**OPERATING SYSTEMS LAB-01**

**MUHAMMAD SAIM NAVEED**

**24F-3043**

**TASK 1**

**User Manual**

**Installation of Xubuntu Linux on Oracle VirtualBox**

**1. Introduction**

Linux is an open-source operating system used widely in academia and industry. Xubuntu is an official lightweight flavor of Ubuntu that uses the XFCE desktop environment. It is designed to perform efficiently on systems with limited hardware resources while maintaining stability and security.

This manual explains the complete procedure for installing **Xubuntu Linux** on **Oracle VirtualBox**, along with the installation of required compilers. The setup enables safe experimentation with Linux without modifying the host operating system.

**2. Why I Chose Xubuntu and Oracle VirtualBox**

I chose Xubuntu instead of Ubuntu because Xubuntu is a lightweight Linux distribution that uses fewer system resources. It runs faster and more smoothly on systems with limited RAM and processing power. Despite being lightweight, Xubuntu is based on Ubuntu and uses the same repositories, security updates, and package management system. This makes it ideal for academic lab work without compromising stability.

I chose Oracle VirtualBox instead of VMware Workstation 17 Pro because VirtualBox is completely free and open-source, making it suitable for student and educational use. VMware Workstation Pro requires a paid license, which is not always accessible for students. VirtualBox provides all the essential virtualization features needed for operating system labs, such as virtual machine creation, snapshots, and hardware configuration, making it a reliable and cost-effective choice.

**Conclusion:** Xubuntu is an official Ubuntu distribution that uses the XFCE desktop environment instead of GNOME, making it more lightweight and resource-efficient.

**3. Why Oracle VirtualBox Instead of VMware Workstation 17 Pro**

Oracle VirtualBox was selected over VMware Workstation 17 Pro for the following reasons:

1. **Free and Open-Source**  
   Oracle VirtualBox is completely free for educational use, whereas VMware Workstation Pro requires a paid license.
2. **No License Restrictions**  
   VirtualBox can be installed and used without activation or license expiry issues.
3. **Easy Installation**  
   VirtualBox is simple to install and configure for beginners.
4. **Better for Academic Use**  
   It is widely recommended in university labs due to its accessibility.
5. **Sufficient Features**  
   VirtualBox provides all necessary virtualization features such as snapshots, shared clipboard, USB support, and networking.

**4. System Requirements**

**Hardware Requirements**

* Minimum 4 GB RAM (8 GB recommended)
* Minimum 20 GB free disk space
* Virtualization enabled in BIOS

**Software Requirements**

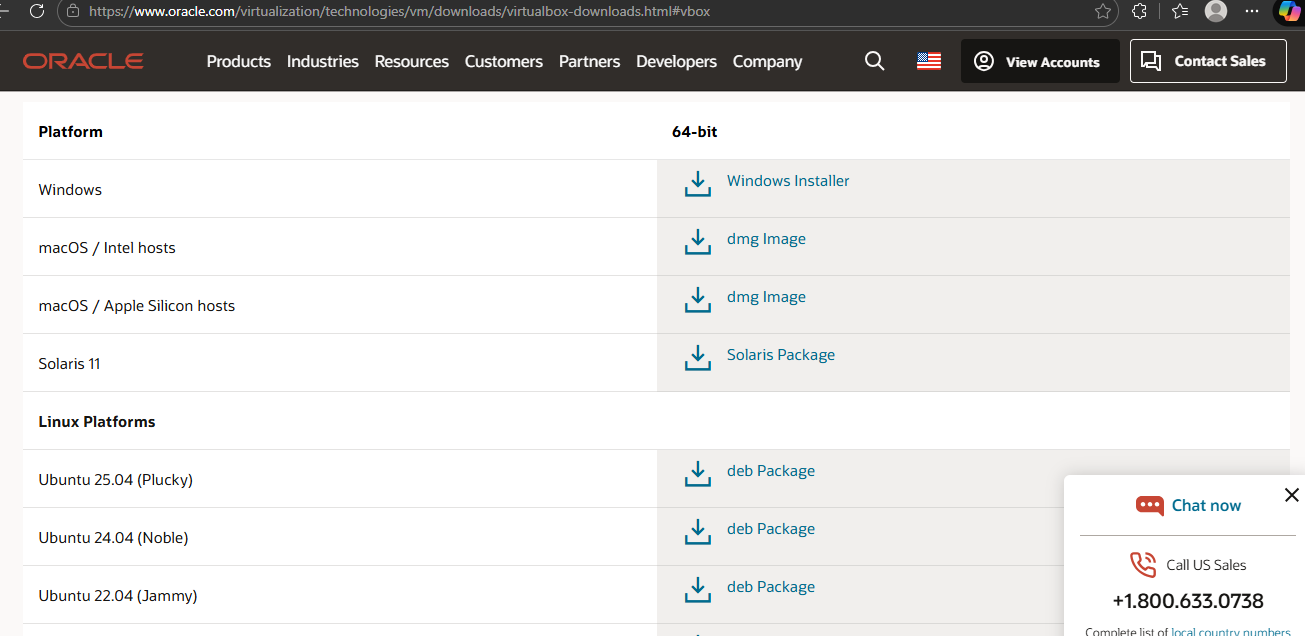
* Windows 10/11 (Host OS)
* Oracle VirtualBox
* Xubuntu ISO Image

**5. Required Software**

1. Oracle VirtualBox
2. Xubuntu ISO (64-bit)
3. GCC Compiler

**6. Step 1: Download Oracle VirtualBox**

1. Open a web browser.
2. Search for **Oracle VirtualBox**.
3. Download the Windows host installer.
4. Run the installer and follow on-screen instructions.



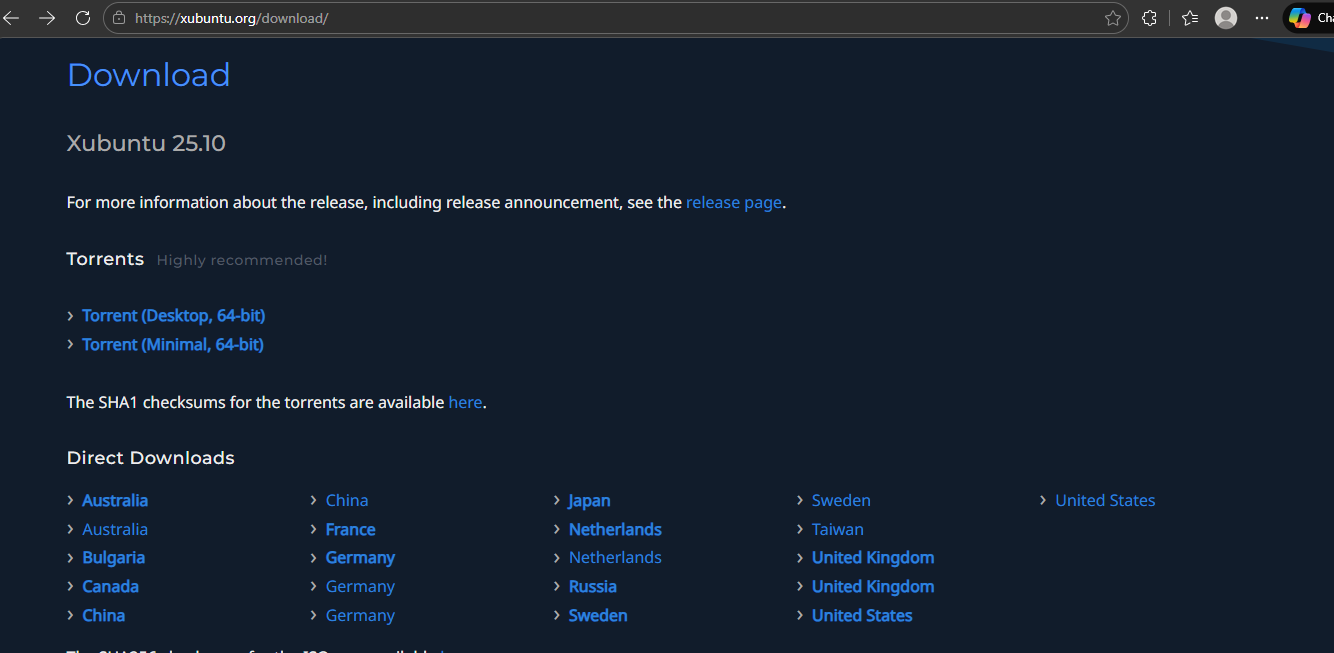
**7. Step 2: Install Oracle VirtualBox**

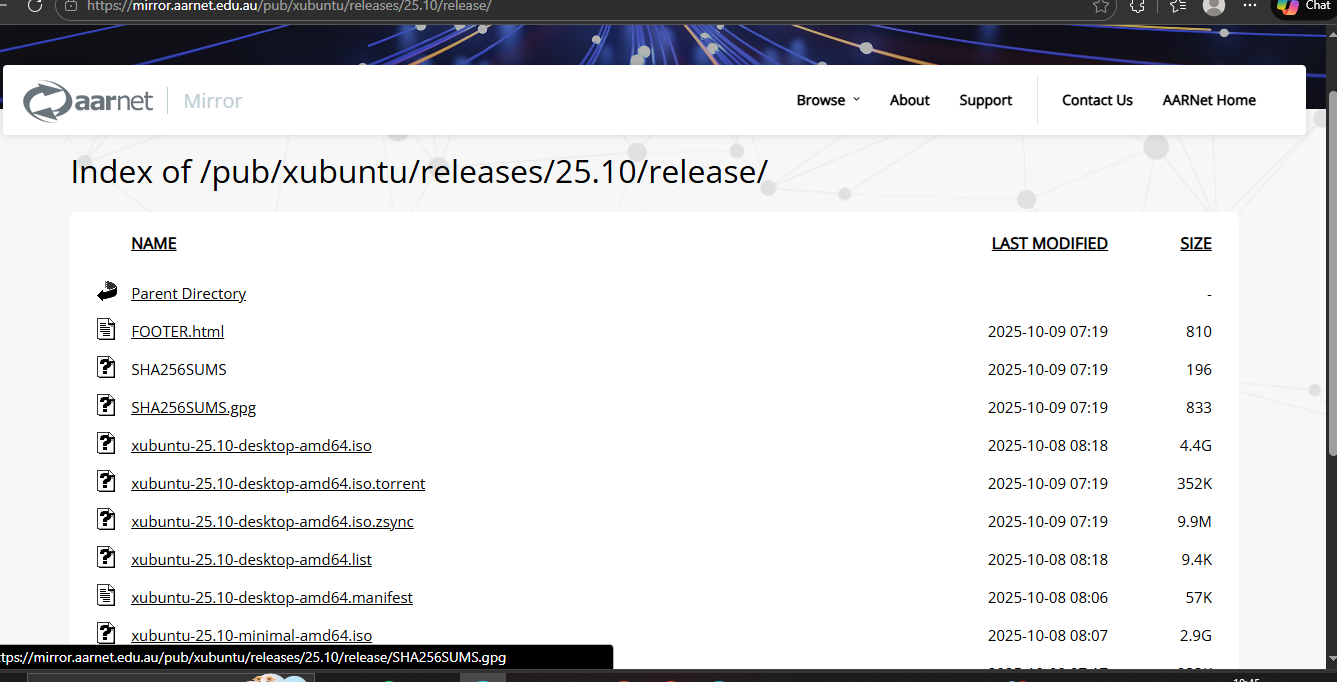
1. Launch the setup file.
2. Accept license agreement.
3. Keep default installation settings.
4. Complete installation.



**8. Step 3: Download Xubuntu ISO**

1. Visit the official Xubuntu website.
2. Download **Xubuntu Desktop (64-bit)** ISO.



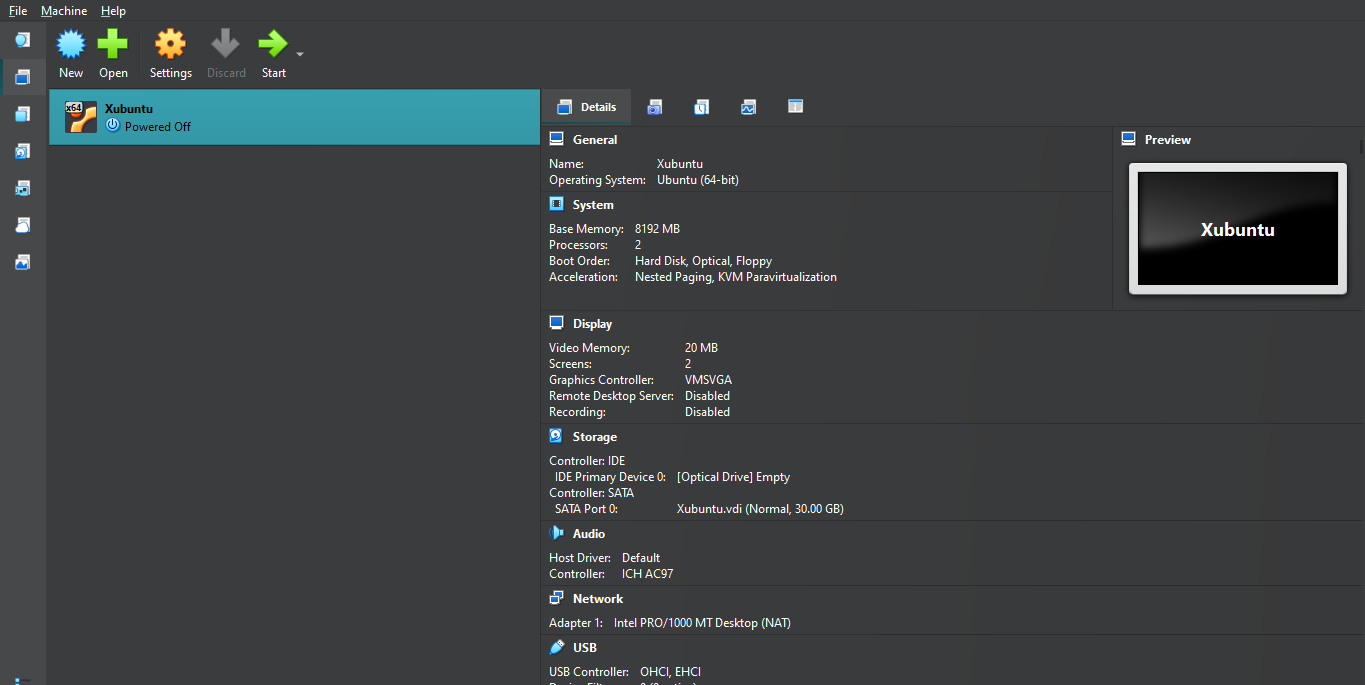


**9. Step 4: Create a New Virtual Machine**

1. Open Oracle VirtualBox.
2. Click **New**.
3. Enter name: Xubuntu OS.
4. Type: Linux, Version: Ubuntu (64-bit).
5. Select Xubuntu ISO file.

**10. Step 5: Configure Virtual Machine Settings**

1. Allocate RAM (minimum 2 GB, recommended 4 GB).
2. Allocate CPU cores (2 recommended).
3. Create virtual hard disk (VDI, dynamically allocated).
4. Set disk size to 20 GB or more.



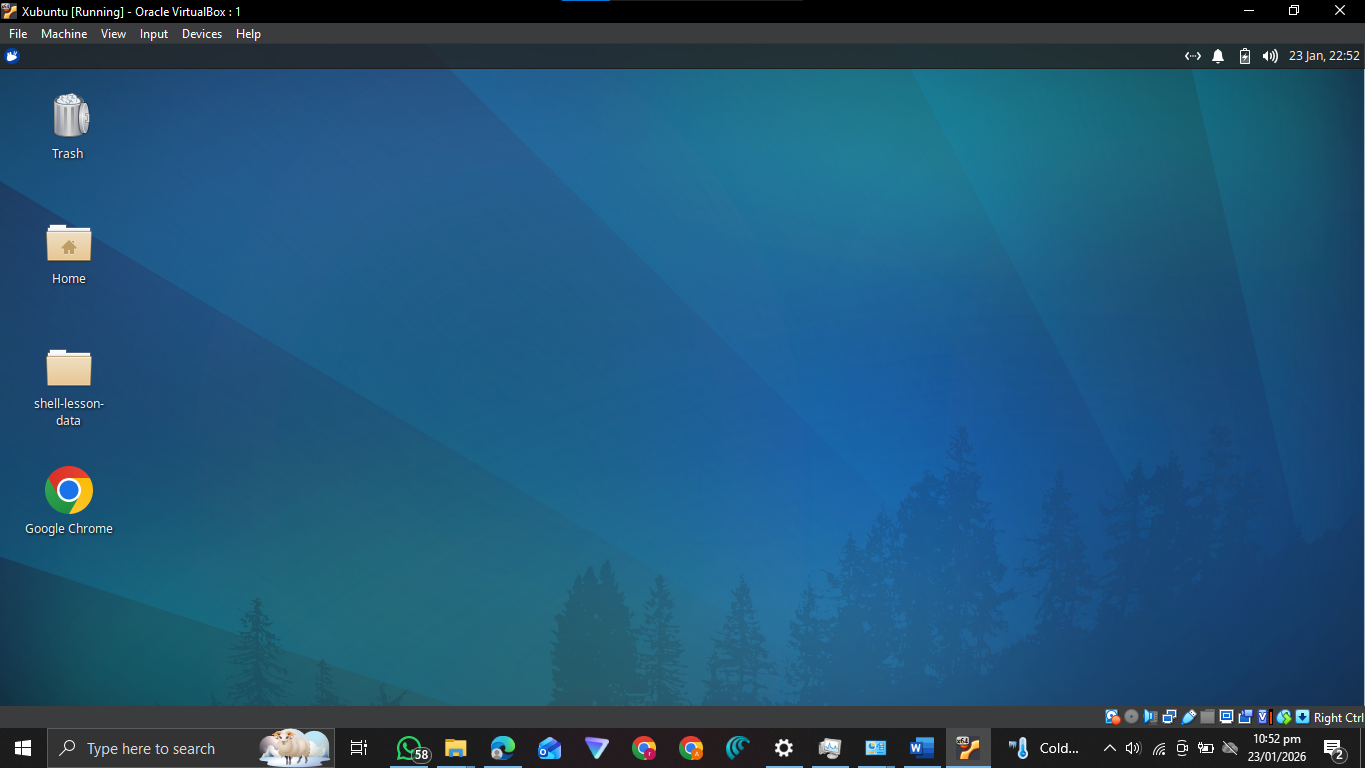
**11. Step 6: Install Xubuntu**

1. Start the virtual machine.
2. Select **Install Xubuntu**.
3. Choose language and keyboard layout.
4. Select **Normal Installation**.
5. Enable updates and third-party software (optional).
6. Choose **Erase disk and install Xubuntu** (safe in VM).
7. Set timezone, username, and password.

*Note: Xubuntu was already installed on the system, so there was no need to install it again. The relevant screenshot is provided below.*

**12. Step 7: Xubuntu Desktop Successfully Installed**

1. Restart the virtual machine.
2. Log in using credentials.
3. Xubuntu desktop appears.



**13. Step 8: Update System Packages**

Open terminal and run:

sudo apt update

sudo apt upgrade

**14. Step 9: Install Required Compilers (GCC)**

Install build tools using:

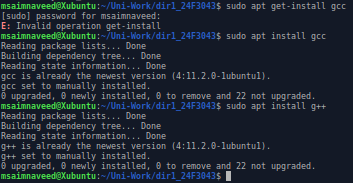
sudo apt install build-essential

Verify installation:

gcc --version

g++ --version

**Compilers Installation**

****

*Note: I already have xubuntu installed in my laptop because ubuntu is heavier and consumes much resources.*

*Xubuntu is the same as ubuntu, only default apps differ slightly (lighter apps in Xubuntu).*

**15. Step 10: Testing Compiler**

Create a test program:

nano test.c

Write:

#include <stdio.h>

int main() {

printf("Hello Xubuntu!\n");

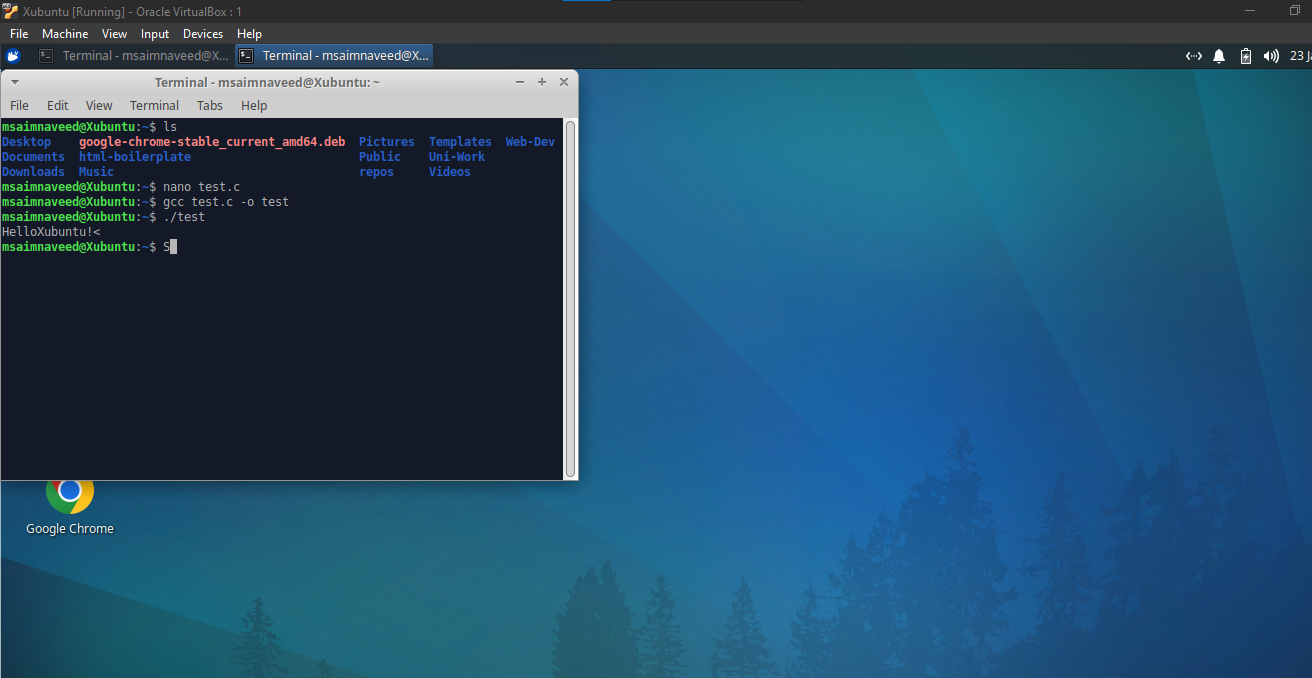
return 0;

}

Compile and execute:

gcc test.c -o test

./test



**16. Conclusion**

Xubuntu Linux was successfully installed on Oracle VirtualBox, and required compilers were configured. Xubuntu’s lightweight nature makes it ideal for academic use, especially on systems with limited resources. Oracle VirtualBox provided a free and efficient virtualization solution for this setup.

**17. Learning Outcomes**

* Understanding lightweight Linux distributions
* Installing Xubuntu on Oracle VirtualBox
* Virtual machine configuration
* Compiler installation and testing
* Basic Linux command-line usage

**End of Manual**

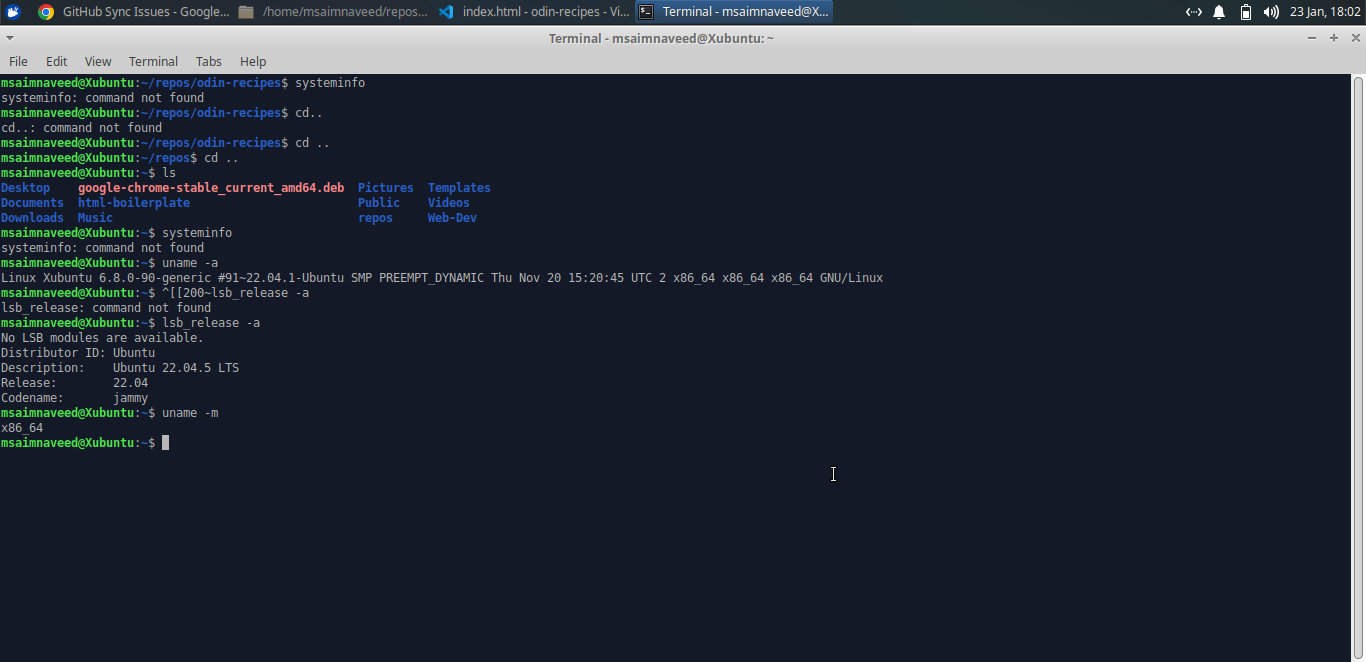
**TASK 2**

*Below screen shots are pasted by considering this* **: “Attach complete**

**screenshots for each task command. Any answer without screenshot or**

**cropped screenshot will result in zero marks in that task”.**

**Q1) show the architecture of the machine and mention whether it is 32-bit or 64-bit.**

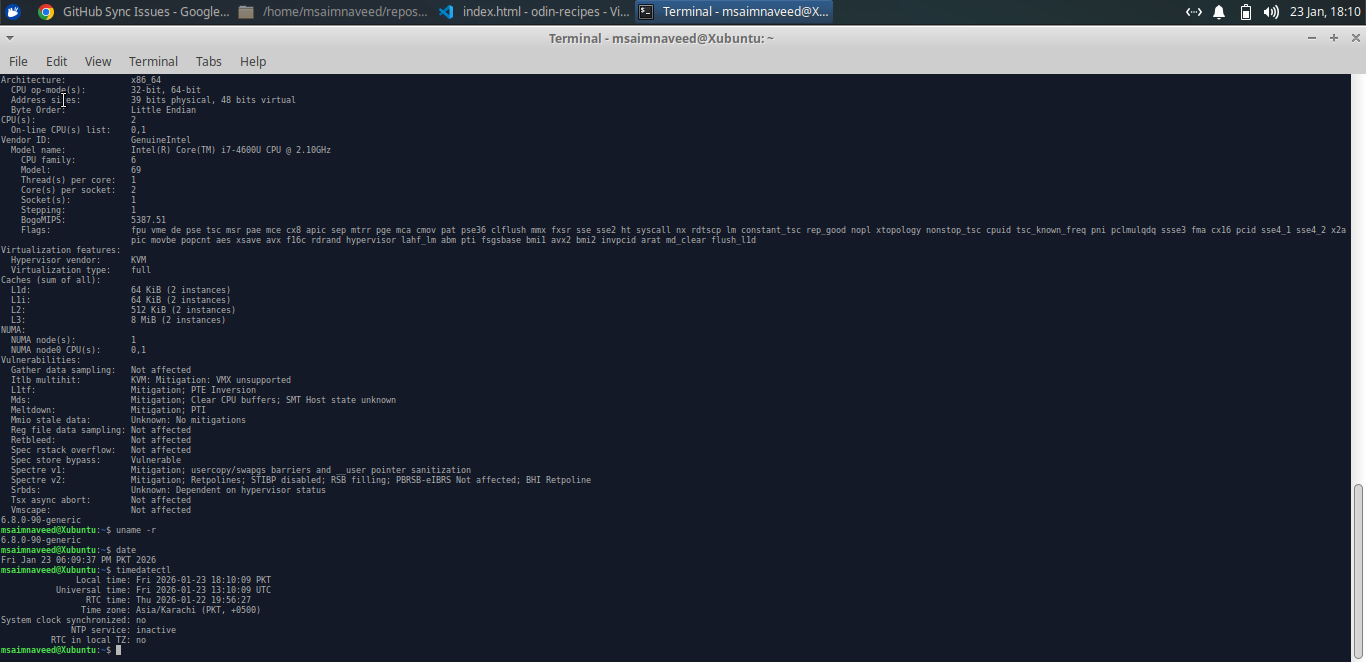


uname -m: give unix machine architecture

**Q2)** **display CPU information and write down the number of CPU cores available along with the version of the kernel**

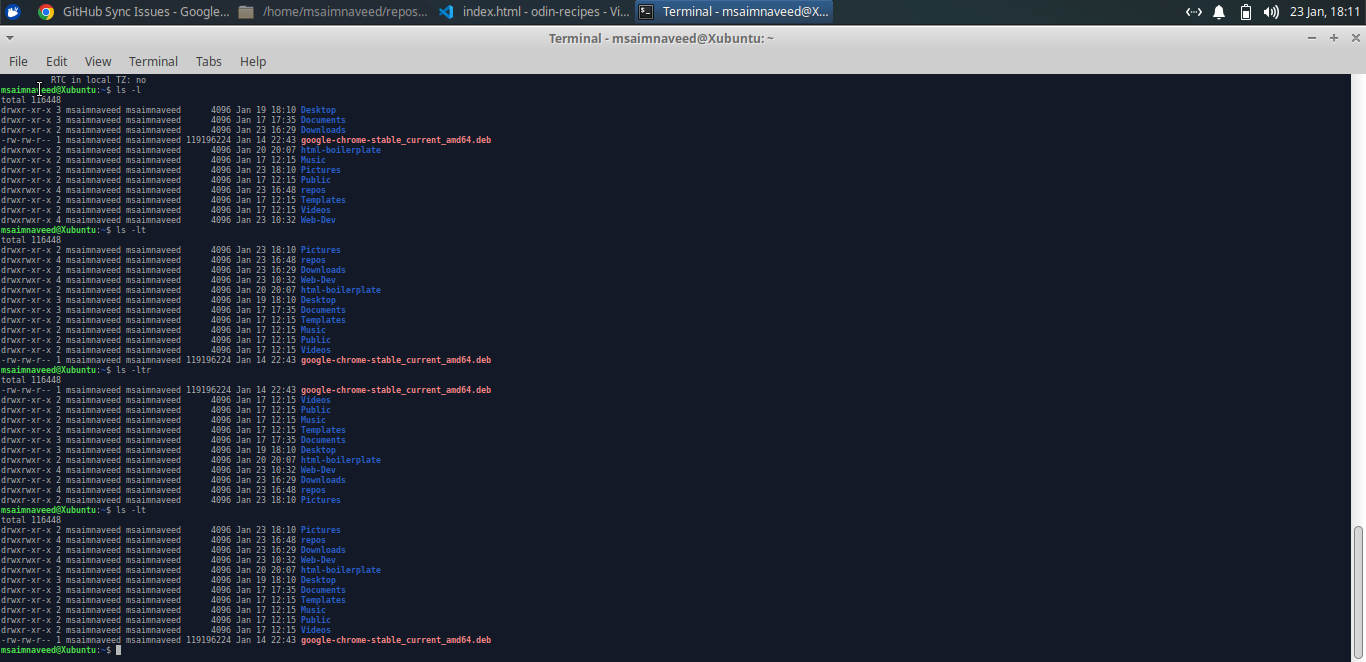
lscpu: display detailed CPU information

**Q3)** **show system date including time and timezone.**

****

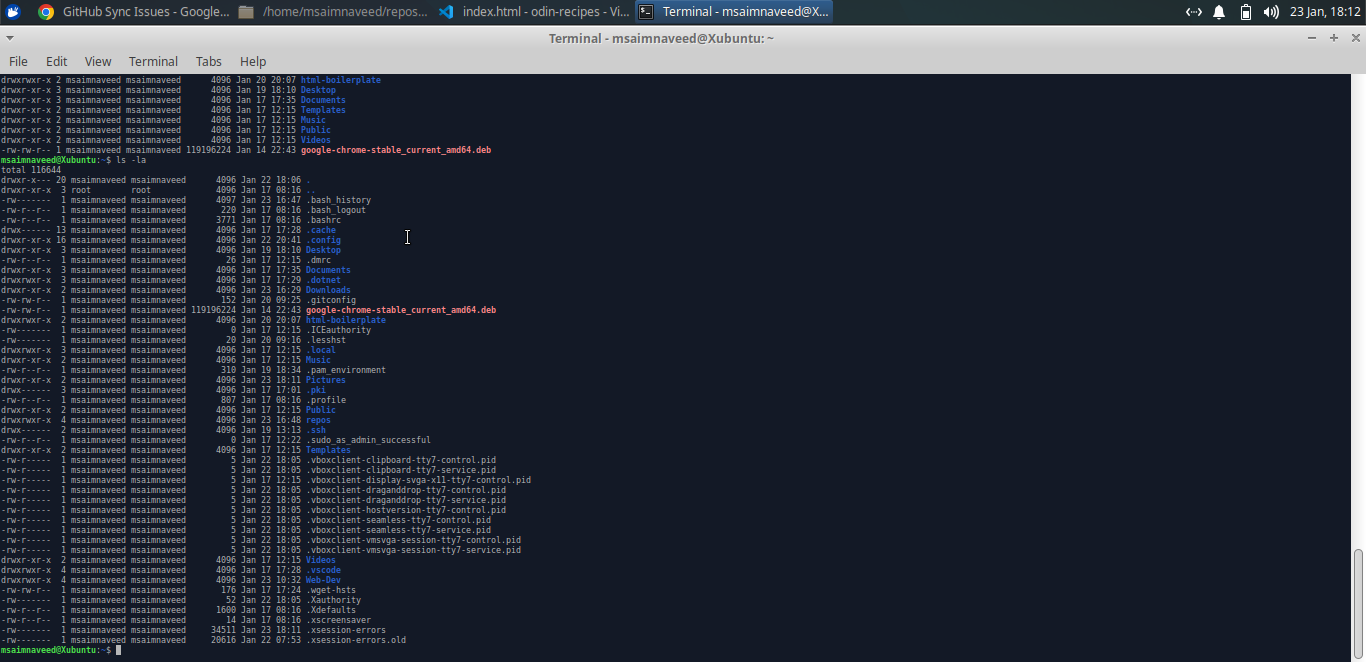
**Timedatectl:** for system time and timezone

**Q4)** **list files and directories in long format sorted by modification time.**

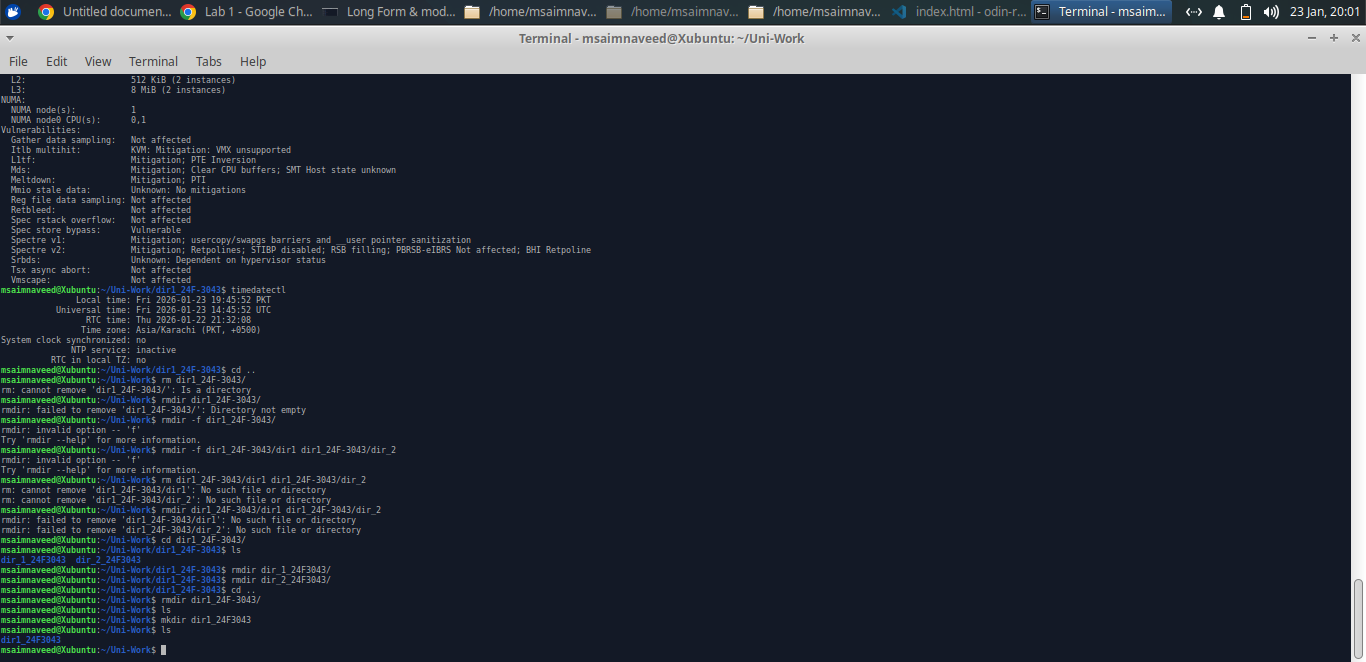
****

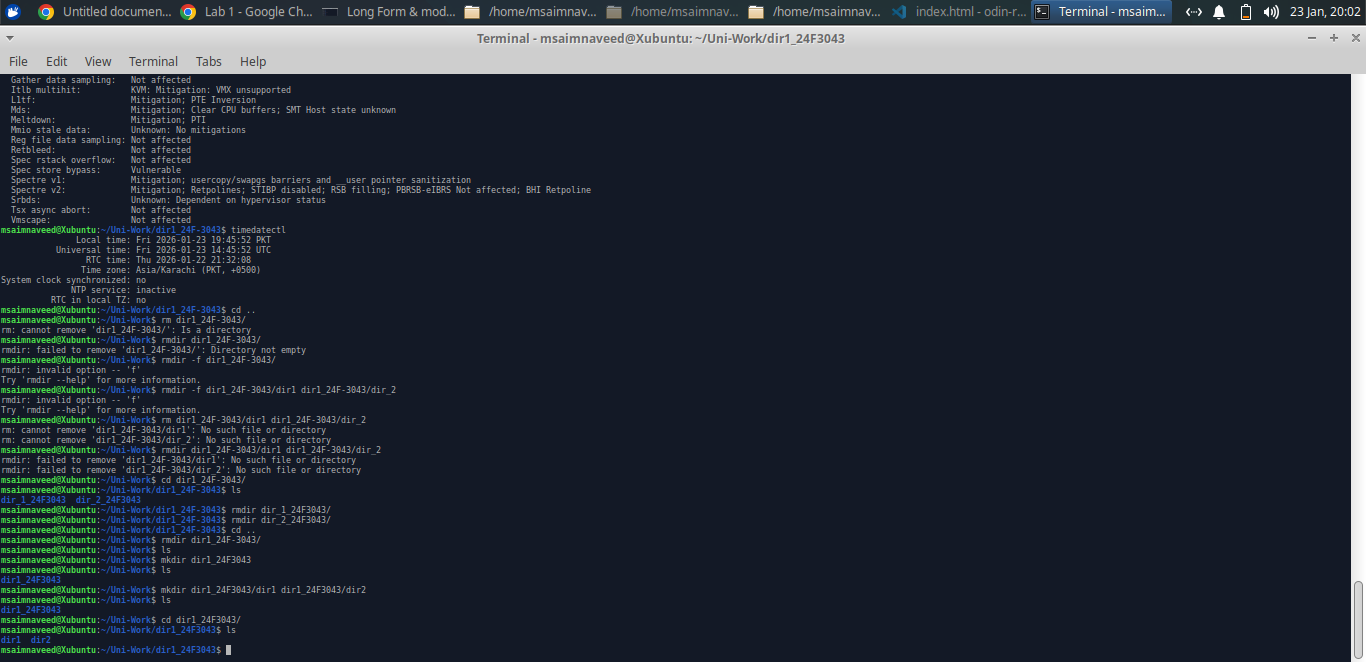
ls -lt:for long list and time sorting

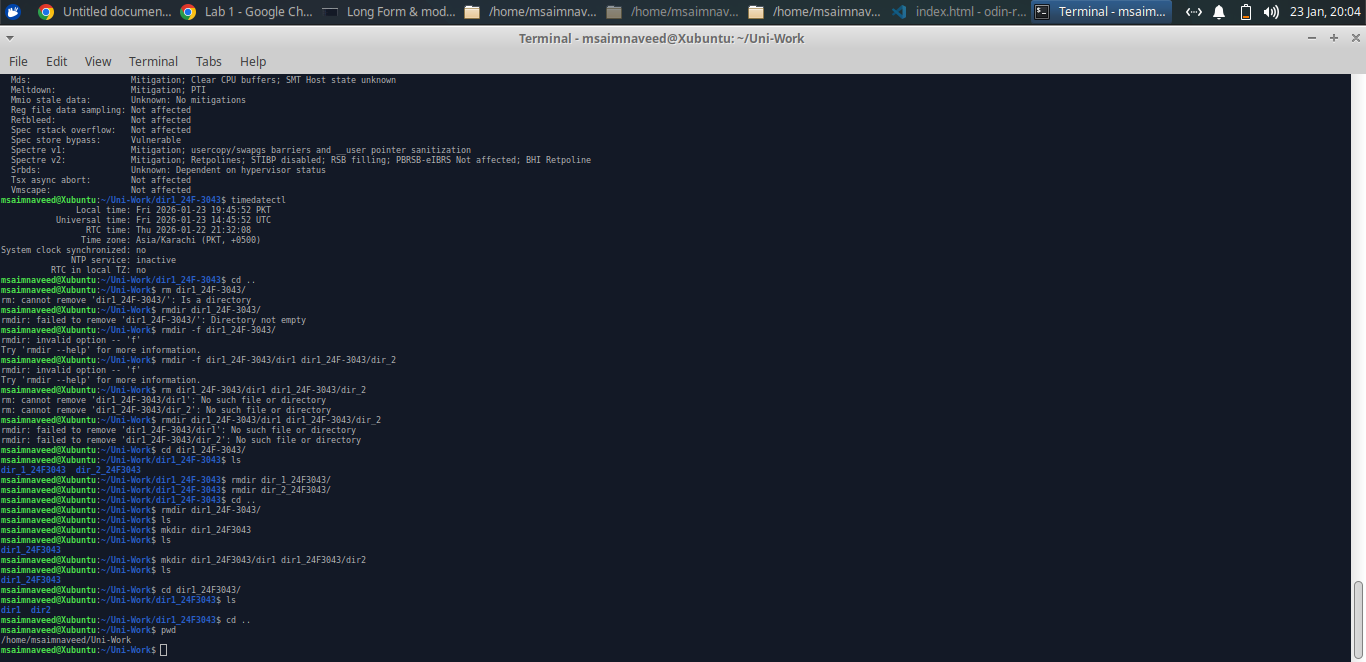
**Q5)** **show hidden files along with file permissions.**

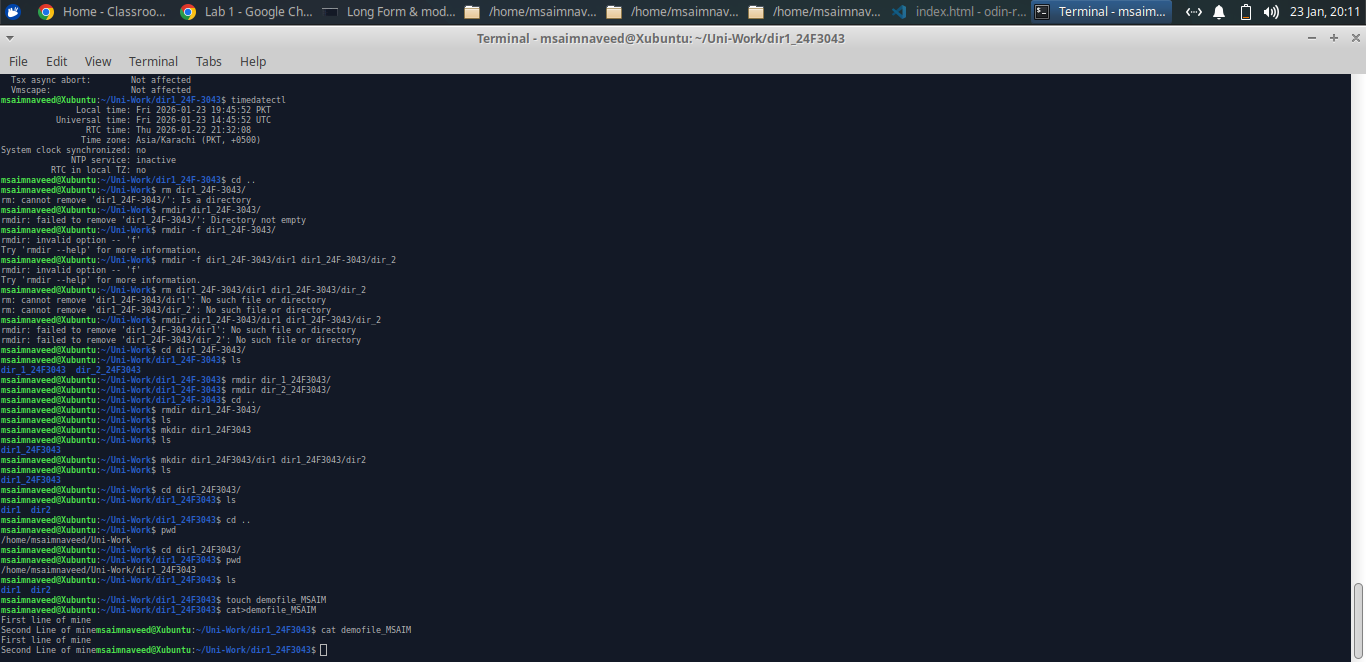
****

ls -la:show list of all files(including hidden) in long format

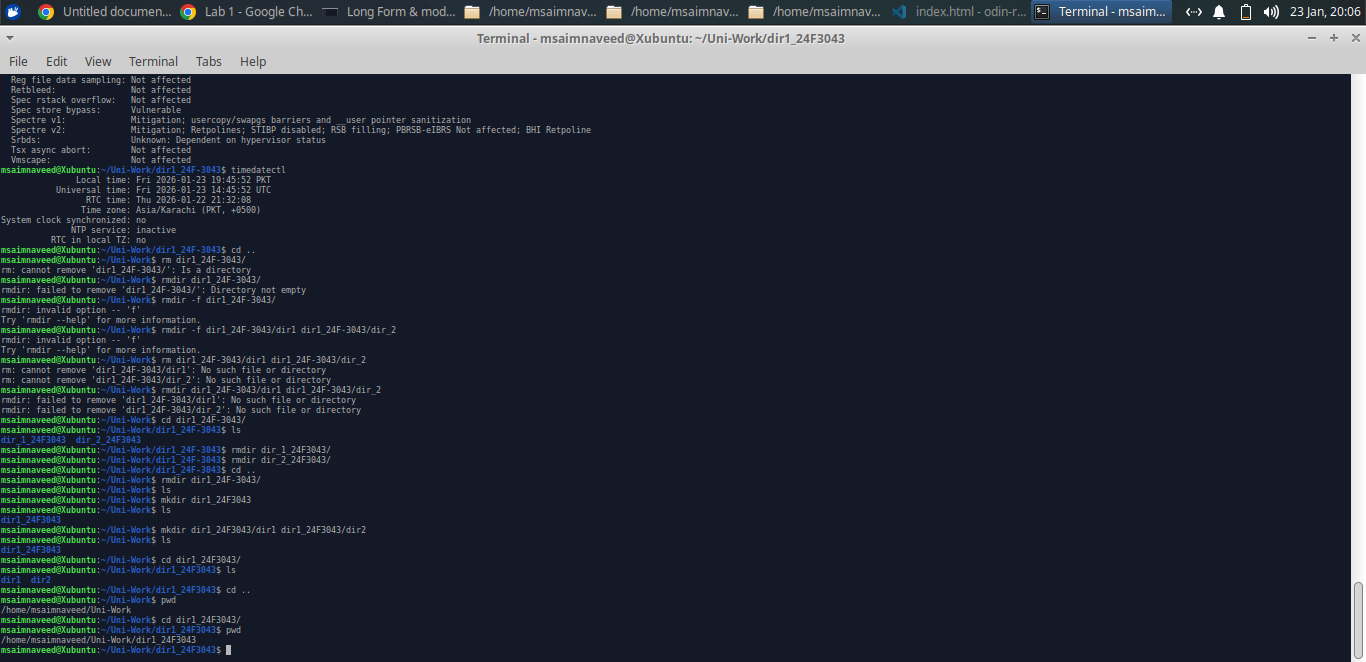
**Q6)** **create a directory called 'dir1\_YourRollNo’ inside your current working directory and confirm its creation. **

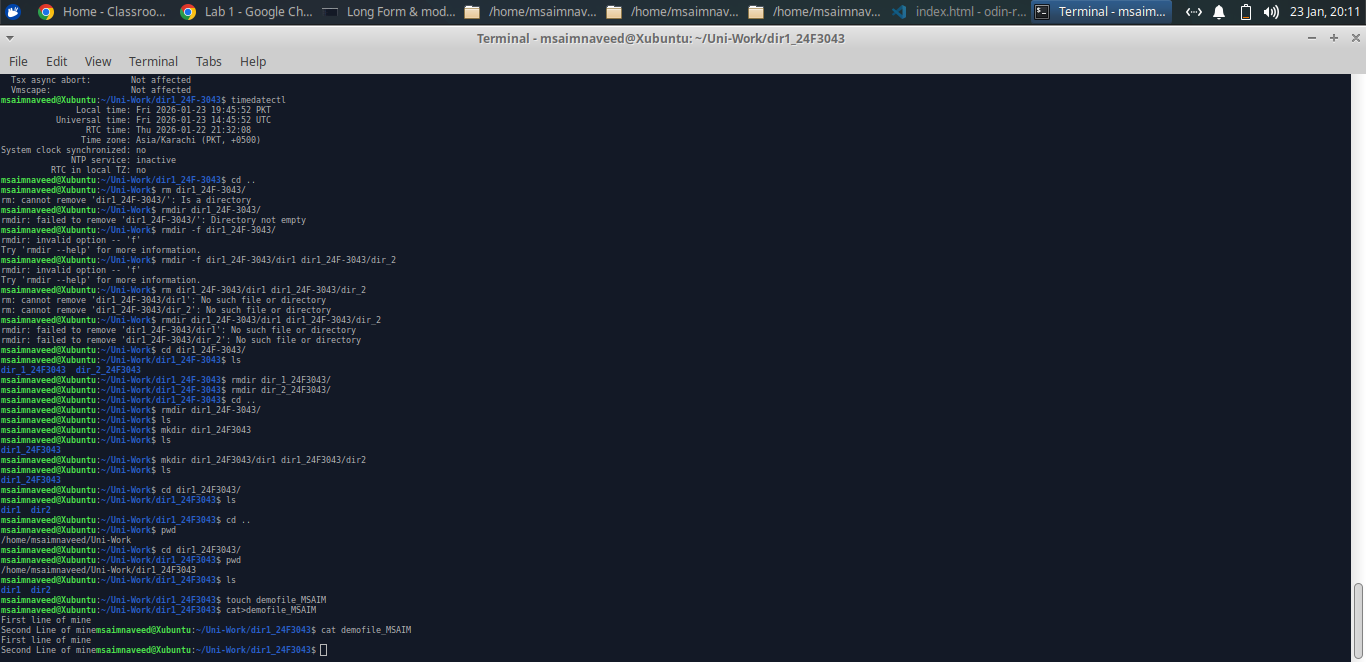
**Q7)** **create two directories simultaneously inside ‘dir1\_YourRollNo’. **

**Q8)** **display the absolute path of the current working directory and verify it changes when you move into ‘dir1\_ YourRollNo’. **

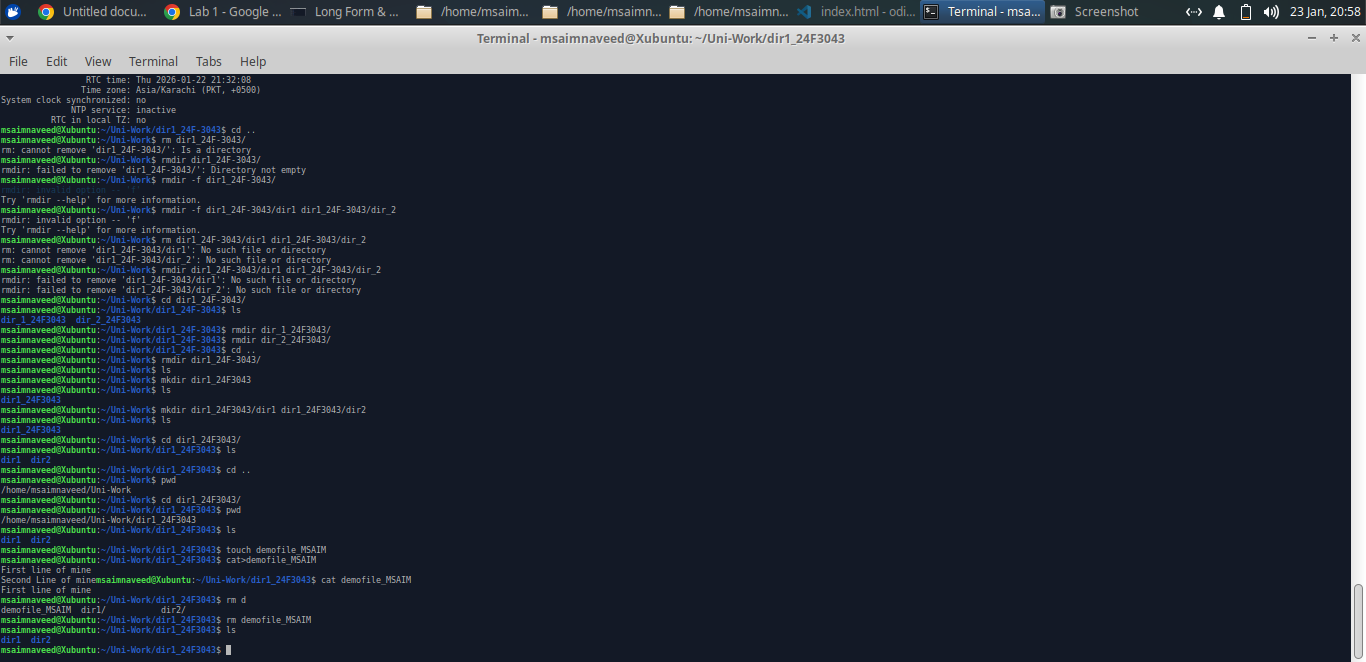
****

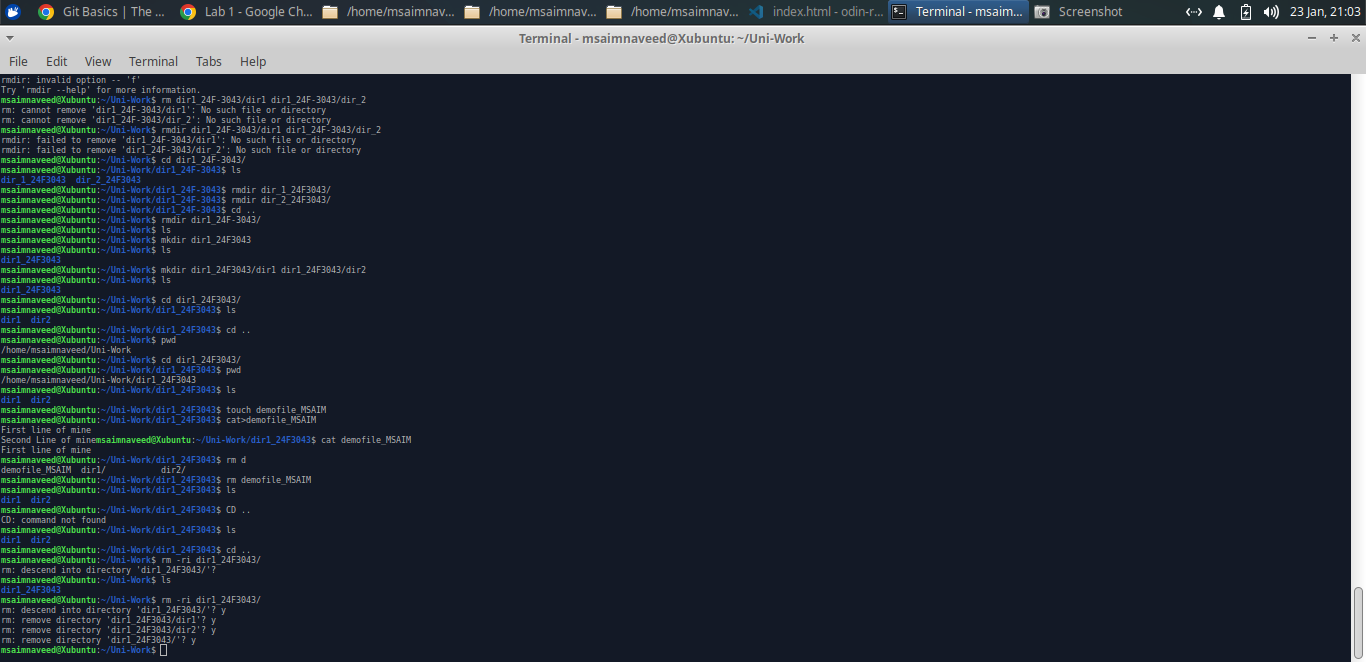
**Q9)** **create a file named ‘demofile\_YourFirstName’ and add at least two lines of text to it.**

****

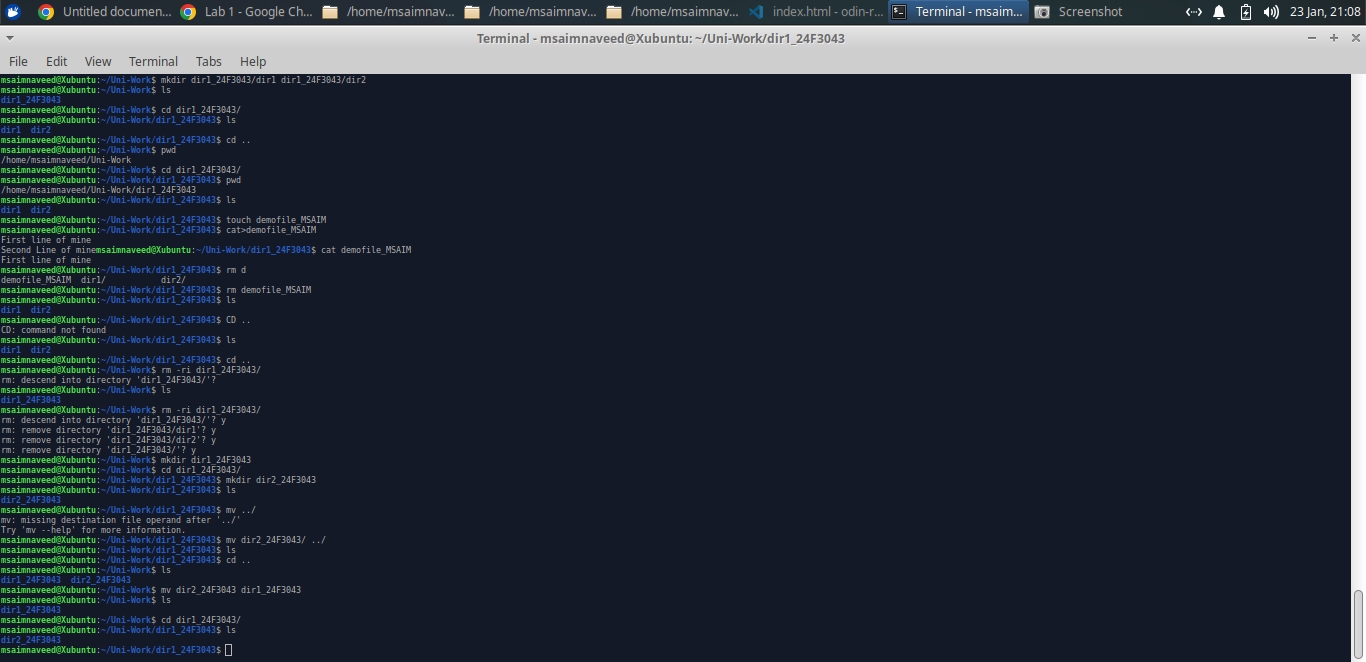
**Q10)** **display the contents of file called 'demofile\_YourFirstName’ on the screen before deleting it.**

*After removing demo file*

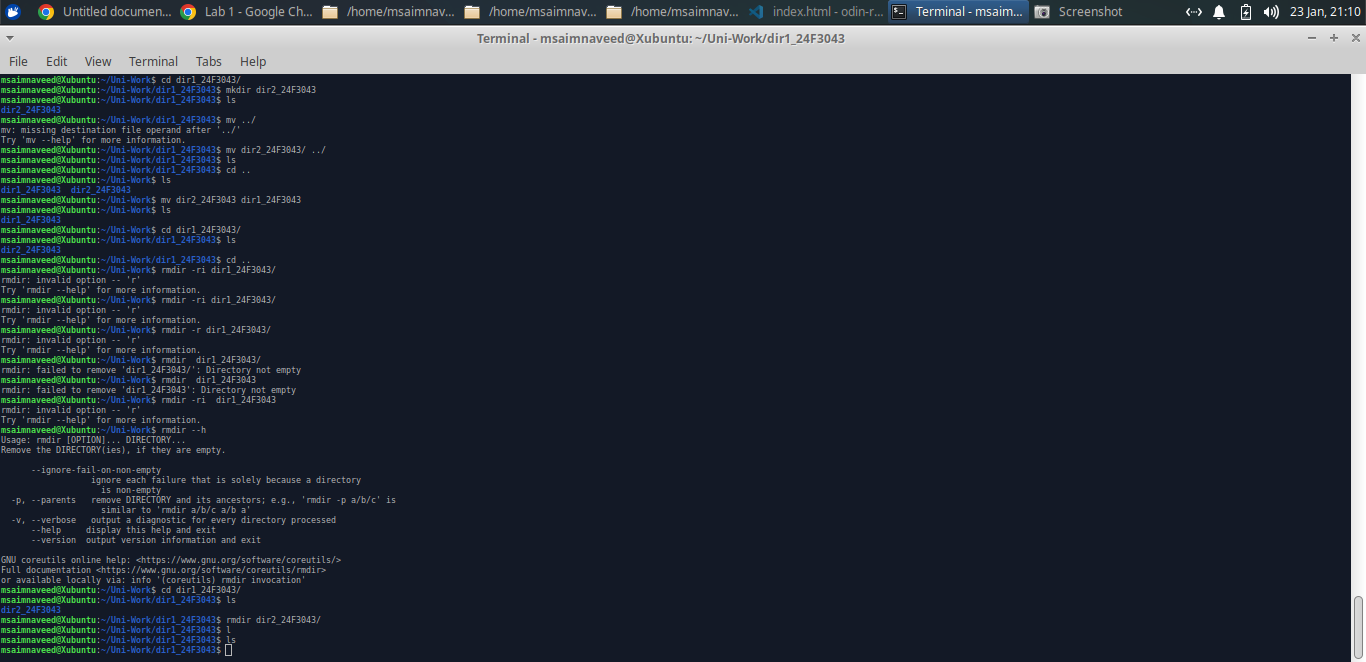
**Q11)** **remove the directory called 'dir1\_YourRollNo' only after confirming it contains subdirectories.**

****

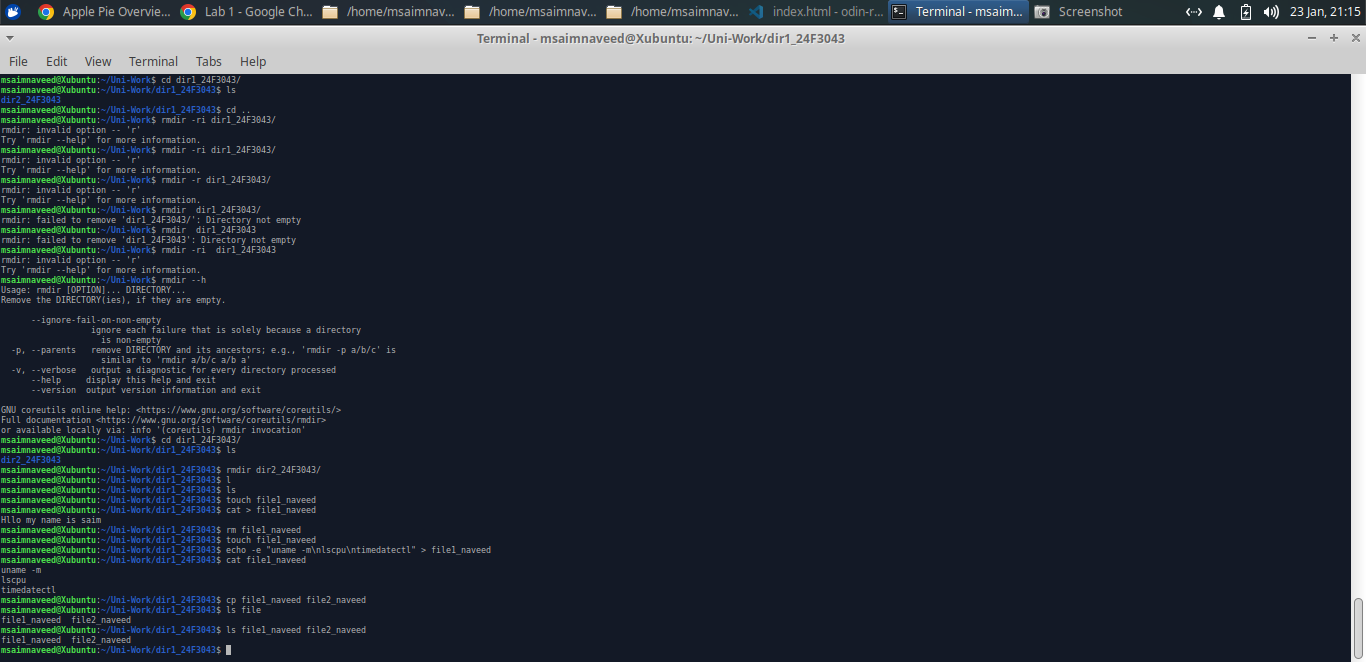
**Q12)** **Create a directory called 'dir2\_YourRollNo’ and move into it, then delete it from its parent directory.**

****

*Create dir 2 and mov in dir 1*

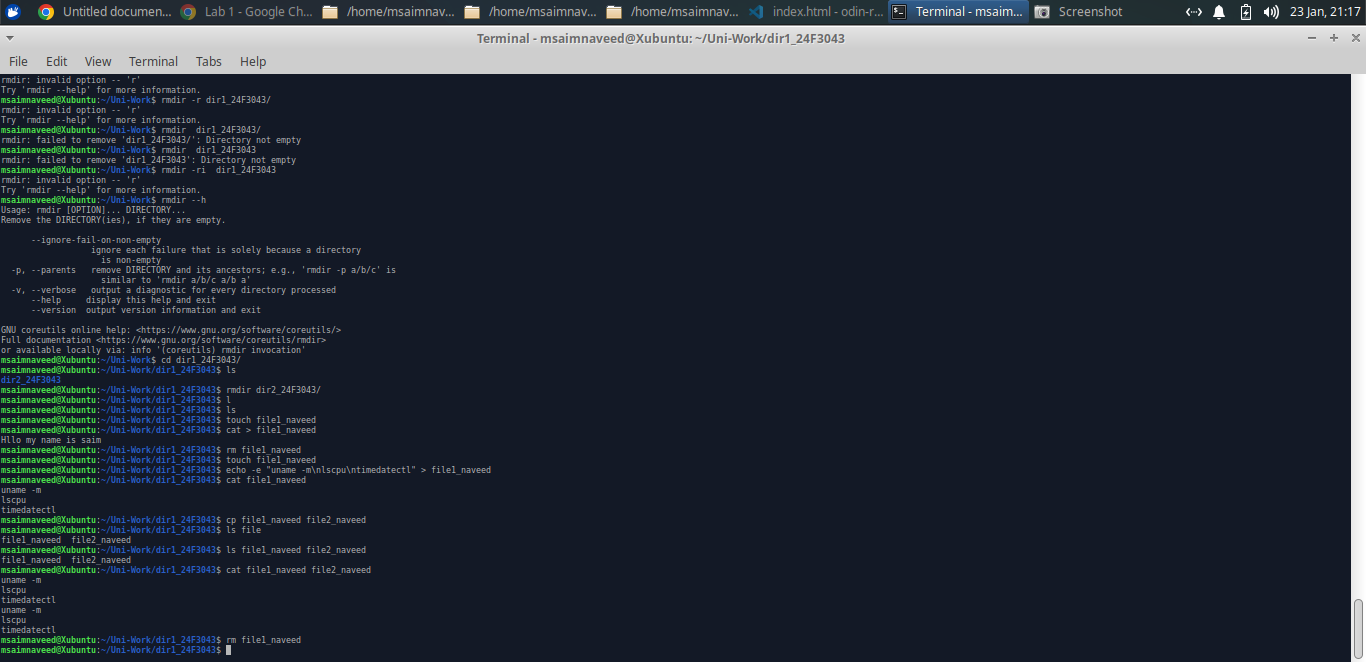
****

**Q13)** **Create the file named ‘file1\_YourLastName’. Write any three commands you used earlier into the file. List the contents of the file ‘file1\_YourLastName’ to the screen. Make a copy of the file ‘file1\_YourLastName’ under the name ‘file2\_YourLastName’. Verify using a single command that the files ‘file1\_YourLastName’ and ‘file2\_YourLastName’ both exist**

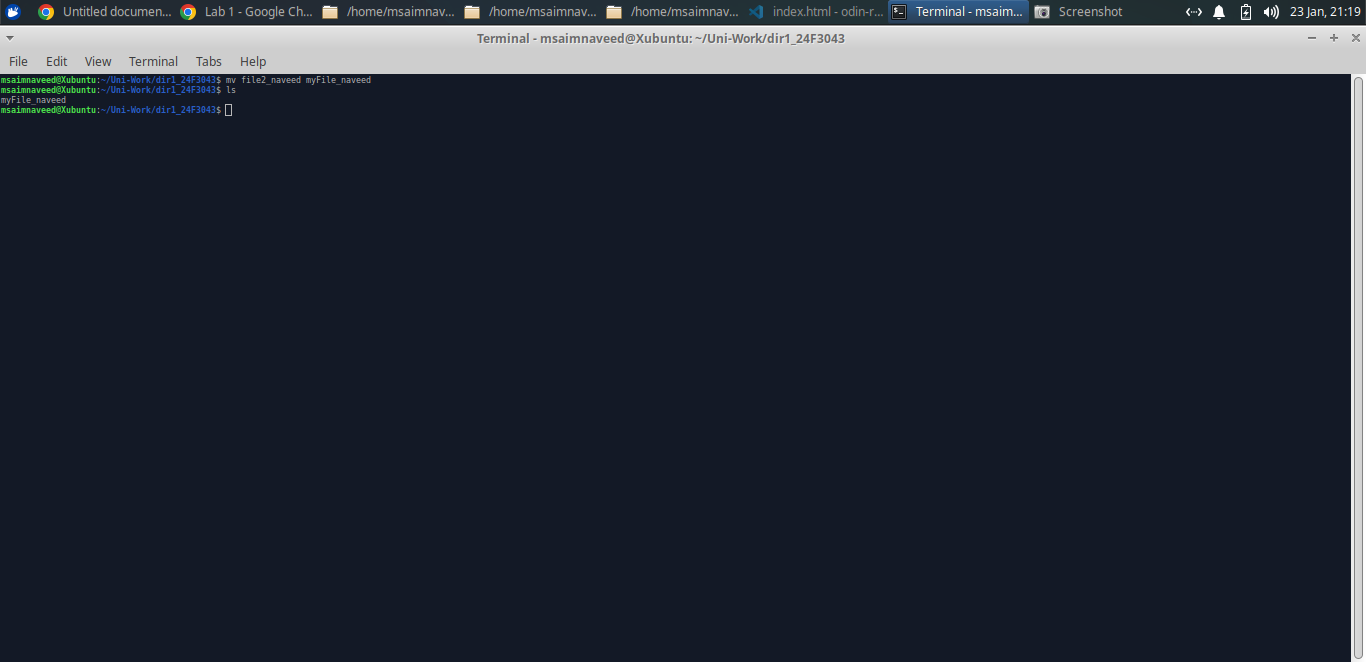
****

**Q14)** **List the contents of both ‘file1\_YourLastName’ and ‘file2\_YourLastName’ side by side or sequentially. Then delete the file ‘file1\_YourLastName’. Clear the terminal screen. Rename ‘file2\_YourLastName’ to ‘myfile\_YourLastName’.**

*Create file2\_naveed and remove file1\_naveed*

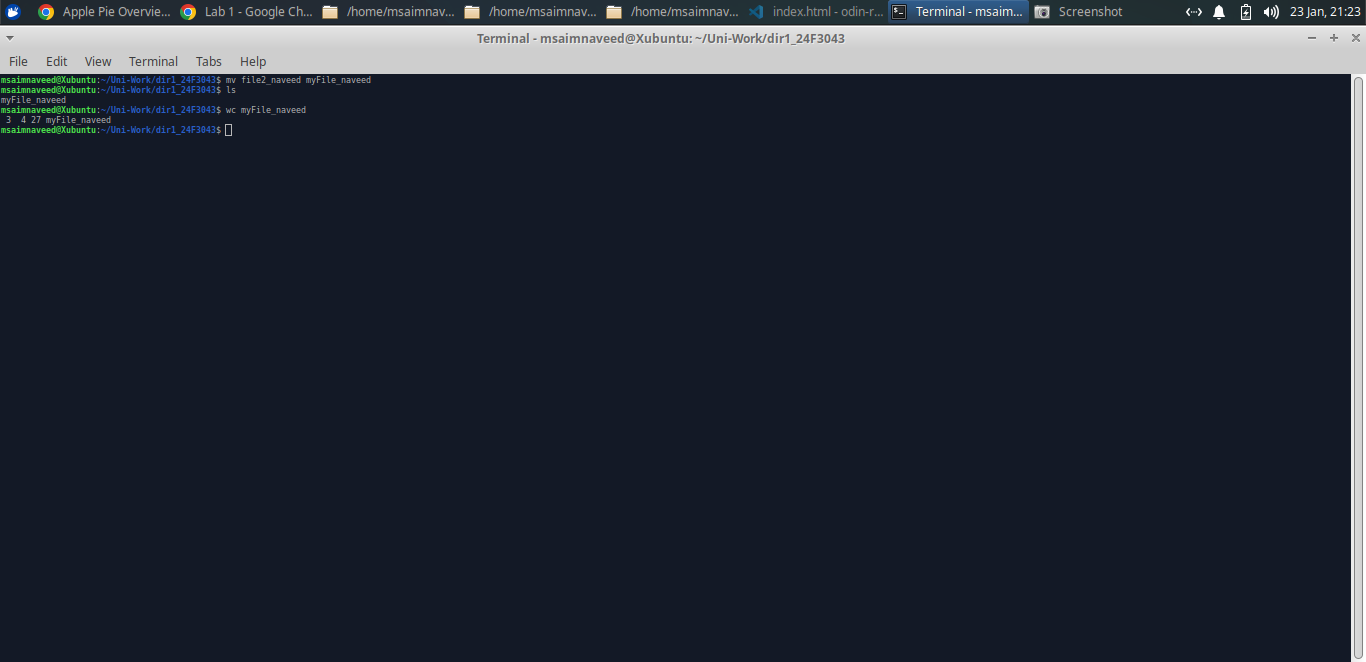
****

*Rename from file\_naveed to myfile\_naveed*

****

*mv is used for renaming files and folders*

**Q15)** **Count the number of characters, words, and lines of ‘myfile\_YourLastName’ and briefly state which count depends on file content.**

****

*All three counts depend on file content but word count depends on how many words are written and how they are separated.*

*Character count depends on the total characters used .*

*Line count depends on how many line breaks are present.*