

Lab Report 1



Course Title: Operating Systems Lab

Course Code: CSE324

Submitted by

Wasikur Chowdhury Wasif

Student ID: 0242220005101955

Section: 63_M1

Semester: Spring-25

Department: Computer Science and Engineering

Submitted to

Nushrat Jahan Oyshi

Lecturer

Computer Science & Engineering

Daffodil International University

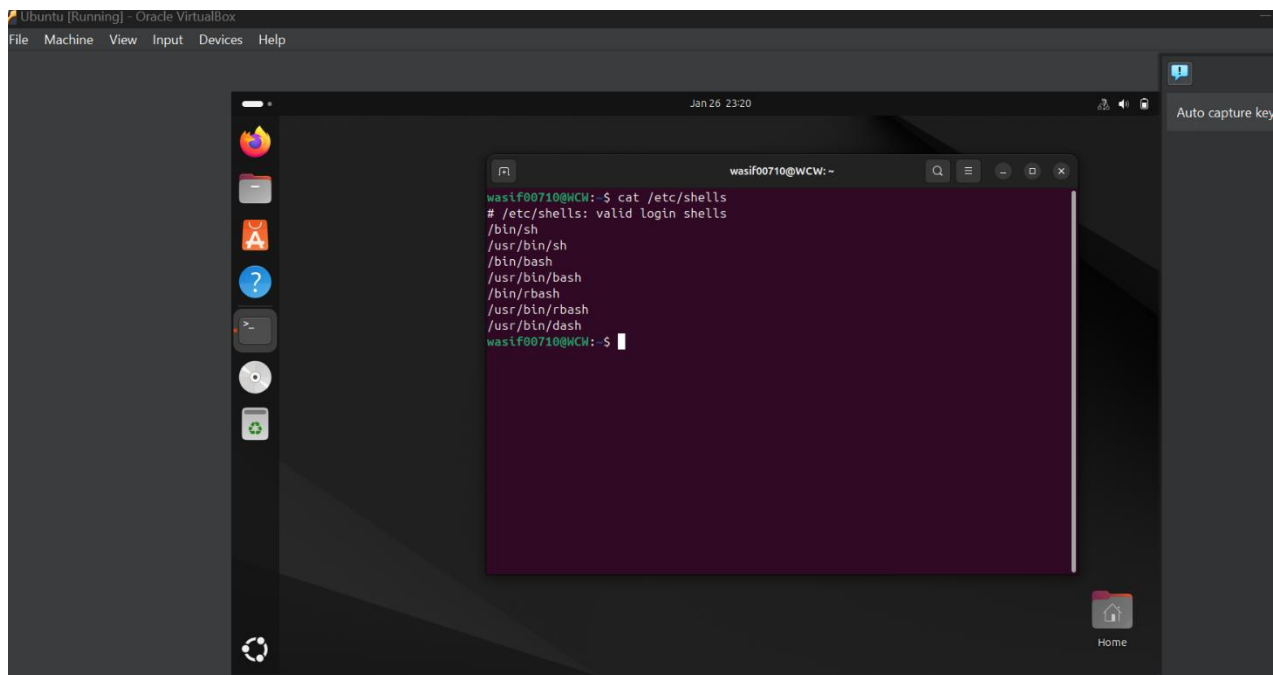
Submission Date: 27 January, 2025

Introduction

This report explores various fundamental Linux commands and their applications in managing files, directories, and scripts. The objective is to demonstrate how to navigate the Linux filesystem, modify file permissions, and create and execute Bash scripts effectively. Through practical examples, such as viewing system files like **/etc/shells**, creating folders and files using **mkdir** and **touch**, and making scripts executable with **chmod**, this report highlights the flexibility and power of Linux as an operating system.

The tasks performed include creating and organizing directories and files, managing file permissions, and executing a Bash script that incorporates user inputs and environmental variables. These foundational skills are crucial for anyone looking to work with the Linux operating system effectively.

Working with Root folders

A screenshot of a terminal window running on an Ubuntu virtual machine. The terminal shows the command 'cat /etc/shells' being executed, which displays the valid login shells: /bin/sh, /usr/bin/sh, /bin/bash, /usr/bin/bash, /bin/rbash, /usr/bin/rbash, and /usr/bin/dash. The terminal window is titled 'wasif00710@WCW: ~' and is set against a dark purple background. The virtual machine interface includes a menu bar at the top with 'File', 'Machine', 'View', 'Input', 'Devices', and 'Help'. A sidebar on the left shows various application icons, and a 'Home' button is visible in the bottom right corner of the terminal window.

```
wasif00710@WCW:~  
wasif00710@WCW:~$ cat /etc/shells  
# /etc/shells: valid login shells  
/bin/sh  
/usr/bin/sh  
/bin/bash  
/usr/bin/bash  
/bin/rbash  
/usr/bin/rbash  
/usr/bin/dash  
wasif00710@WCW:~$
```

In Linux, the root directory (/) serves as the starting point for the entire filesystem. All other directories and files are organized under this root directory. The /etc directory, a critical subdirectory within the root, contains configuration files essential for system operations.

One example of accessing such files is viewing the contents of the /etc/shells file using the command:

```
$ cat /etc/shells
```

This command retrieves and displays the list of valid login shells available on the system. While the /etc directory is part of the root-level structure, this specific command does not require administrative privileges because /etc/shells is readable by all users. However, modifying files in the /etc directory typically requires superuser (root) privileges.

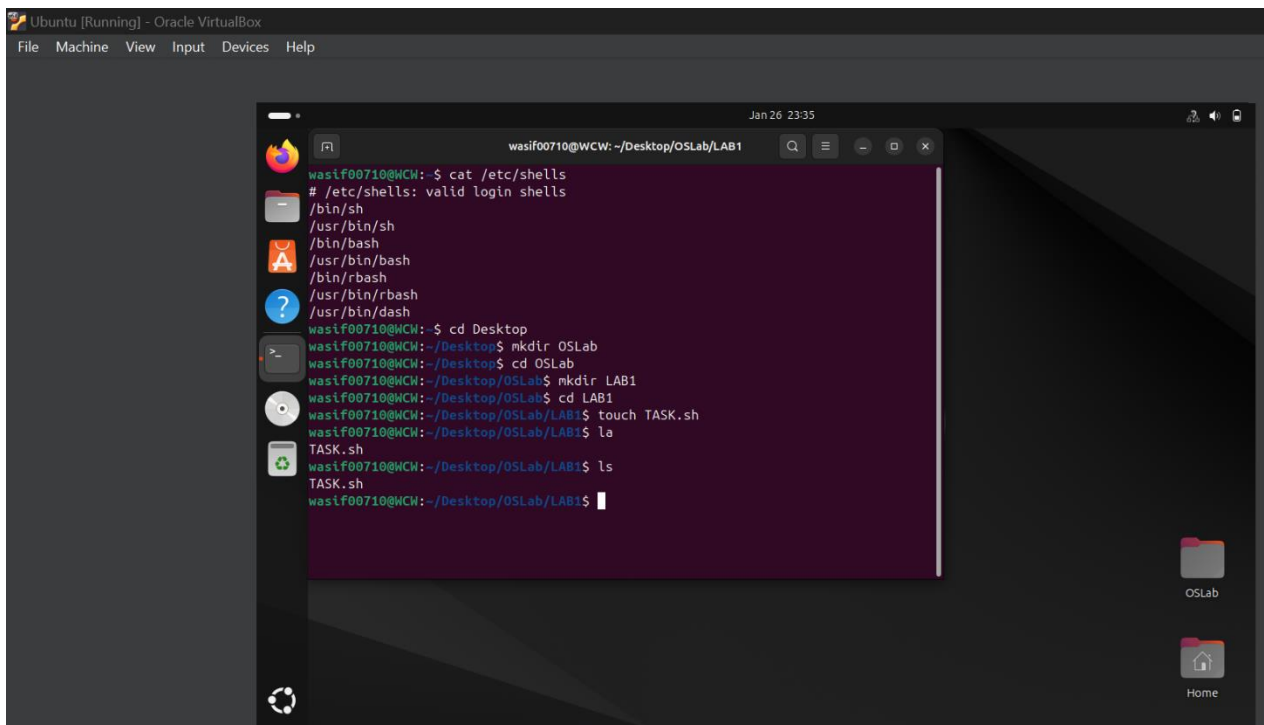
Creating Directories and files

In Linux, directories and files can be easily created and organized using terminal commands. Here is an example demonstrating this process:

Creating a Folder:

I used the mkdir command to create a folder named OSLab with the following command:

```
$ mkdir OSLab
```



The screenshot shows a terminal window titled 'wasif00710@WCW: ~/Desktop/OSLab/LAB1' with a search bar and window controls. The terminal output is as follows:

```
wasif00710@WCW:~$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/usr/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/usr/bin/dash
wasif00710@WCW:~$ cd Desktop
wasif00710@WCW:~/Desktop$ mkdir OSLab
wasif00710@WCW:~/Desktop$ cd OSLab
wasif00710@WCW:~/Desktop/OSLab$ mkdir LAB1
wasif00710@WCW:~/Desktop/OSLab$ cd LAB1
wasif00710@WCW:~/Desktop/OSLab/LAB1$ touch TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls
TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$
```

The desktop environment is visible in the background, showing a folder icon labeled 'OSLab' and a 'Home' icon.

Navigating into the Folder:

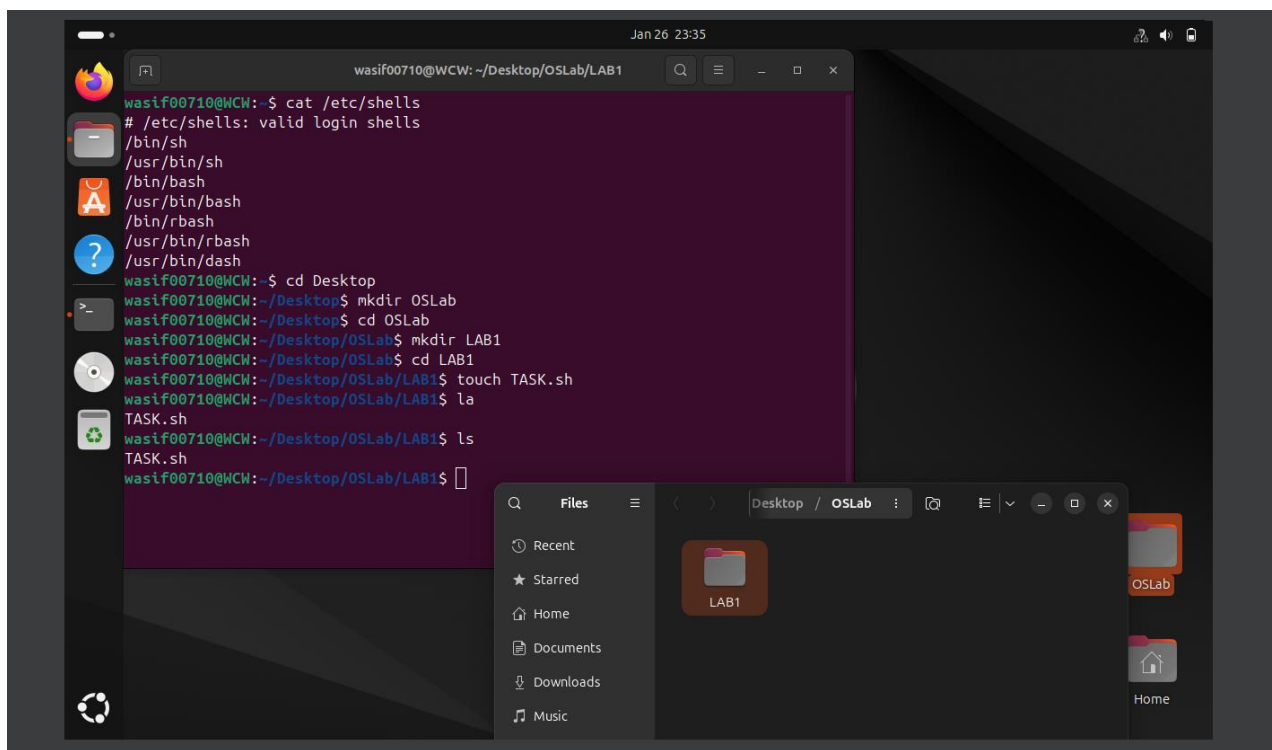
To work inside the newly created folder, I used the cd command:

```
$ cd OSLab
```

Creating a Subfolder:

Inside OSLab, I created another folder named LAB1 using:

```
$ mkdir LAB1
```



Navigating into the Subfolder:

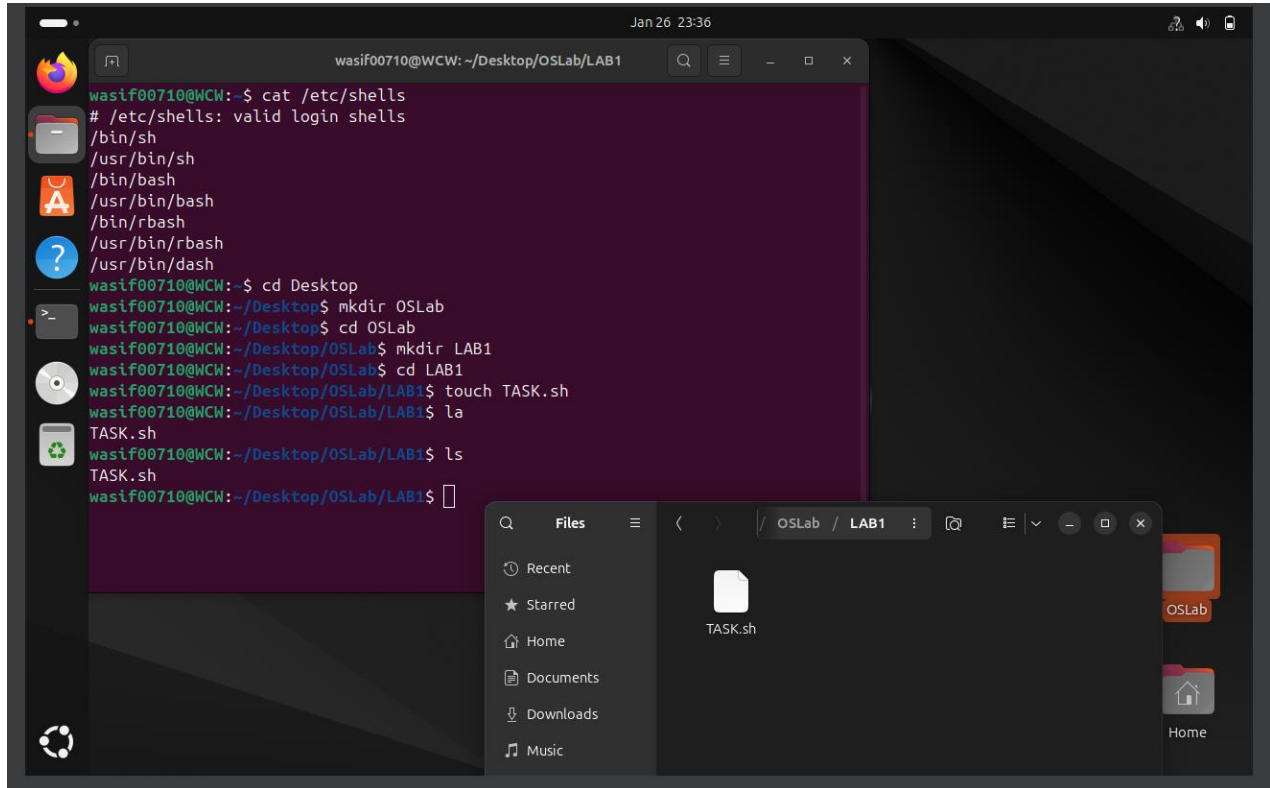
To access the Lab1 folder, I navigated into it using:

```
$ cd Lab1
```

Creating a File:

Within the Lab1 folder, I created a file named TASK.sh using the touch command:

\$ touch TASK.sh



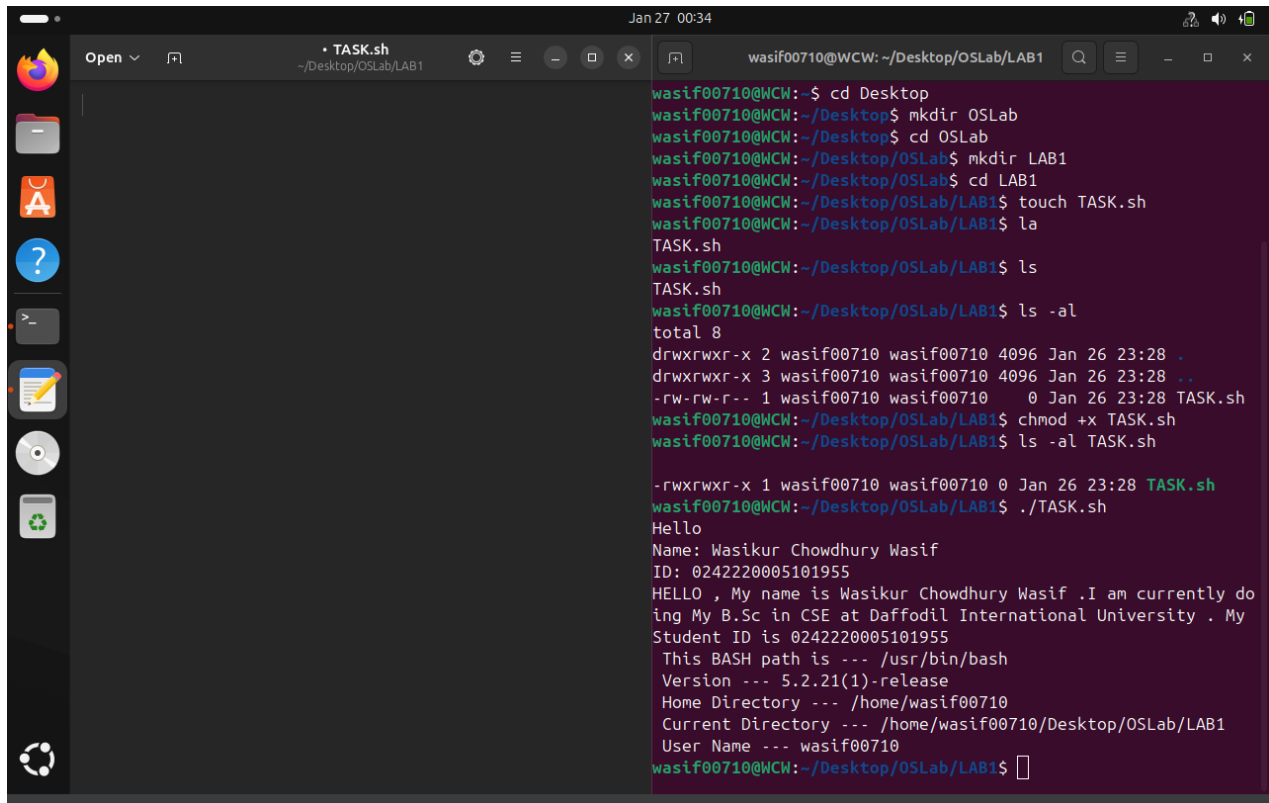
Viewing the Folder Contents:

To confirm the creation of the file, I used the ls command to list the contents of the folder:

\$ ls

The output showed the file TASK.sh in the folder.

File Permissions

A screenshot of a Linux terminal window. The window title is "TASK.sh" and the path is "~/Desktop/OSLab/LAB1". The terminal shows a series of commands and their outputs. The user navigates to the Desktop, creates a directory OSLab, then LAB1, and creates a file TASK.sh. They then use 'ls -al' to view file permissions, which are initially -rw-rw-r--. The user then runs 'chmod +x TASK.sh' to make the file executable. A second 'ls -al' shows the permissions as -rwxrwxr-x. Finally, the user runs './TASK.sh', which outputs a greeting and system information.

```
wasif00710@WCW:~$ cd Desktop
wasif00710@WCW:~/Desktop$ mkdir OSLab
wasif00710@WCW:~/Desktop$ cd OSLab
wasif00710@WCW:~/Desktop/OSLab$ mkdir LAB1
wasif00710@WCW:~/Desktop/OSLab$ cd LAB1
wasif00710@WCW:~/Desktop/OSLab/LAB1$ touch TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls
TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls -al
total 8
drwxrwxr-x 2 wasif00710 wasif00710 4096 Jan 26 23:28 .
drwxrwxr-x 3 wasif00710 wasif00710 4096 Jan 26 23:28 ..
-rw-rw-r-- 1 wasif00710 wasif00710 0 Jan 26 23:28 TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ chmod +x TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls -al TASK.sh
-rwxrwxr-x 1 wasif00710 wasif00710 0 Jan 26 23:28 TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ./TASK.sh
Hello
Name: Wasikur Chowdhury Wasif
ID: 0242220005101955
HELLO , My name is Wasikur Chowdhury Wasif .I am currently doing My B.Sc in CSE at Daffodil International University . My Student ID is 0242220005101955
This BASH path is --- /usr/bin/bash
Version --- 5.2.21(1)-release
Home Directory --- /home/wasif00710
Current Directory --- /home/wasif00710/Desktop/OSLab/LAB1
User Name --- wasif00710
wasif00710@WCW:~/Desktop/OSLab/LAB1$
```

In Linux, file permissions determine who can read, write, or execute a file. These permissions can be viewed and modified using specific commands. Below is an example of working with file permissions:

Viewing File Permissions:

I used the **ls -al** command to list all files and directories in the current folder along with their permissions:

```
$ ls -al
```

This command displayed detailed information, including the file name, ownership, and permissions. For the file **TASK.sh**, the permissions were initially displayed as **-rw-r--r--**, indicating it was readable and writable by the owner but not executable.

Changing Permissions:

To make the file TASK.sh executable, I used the chmod command with the +x flag:

```
$ chmod +x TASK.sh
```

This command added execute permissions to the file, allowing it to be run as a script.

Verifying the Updated Permissions:

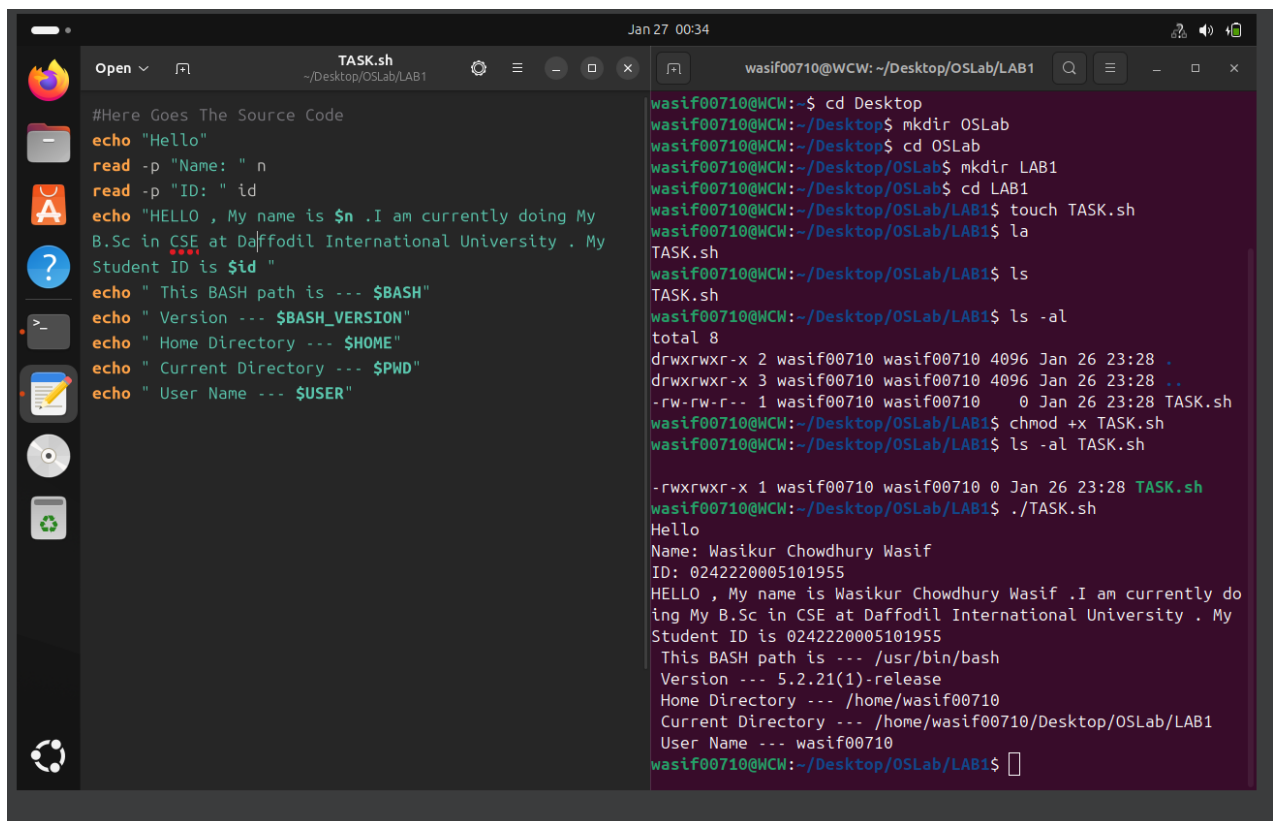
After changing the permissions, I used the following command to check the updated permissions of TASK.sh:

```
$ ls -al TASK.sh
```

The output showed the new permissions as **-rwxr--r--**, indicating that the file is now executable by the owner while retaining read permissions for others.

Running Bash script using TASK.sh and use of variables

Bash scripts allow automation and execution of commands sequentially. Below is an example of creating and running a Bash script:



```
Jan 27 00:34
TASK.sh
~/Desktop/OSLab/LAB1
wasif00710@WCW: ~/Desktop/OSLab/LAB1

#Here Goes The Source Code
echo "Hello"
read -p "Name: " n
read -p "ID: " id
echo "HELLO , My name is $n .I am currently doing My
B.Sc in CSE at Daffodil International University . My
Student ID is $id "
echo " This BASH path is --- $BASH"
echo " Version --- $BASH_VERSION"
echo " Home Directory --- $HOME"
echo " Current Directory --- $PWD"
echo " User Name --- $USER"

wasif00710@WCW:~$ cd Desktop
wasif00710@WCW:~/Desktop$ mkdir OSLab
wasif00710@WCW:~/Desktop$ cd OSLab
wasif00710@WCW:~/Desktop/OSLab$ mkdir LAB1
wasif00710@WCW:~/Desktop/OSLab$ cd LAB1
wasif00710@WCW:~/Desktop/OSLab/LAB1$ touch TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ la
TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls
TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls -al
total 8
drwxrwxr-x 2 wasif00710 wasif00710 4096 Jan 26 23:28 .
drwxrwxr-x 3 wasif00710 wasif00710 4096 Jan 26 23:28 ..
-rw-rw-r-- 1 wasif00710 wasif00710  0 Jan 26 23:28 TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ chmod +x TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ls -al TASK.sh
-rwxrwxr-x 1 wasif00710 wasif00710 0 Jan 26 23:28 TASK.sh
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ./TASK.sh
Hello
Name: Wasikur Chowdhury Wasif
ID: 0242220005101955
HELLO , My name is Wasikur Chowdhury Wasif .I am currently do
ing My B.Sc in CSE at Daffodil International University . My
Student ID is 0242220005101955
 This BASH path is --- /usr/bin/bash
Version --- 5.2.21(1)-release
Home Directory --- /home/wasif00710
Current Directory --- /home/wasif00710/Desktop/OSLab/LAB1
User Name --- wasif00710
wasif00710@WCW:~/Desktop/OSLab/LAB1$
```

Writing the Script:

In the TASK.sh file, I wrote the following code:

```
#Here Goes The Source Code
echo "Hello"
read -p "Name: " n
read -p "ID: " id
echo "HELLO , My name is $n .I am currently doing My
B.Sc in CSE at Daffodil International University . My
Student ID is $id "
echo " This BASH path is --- $BASH"
echo " Version --- $BASH_VERSION"
echo " Home Directory --- $HOME"
echo " Current Directory --- $PWD"
echo " User Name --- $USER"
```

Running the Script:

To execute the script, I used the command:

```
$ ./TASK.sh
```

Providing Input:

When the script prompted for inputs, I entered the following:

```
$ Name: Wasikur Chowdhury Wasif
```

```
$ ID: 0242220005101955
```


Output: After providing the inputs, the script displayed the following

Output:

```
wasif00710@WCW:~/Desktop/OSLab/LAB1$ ./TASK.sh
Hello
Name: Wasikur Chowdhury Wasif
ID: 0242220005101955
HELLO , My name is Wasikur Chowdhury Wasif .I am currently doing My B.Sc in CSE at Daffodil International University . My Student ID is 0242220005101955
This BASH path is --- /usr/bin/bash
Version --- 5.2.21(1)-release
Home Directory --- /home/wasif00710
Current Directory --- /home/wasif00710/Desktop/OSLab/LAB1
User Name --- wasif00710
wasif00710@WCW:~/Desktop/OSLab/LAB1$
```

Conclusion

This report explored fundamental Linux commands, including creating directories and files with **mkdir**, **cd**, and **touch**, managing file permissions with **chmod**, and automating tasks with Bash scripts. These exercises demonstrated the importance of navigating the Linux filesystem, controlling access through permissions, and leveraging scripts for efficient task execution. The skills gained are foundational for advancing in system management and scripting, essential for academic and professional pursuits in computer science.

Discussion

The report emphasized Linux's flexibility and power, from managing directories and permissions to automating tasks with Bash scripts. Understanding the hierarchical filesystem and file permissions is crucial for maintaining security and efficiency in multi-user environments. Writing interactive Bash scripts highlighted the practical benefits of automation in streamlining workflows. These tasks provided valuable hands-on experience, forming a strong base for tackling advanced Linux operations in real-world scenarios.