

# Linux Assignment

## Task 1: File Management

Navigate to your home directory (~/).

Create a directory named `assignment_files`.

Inside `assignment_files`, create the following directory structure as in pic:

Create an empty file named `readme.txt` inside the `assignment_files` directory.

List the contents of `assignment_files` to ensure the structure is correct.

```
assignment_files
├── documents
│   ├── file1.txt
│   └── file2.txt
├── scripts
│   ├── script1.sh
│   └── script2.sh
├── logs
└── backups
```

Ans:

```
[rithvik@localhost home]$ ls -l
total 12
drwxr-xr-x.  6 root    root      65 Jul 30 23:35 assignment_files
-rwxr--rwx.  1 root    root     515 Jul 31 00:02 backup_script.sh
-rwxr--rwx.  1 root    root     830 Jul 31 00:29 cleanup_script.sh
drwx-----. 14 rithvik rithvik 4096 Jul 31 00:30 rithvik
```

```
rithvik@localhost:/home/assignmen
[rithvik@localhost assignment_files]$ ls
backups documents logs readme.txt scripts
[rithvik@localhost assignment_files]$
```

```
[rithvik@localhost scripts]$ ls -l
total 0
-rw-r--r--. 1 root root 0 Jul 31 21:05 script1.sh
-rw-r--r--. 1 root root 0 Jul 31 21:05 script2.sh
```

```
[rithvik@localhost documents]$ ls -l
total 0
-rw-r--r--. 1 root root 0 Jul 30 23:35 file1.txt
-rw-r--r--. 1 root root 0 Jul 30 23:35 file2.txt
```

```
[rithvik@localhost assignment_files]$ cat readme.txt
backups
documents
logs
scripts
```

## Task 2: Bash

Write a bash script named `backup_script.sh` that does the following:

Checks if the `backups` directory exists in `assignment_files`. If not, create it.

Copies all `.txt` files from the `documents` directory to the `backups` directory.

Appends the current date and time to each file name copied to indicate the backup timestamp (e.g., `file1.txt` becomes `file1_2024-06-25_1430.txt` where 1430 represents 2:30 PM).

Displays a message indicating the number of files backed up and their new names.

Ans:

```
[rithvik@localhost home]$ cat backup_script.sh
#!/bin/bash
if [ -d assignment_files/backups ]
then
    echo directory is Existing
else
    mkdir assignment_files/backups
fi
cp assignment_files/documents/*.txt assignment_files/backups

a=$(ls assignment_files/documents | grep .txt)
for i in ${a[@]}
do
    a=$(date +"_%Y-%m-%d_%H%M")
    file=${i%.*}${a}.txt
    mv assignment_files/backups/$i assignment_files/backups/$file
done
echo Number of files backed up : $(ls assignment_files/backups | wc -l)
for i in [ $(ls assignment_files/backups) ]
do
    echo $i
done
```

```
[rithvik@localhost home]$ sudo ./backup_script.sh
directory is Existing
Number of files backed up : 2
[
file1_2024-07-31_2118.txt
file2_2024-07-31_2118.txt
]
```

### Task 3: File Permissions

Navigate to the assignment\_files directory.

Modify the permissions of the following files and directories:

**documents directory:** Ensure it is readable, writable, and executable by the owner, and readable and executable by others.

**scripts directory:** Ensure it is readable, writable, and executable by the owner only.

**logs directory:** Ensure it is writable and executable by the owner only.

**backups directory:** Ensure it is readable and writable by the owner only.

All .sh files inside the scripts directory: Ensure they are executable by the owner only.

Ans:

```
[rithvik@localhost assignment_files]$ ls -l
total 4
drw-----. 2 root root 108 Jul 31 21:23 backups
drwx---r-x. 2 root root  40 Jul 30 23:35 documents
d-wx-----. 2 root root   6 Jul 30 23:35 logs
-rw-r--rw-. 1 root root  31 Jul 31 20:52 readme.txt
drwx-----. 2 root root   6 Jul 31 21:23 scripts
```

```
[rithvik@localhost scripts]$ sudo chmod u+x *.sh
[rithvik@localhost scripts]$ ls -l
total 0
---x-----. 1 root root 0 Jul 31 21:35 script1.sh
---x-----. 1 root root 0 Jul 31 21:35 script2.sh
```

## Task 4: File Management

Write a bash script named cleanup\_script.sh that accomplishes the following:

Deletes all .txt files from the documents directory that are older than 7 days.

Moves all .sh files from the scripts directory to the backups directory.

Displays a message indicating the number of files deleted from documents and the number of files moved to backups.

Ans:

```
[rithvik@localhost home]$ cat cleanup_script.sh
#!/bin/bash
al=$(ls -l assignment_files/documents | grep .txt)
ak=${#al[@]}
count_doc=$((0))
if [ $ak -gt 1 ]
then
    for i in assignment_files/documents/*.txt
    do
        a=$(stat -c %w $i | awk '{print $1}')
        b=$(date -d $a +%s)
        c=$(date +%s)
        d=$((($c-$b)/86400))
        if [ $d -gt 7 ]
        then
            count_doc=$((count_doc+1))
            rm $i
        fi
    done
else
    echo There is no files with .txt extention
fi

count_bak=$((0))
```

```
count_bak=$((0))
bl=$(ls -l assignment_files/scripts | grep .sh)
bk=${#bl[@]}
if [ "$bk" -gt 1 ]
then
    count_bak=$(find assignment_files/scripts -type f -name "*.sh" | wc -l)
    mv assignment_files/scripts/*.sh assignment_files/backups
else
    echo There is no files with .sh extention
fi
echo number of files deleted from the document : $count_doc
echo number of files moved to backups: $count_bak
```

```
echo number of files moved to backups: $count_bak
[rithvik@localhost home]$ sudo ./cleanup_script.sh
number of files deleted from the document : 0
number of files moved to backups: 2
```

```
[rithvik@localhost backups]$ ls -l
total 0
-rw-r--r--. 1 root root 0 Jul 31 21:18 file1_2024-07-31_2118.txt
-rw-r--r--. 1 root root 0 Jul 31 21:18 file2_2024-07-31_2118.txt
-rw-r--r--. 1 root root 0 Jul 31 21:20 script1.sh
-rw-r--r--. 1 root root 0 Jul 31 21:20 script2.sh
```