



**Module Code & Module Title**

**Level 5 – Network Operating System**

**Assessment Type**

**Logbook 7**

**Semester**

**2024 Spring**

**Student Name: Yogesh Dhakal**

**London Met ID: 23057067**

**College ID: NP04CP4S240032**

**Assignment Submission Date: 21<sup>st</sup> December 2024**

**Submitted To: Mr. Prashant Adhikari**

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

## Table of Contents

1. Introduction .....	3
2. Objective .....	3
3. Conclusion .....	12

## Tables of Figures

Figure 1: Creating directory using -p .....	4
Figure 2: Changing path .....	4
Figure 3: Moving to different director .....	4
Figure 4: Creating a file .....	5
Figure 5: Moving to different directory and creating a file .....	5
Figure 6: Moving a file .....	5
Figure 7: Using \n .....	6
Figure 8: Using ls commands .....	6
Figure 9: Removing directories.....	7
Figure 10: Displaying permission .....	7
Figure 11: Removing all the permissions.....	8
Figure 12:Displaying all removed permissions .....	8
Figure 13: Giving permission to user.....	9
Figure 14: Displaying all the accessed permission .....	9
Figure 15: Displaying access for W7-1 .....	10
Figure 16: Removing all the permissions .....	10
Figure 17: Displaying denied permissions.....	11
Figure 18: Adding permissions .....	11
Figure 19: Testing all the accessed permissions .....	12

## 1. Introduction

This exercise is intended to get you using basic UNIX utilities to establish command-line skills. Your `mkdir` command will create a specific directory structure with and without options, particularly `-p`. You'll learn how to navigate directories using relative paths (`...`) and carry out file operations like copying, moving, and renaming files. Other aspects to learn would be all about controlling access to file and directory permissions by using the `chmod` command. There would also be practicing important commands like `echo`, `ls`, `cat`, and `rm` for the management as well as output of files. It would entail efficiently navigating through directories, manipulating files as well as handling permissions in the UNIX environment. This is going to strengthen your UNIX skills hands-on!.

## 2. Objective

You're basically getting hands-on with UNIX utilities through hands-on command-line practice. By doing this task, you will have:

- i.) Be Creating directory structures systematically with command `mkdir` with or without options like `-p`.
- ii.) Confident in navigating directories using relevant paths (`..` for parent, `.` for current directories).
- iii.) Be understanding the most key file operations: creating, copying, moving and renaming files from all over the directories.
- iv.) Be Using `chmod` for understanding how to control permissions bit by bit: read, write and execute those permissions for files and directories.
- v.) Be Practicing such important commands as `ls`, `echo`, `cat`, `rm` with which you can do exploring directories and manipulating files with commands for output texts.
- vi.) Understand the file system behaviour, so that you're able to handle access restrictions, cleanup directories, and troubleshoot permissions.

```
ubuntu@ubuntu: ~  
ubuntu@ubuntu:~$ mkdir -p W7/{W7-1/{1level3,2level3},W7-2/{3level3,4level3}}  
ubuntu@ubuntu:~$ tree W7  
W7  
├── W7-1  
│   ├── 1level3  
│   └── 2level3  
└── W7-2  
    ├── 3level3  
    └── 4level3  
  
7 directories, 0 files
```

Figure 1: Creating directory using -p

So first we created a directory using -p and mkdir. -p allows us to create directory with its parent directories if it doesn't exist.

```
ubuntu@ubuntu: ~/W7/W7-2/4level3  
ubuntu@ubuntu:~$ cd W7/W7-1/1level3/  
ubuntu@ubuntu:~/W7/W7-1/1level3$ cd ../../W7-2/4level3  
ubuntu@ubuntu:~/W7/W7-2/4level3$
```

Figure 2: Changing path

Since we aren't using -p we can't create inner directory while making parent directory.

```
ubuntu@ubuntu: ~/W7/W7-2/4level3  
ubuntu@ubuntu:~$ cd W7/W7-1/1level3/  
ubuntu@ubuntu:~/W7/W7-1/1level3$ cd ../../W7-2/4level3  
ubuntu@ubuntu:~/W7/W7-2/4level3$
```

Figure 3: Moving to different director

Moving to different directories using the cd code.

```
ubuntu@ubuntu: ~/W7/W7-1/1level3
ubuntu@ubuntu:~/W7/W7-2/4level3$ cd ../../W7-1/1level3/
ubuntu@ubuntu:~/W7/W7-1/1level3$ cat>file
This is my file
^Z
[1]+  Stopped                  cat > file
ubuntu@ubuntu:~/W7/W7-1/1level3$
```

Figure 4: Creating a file

Since we are in a different directory and if we want to move to different parent directory we will have to use the symbol “..” along with cd command.

```
ubuntu@ubuntu: ~/W7/W7-1/1level3
ubuntu@ubuntu:~/W7/W7-1/1level3$ cp file file1
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls
file  file1
ubuntu@ubuntu:~/W7/W7-1/1level3$ cp file../2level3/
cp: missing destination file operand after 'file../2level3/'
Try 'cp --help' for more information.
ubuntu@ubuntu:~/W7/W7-1/1level3$ cp file ../2level3/
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls ../2level3/
file
```

Figure 5: Moving to different directory and creating a file

Then I moved on to earlier directory and created a file using the cat command with redirect > feature and then checked if the file was created or not using ls command.

```
ubuntu@ubuntu:~/W7/W7-1/1level3$ mv file ../../W7-2/4level3
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls ../../W7-2/4level3
file
ubuntu@ubuntu:~/W7/W7-1/1level3$
```

Figure 6: Moving a file

Later we moved the file named “file” to a different parent directory using mv command and also checked if it was moved properly and then we checked if the moved file still exists in the current directory (it doesn't).

```
ubuntu@ubuntu: ~/W7/W7-1/1level3
ubuntu@ubuntu:~/W7/W7-1/1level3$ echo -e "Hello! I can do it \n 5 > (20 : 8) <
( 30 * 2) \n Line1 \n Line 2 \n a-b, A-B, -, +, <, >, #, $, %, &. "
Hello! I can do it
 5 > (20 : 8) < ( 30 * 2)
Line1
Line 2
a-b, A-B, -, +, <, >, #, $, %, &.
ubuntu@ubuntu:~/W7/W7-1/1level3$
```

Figure 7: Using \n

We printed few words using the echo -e command where -e allows for escape sequences like \n, \t, \\\, \b, etc.

```
ubuntu@ubuntu: ~/W7/W7-1/1level3
ubuntu@ubuntu:~/W7/W7-1/1level3$ man ls
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls -a
.  ..  file1
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls -d
.
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls -g
total 4
-rw-rw-r-- 1 ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls -l
total 4
-rw-rw-r-- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1/1level3$ ls -R
.:
file1
```

Figure 8: Using ls commands

Using the ls -a command to check all files in the directories which also includes hidden file. ls -d \* lists all the folders inside a certain directory. I used ls -g command in Linux which is very similar to ls -l but hides the owner's information from long listing format. Here in ls -l it shows both owner's information along with group's information. Then we used ls -r to get the list in descending alphabetical order, but since there is only 1 folder in this directory, the change isn't very visible.

```
ubuntu@ubuntu: ~/W7
ubuntu@ubuntu:~/W7$ rm -ri W7-2
rm: descend into directory 'W7-2'? y
rm: descend into directory 'W7-2/4level3'? y
rm: remove regular file 'W7-2/4level3/file'? y
rm: remove directory 'W7-2/4level3'? y
rm: remove directory 'W7-2/3level3'? y
rm: remove directory 'W7-2'? y
ubuntu@ubuntu:~/W7$
```

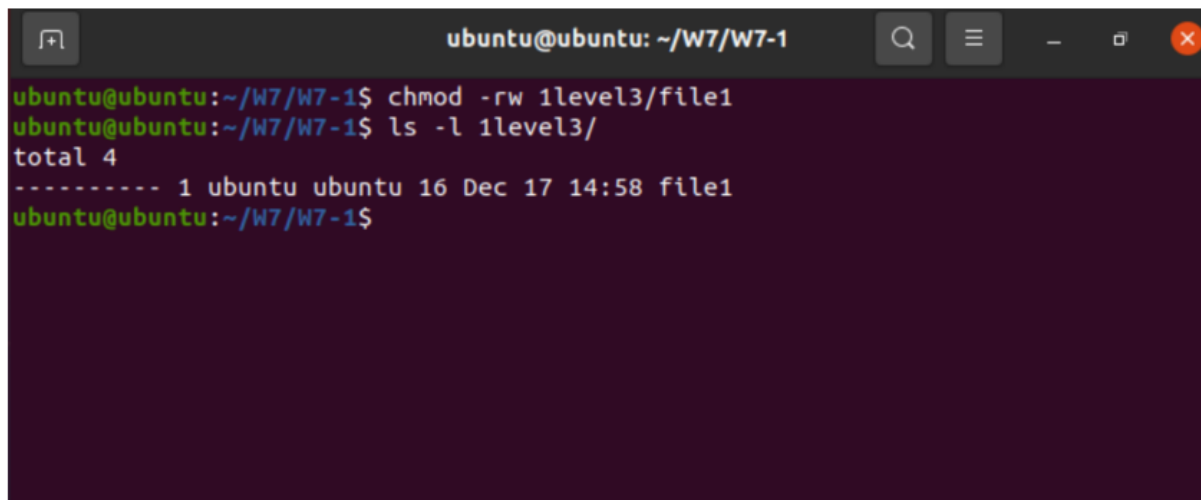
Figure 9: Removing directories

I used the `rm -ri` to safe delete the folders in which `-r` removes the directories and content recursively and `-i` asks user for confirmation before deleting each file or folder.

```
ubuntu@ubuntu: ~/W7/W7-1
ubuntu@ubuntu:~/W7/W7-1$ ls -l 1level3/
total 4
-rw-rw-r-- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 10: Displaying permission

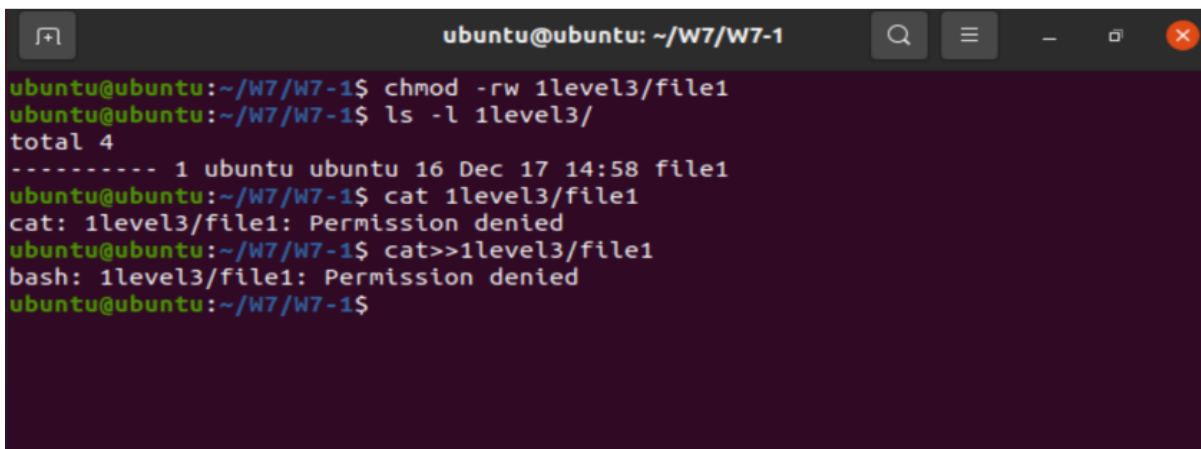
I used the `ls -l` to check the permission of files and it shows that user has read and write permission, same with the group members but others only has the permission to read the file.

A terminal window titled 'ubuntu@ubuntu: ~/W7/W7-1' with search, menu, and window control icons. The terminal shows the following commands and output:

```
ubuntu@ubuntu:~/W7/W7-1$ chmod -rw 1level3/file1
ubuntu@ubuntu:~/W7/W7-1$ ls -l 1level3/
total 4
----- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 11: Removing all the permissions.

To remove the file permission, we used the chmod with 000 to remove all the permissions from all the users of file 1.

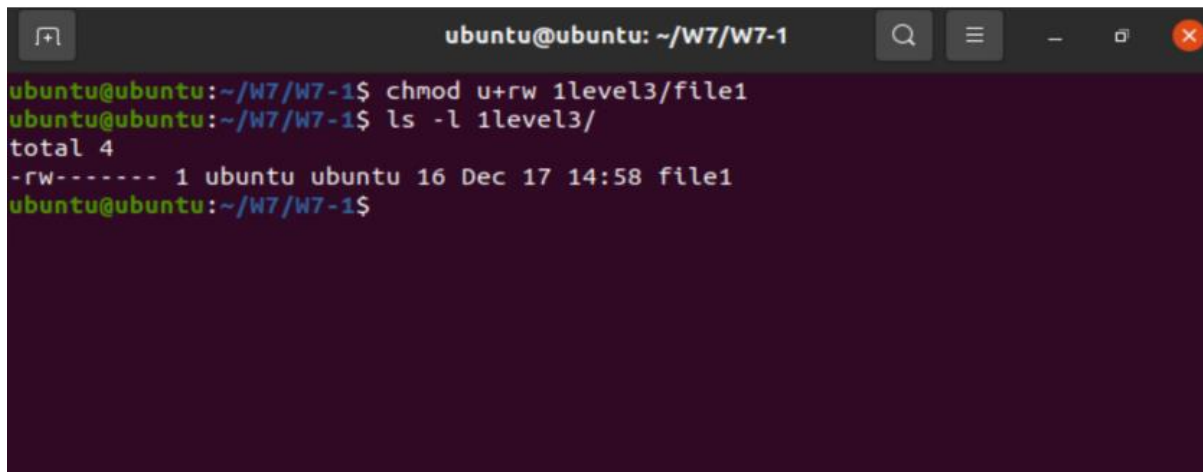
A terminal window titled 'ubuntu@ubuntu: ~/W7/W7-1' with search, menu, and window control icons. The terminal shows the following commands and output:

```
ubuntu@ubuntu:~/W7/W7-1$ chmod -rw 1level3/file1
ubuntu@ubuntu:~/W7/W7-1$ ls -l 1level3/
total 4
----- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1$ cat 1level3/file1
cat: 1level3/file1: Permission denied
ubuntu@ubuntu:~/W7/W7-1$ cat>>1level3/file1
bash: 1level3/file1: Permission denied
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 12: Displaying all removed permissions

Then we tried to read the file permission using cat command but we couldn't since we removed everyone's every permission from that certain file. Then we tried to write on the same permission less file but obviously the system didn't let us manipulate it.

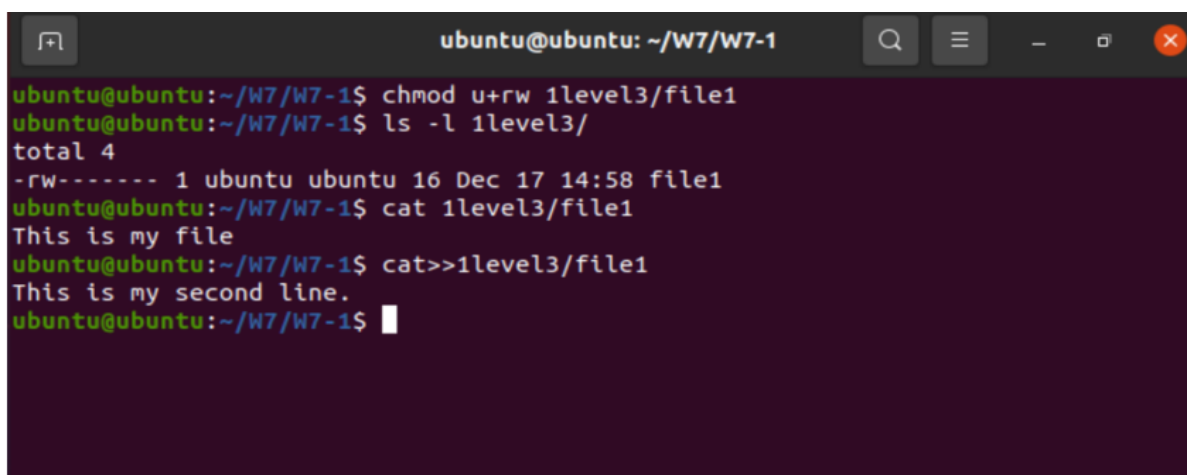


A terminal window with a dark purple background. The title bar shows 'ubuntu@ubuntu: ~/W7/W7-1'. The terminal text is as follows:

```
ubuntu@ubuntu:~/W7/W7-1$ chmod u+rw 1level3/file1
ubuntu@ubuntu:~/W7/W7-1$ ls -l 1level3/
total 4
-rw----- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 13: Giving permission to user

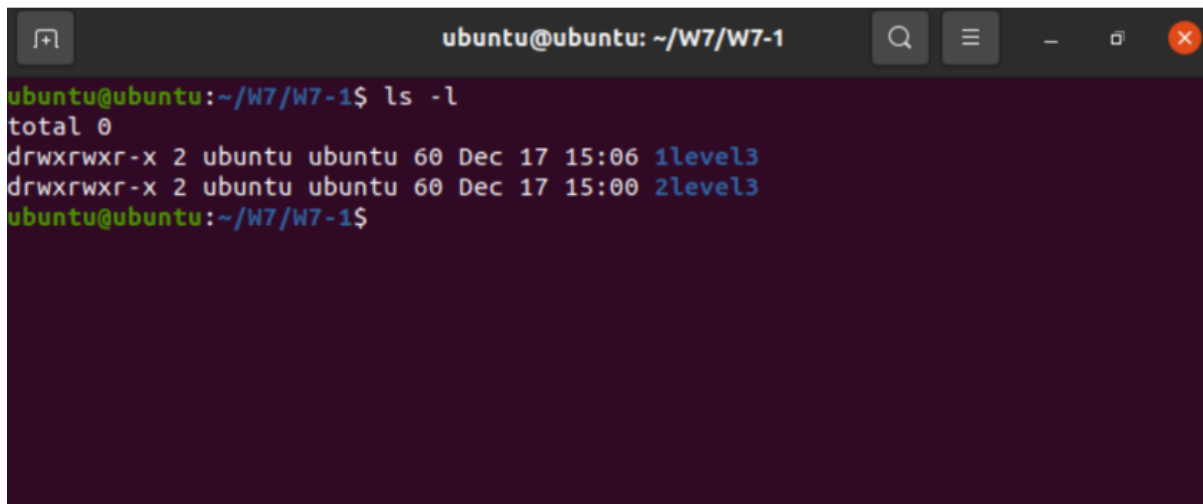
Then I added reading and writing permission to the owner only by giving the 600 which in summary reading is 4 writing is 2 and executing is 1 and here we did 4+2 just for user.

A terminal window with a dark purple background. The title bar shows 'ubuntu@ubuntu: ~/W7/W7-1'. The terminal text is as follows:

```
ubuntu@ubuntu:~/W7/W7-1$ chmod u+rw 1level3/file1
ubuntu@ubuntu:~/W7/W7-1$ ls -l 1level3/
total 4
-rw----- 1 ubuntu ubuntu 16 Dec 17 14:58 file1
ubuntu@ubuntu:~/W7/W7-1$ cat 1level3/file1
This is my file
ubuntu@ubuntu:~/W7/W7-1$ cat>>1level3/file1
This is my second line.
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 14: Displaying all the accessed permission

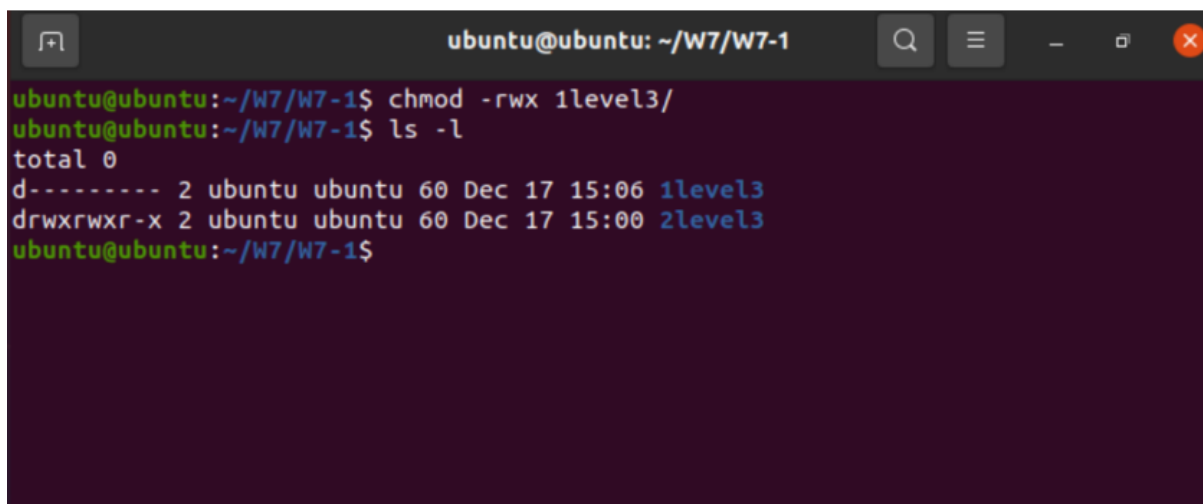
After I gave permission to the file, I tried reading it with cat command which was successful. After that I tried redirecting a new line in the file1 using echo and >> which was successful and I read the file using the utility called cat.

A terminal window titled 'ubuntu@ubuntu: ~/W7/W7-1' with search, menu, and window control icons in the title bar. The terminal shows the command 'ls -l' being executed. The output displays the permissions for the current directory and two subdirectories, '1level3' and '2level3'. Both subdirectories have permissions 'drwxrwxr-x' and are owned by 'ubuntu' with a group of 'ubuntu' and a size of '60'. The timestamps are 'Dec 17 15:06' for '1level3' and 'Dec 17 15:00' for '2level3'.

```
ubuntu@ubuntu: ~/W7/W7-1
ubuntu@ubuntu:~/W7/W7-1$ ls -l
total 0
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:06 1level3
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:00 2level3
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 15: Displaying access for W7-1

By using `ls -l` i checked the permission of both of the directory of current directory.

A terminal window titled 'ubuntu@ubuntu: ~/W7/W7-1' with search, menu, and window control icons in the title bar. The terminal shows the command 'chmod -rwx 1level3/' being executed, followed by 'ls -l'. The output shows that the permissions for '1level3' have been removed, indicated by 'd-----' instead of 'drwxrwxr-x'. The other details for '1level3' (owner 'ubuntu', group 'ubuntu', size '60', timestamp 'Dec 17 15:06') and for '2level3' (owner 'ubuntu', group 'ubuntu', size '60', timestamp 'Dec 17 15:00') remain the same.

```
ubuntu@ubuntu:~/W7/W7-1$ chmod -rwx 1level3/
ubuntu@ubuntu:~/W7/W7-1$ ls -l
total 0
d----- 2 ubuntu ubuntu 60 Dec 17 15:06 1level3
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:00 2level3
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 16: Removing all the permissions

I used `chmod -rwx` again to remove all the permissions from the directory. And checked it by using the `ls -l` command.

```
ubuntu@ubuntu: ~/W7/W7-1
ubuntu@ubuntu:~/W7/W7-1$ chmod -rwx 1level3/
ubuntu@ubuntu:~/W7/W7-1$ ls -l
total 0
d----- 2 ubuntu ubuntu 60 Dec 17 15:06 1level3
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:00 2level3
ubuntu@ubuntu:~/W7/W7-1$ cat 1level3/file1
cat: 1level3/file1: Permission denied
ubuntu@ubuntu:~/W7/W7-1$ cat>>1level3/file1
bash: 1level3/file1: Permission denied
ubuntu@ubuntu:~/W7/W7-1$ ls 1level3/
ls: cannot open directory '1level3/': Permission denied
ubuntu@ubuntu:~/W7/W7-1$
```

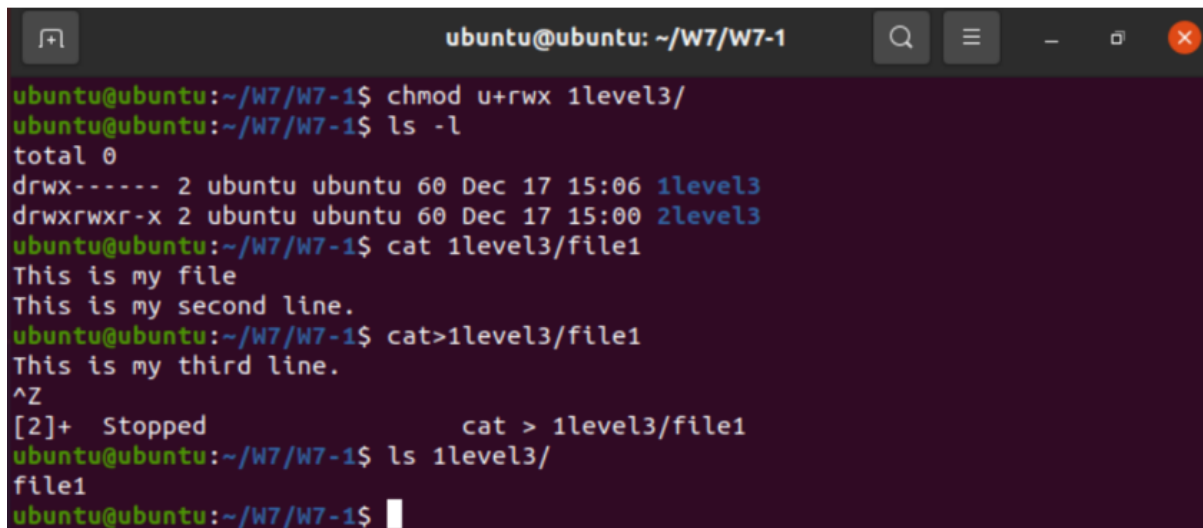
Figure 17: Displaying denied permissions

Then I tried to read the file of the same directory where I removed all the permission from but the permission was denied. Then again I tried writing in the same directory by using cat >> which would let us write inside the file of 1level3 directory.

```
ubuntu@ubuntu: ~/W7/W7-1
ubuntu@ubuntu:~/W7/W7-1$ chmod u+rwx 1level3/
ubuntu@ubuntu:~/W7/W7-1$ ls -l
total 0
drwx----- 2 ubuntu ubuntu 60 Dec 17 15:06 1level3
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:00 2level3
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 18: Adding permissions

Then I used chmod u+rwx to give permission of the folder to read write and execute to myself.

A terminal window titled 'ubuntu@ubuntu: ~/W7/W7-1' with search, menu, and window control icons. The terminal shows the following commands and output:

```
ubuntu@ubuntu:~/W7/W7-1$ chmod u+rwX 1level3/
ubuntu@ubuntu:~/W7/W7-1$ ls -l
total 0
drwx----- 2 ubuntu ubuntu 60 Dec 17 15:06 1level3
drwxrwxr-x 2 ubuntu ubuntu 60 Dec 17 15:00 2level3
ubuntu@ubuntu:~/W7/W7-1$ cat 1level3/file1
This is my file
This is my second line.
ubuntu@ubuntu:~/W7/W7-1$ cat>1level3/file1
This is my third line.
^Z
[2]+  Stopped                  cat > 1level3/file1
ubuntu@ubuntu:~/W7/W7-1$ ls 1level3/
file1
ubuntu@ubuntu:~/W7/W7-1$
```

Figure 19: Testing all the accessed permissions

Then I checked if I changed the permission of the directory properly or not using the `ls -l` command. After I gave the permission to user, I tried writing inside it using the `cat` command and redirect `>>` symbol. And we didn't get denied this time. And at last searching file in the directory.

### 3. Conclusion

You practiced essential UNIX commands through exercise by doing the following: Creating a directory, traversing it, manipulating files, and managing permissions, which further enlightened you on how the file system works. It also used commands like `mkdir`, `ls`, `chmod`, and `rm`-all-important commands for efficient file and directory management. The load was lightened by practice, and theory was enabled preparing you for real-world UNIX skills-needed system administration and development tasks. Continued practice is essential for mastery of these tools and for independence in managing UNIX environments.