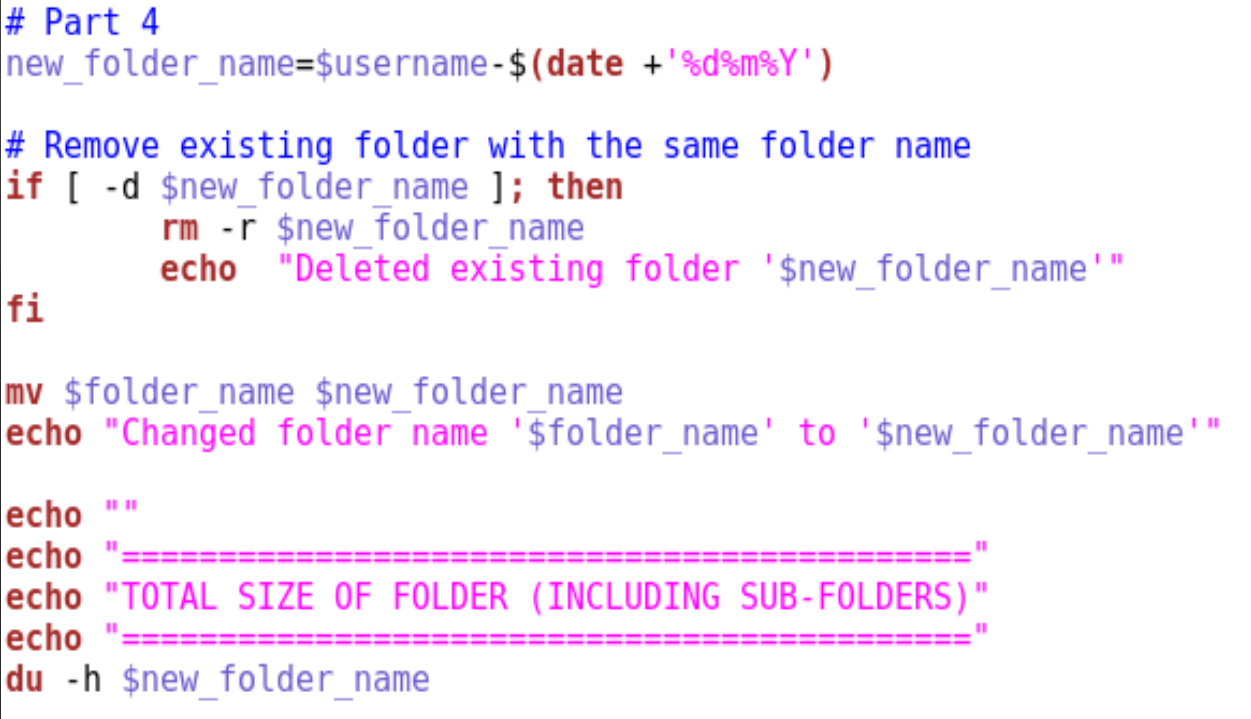
2>&1 means the Standard Error (stderr) will be redirected to whatever is set to the Standard Output (stdout).

Hence, the if statement is checking if the username exist in the system. If it does not exist, because of the while loop, user will continuously be prompted to enter a username until an existing username is entered. If it does exist, it will break out of the while loop.

**Result**







The variable new\_folder\_name is the variable username + ‘-‘ + the variable date in the format of DDMMYYY.

The if statement is checking if folder new\_folder\_name exists in the current directory.

-r tells the command rm to recursively remove the files in the folder and its sub-folders. Hence, removing existing folder new\_folder\_name, regardless of it being empty or not.

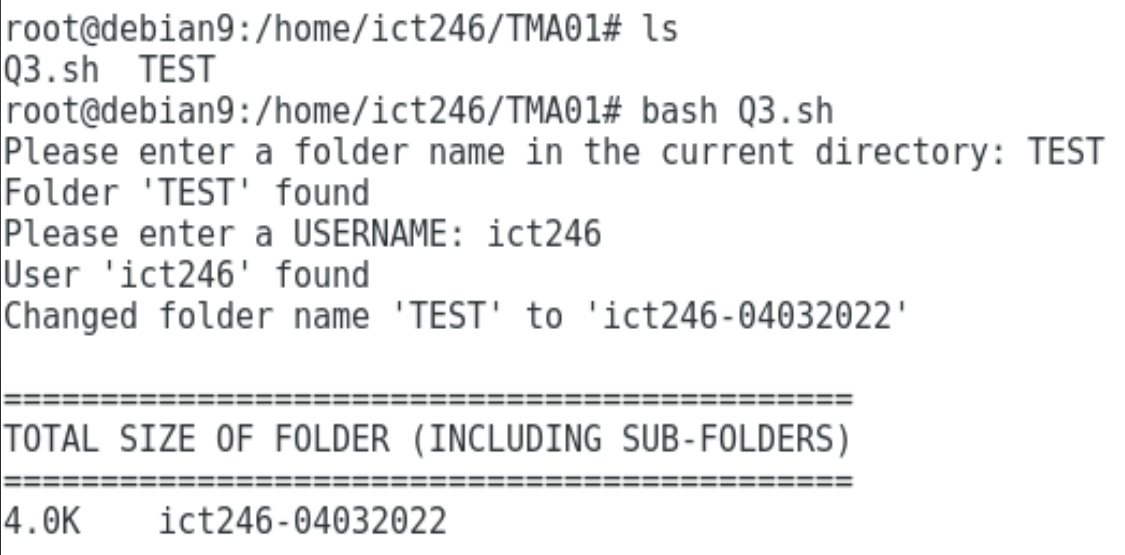
The folder will then be renamed according to the variable new\_folder\_name with the command mv.

-h tells du to display the file size in a human readable format.

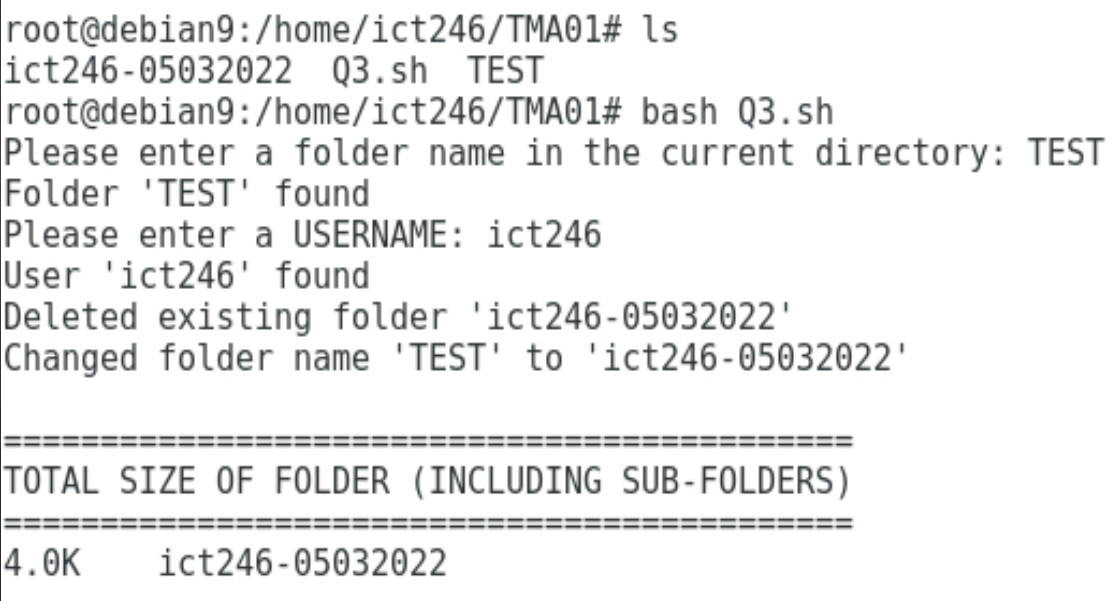
The total size of the folder, including its sub-folders, will then be displayed.

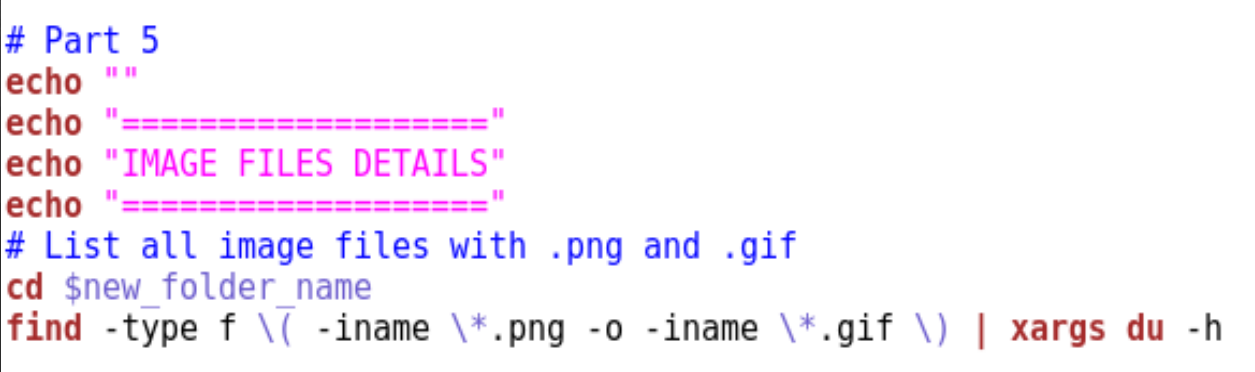
**Result**

Folder does not exist



Folder does exist





It will then cd into the folder.

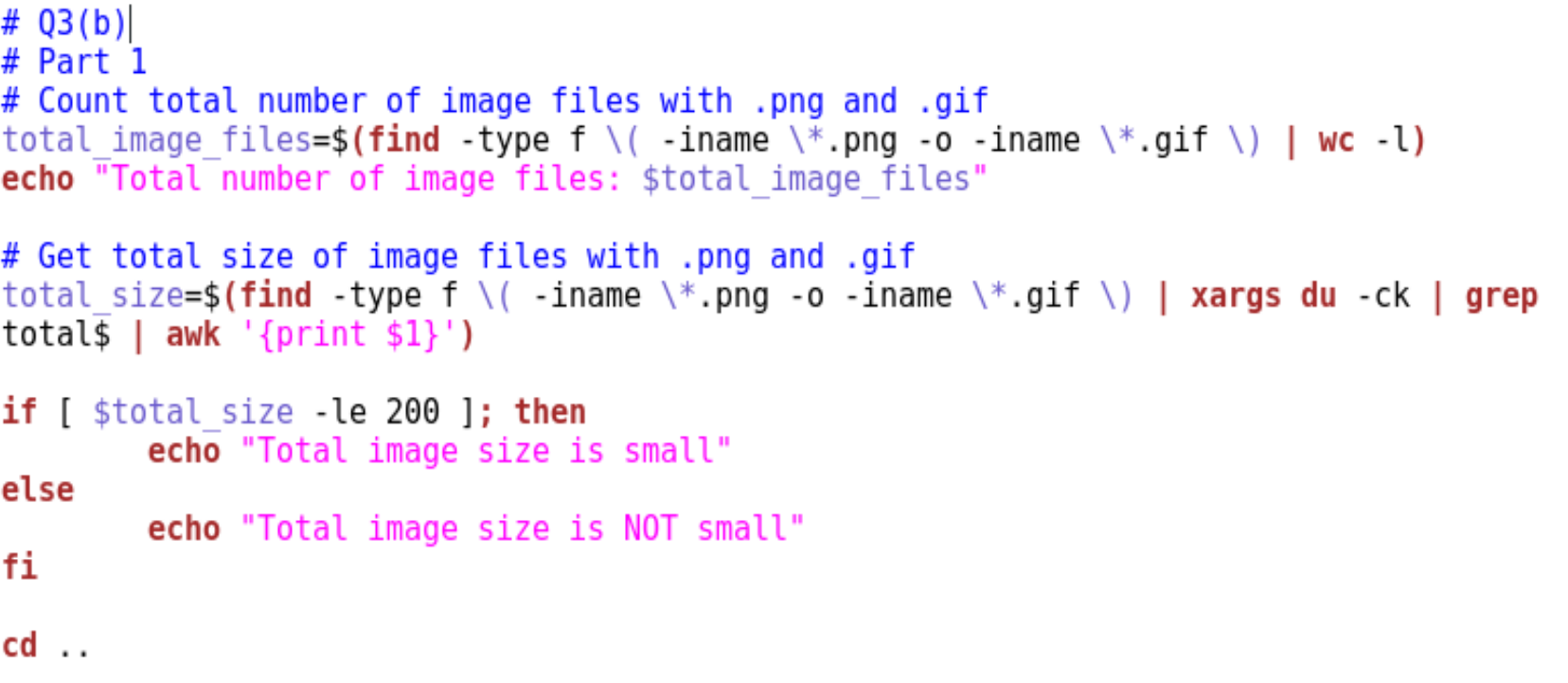
-type f tells the command find to search for type file.

-iname tells the command find that the filename is case-insensitive.

-o tells the command find to search for either .png or .gif.

Hence, regardless of how it is named, the command will find and display the file size of all image files with extension .png and .gif in the folder and its sub-folders.

An empty directory stores . and .., therefore, it has 4.0 kilobytes of file size.



It will then count the number of image files with extension .png and .gif and display it.

-ck tells the command du for a total in kilobytes.

Hence, the total file size of the image files in kilobytes will be stored in the variable total\_size.

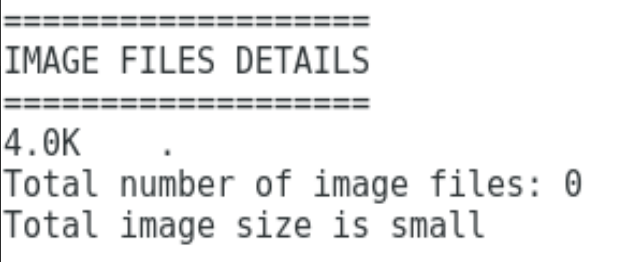
-le means less than.

The if statement is checking if total\_size is less than 200. If it is, then the total image size is small. If not, the total image size is not small.

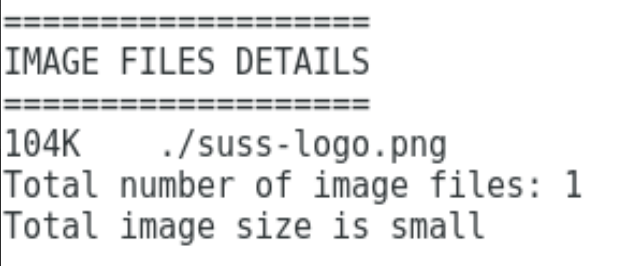
It will then cd back to the parent directory.

**Result**

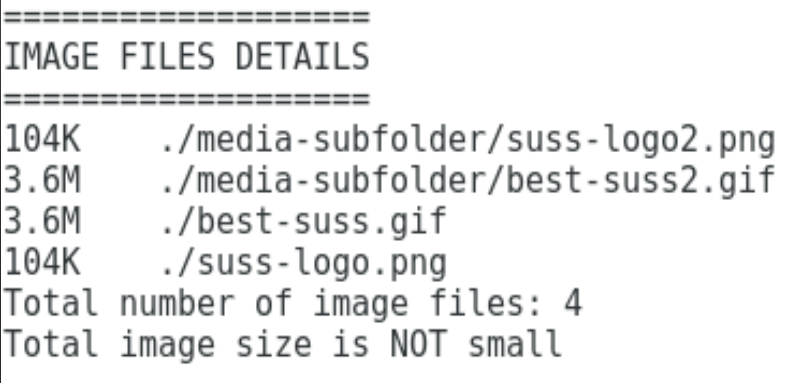
Empty folder

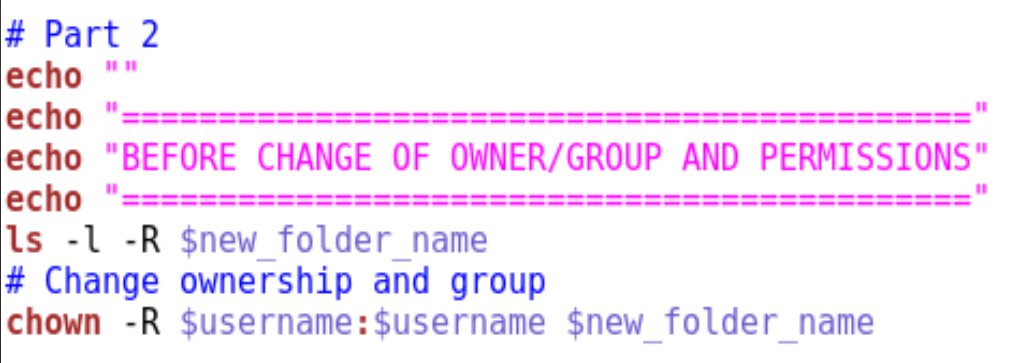


Total image size is small



Total image size is not small



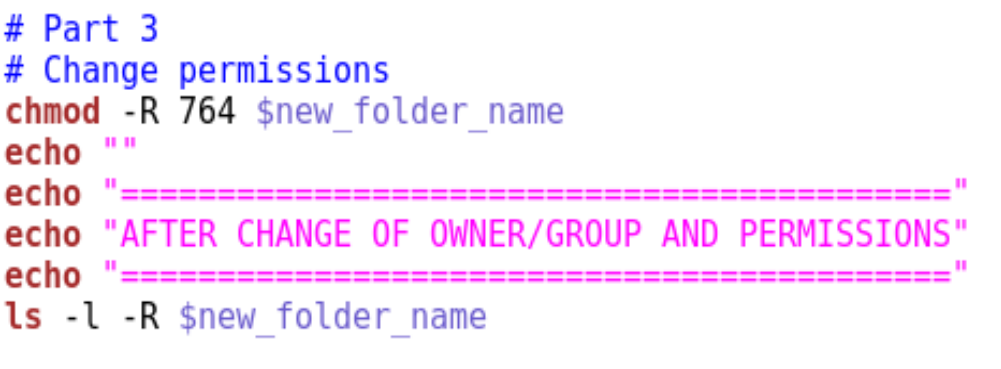


-l and -R tell the command ls to recursively display the long format for each file in the folder and its sub-folders.

The before status of the files in the folder and its sub-folders will then be displayed.

-R tells the command chown to recursively change all the files in the folder and its subfolders.

$username:$username -> the first $username refers to the owner, and the second $username refers to the group.



-R tells the command chmod to recursively change all the files in the folder and its subfolders.

4 refers to read permission, 2 refers to write permission, and 1 refers to execute permission.

764 -> the first digit refers to permission for the owner of the file, the second digit refers to permission for the group of the file, and the third digit refers to others.

Therefore, all the files will change to owner having all permissions (4+2+1=7), group having read and write permissions (4+2=6), and others having only read permission (4).

The after status of the files in the folder and its sub-folders will then be displayed.

**Result**

