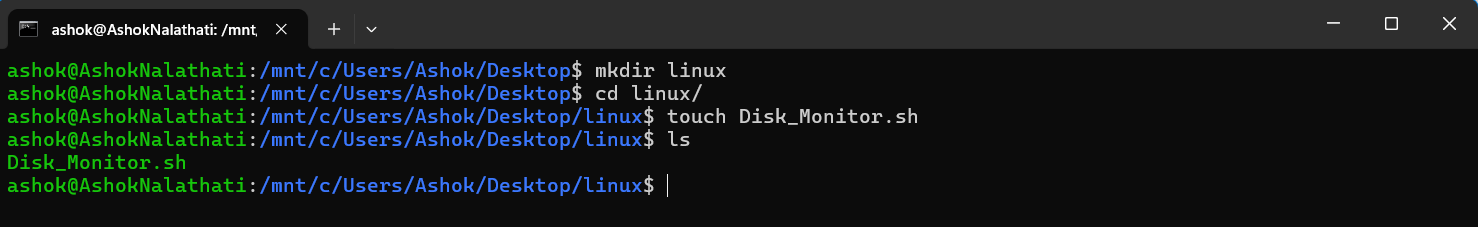
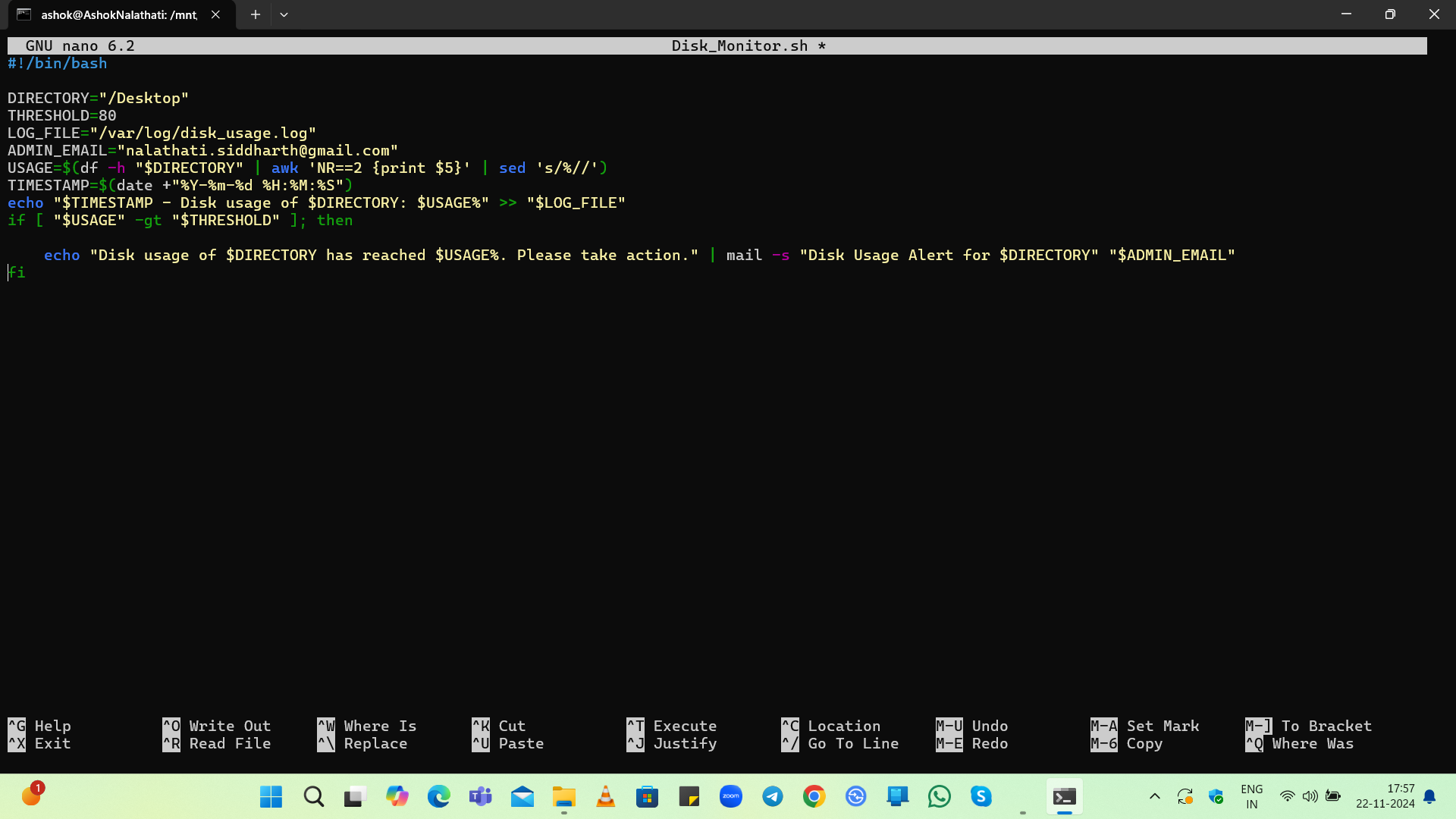
Disk-Monitor

Step 1: Create the Shell Script:

**Open a terminal** and navigate to the directory where to create the script (Desktop).



Create a shell script file



Explanation of code:

#!/bin/bash

This is called a shebang. It tells the system that this script should be run using the Bash shell.

DIRECTORY= "/home"

This line defines a variable named DIRECTORY and assigns it the value "/home". This is the directory whose disk usage will be monitored. Note that there should not be a space before the = sign.

THRESHOLD=80

This line defines a variable named THRESHOLD and sets it to 80. This means that if the disk usage exceeds 80%, an alert will be triggered.

LOG\_FILE= " /mnt/c/Users/Ashok/Desktop/linux "

This line defines a variable named LOG\_FILE and assigns it the path to the log file where disk usage information will be recorded. Again, there should not be a space before the = sign.

ADMIN\_EMAIL= "Nalathati.siddharth@gmail.com.com"

This line defines a variable named ADMIN\_EMAIL and assigns it an email address. This is the address where notifications will be sent if the disk usage exceeds the threshold.

USAGE=$( df -h " $DIRECTORY " | awk 'NR==2 {print $5}' | sed 's/%//' )

This line calculates the current disk usage percentage for the specified directory.

df -h "$DIRECTORY": The df command reports file system disk space usage. The -h option makes the output human-readable (e.g., in GB or MB).

awk 'NR==2 {print $5}': This part processes the output of df. It selects the second line (which contains the usage information for the specified directory) and prints the fifth column, which is the percentage of disk usage.

sed 's/%//': This removes the percentage sign from the output, leaving just the numeric value.

TIMESTAMP=$( date + "%Y-%m-%d %H:%M:%S" )

This line gets the current date and time in a specific format (YYYY-MM-DD HH:MM:SS) and assigns it to the variable TIMESTAMP. The date command is used to retrieve the current date and time.

echo " $TIMESTAMP - Disk usage of $DIRECTORY : $USAGE %" >> " $LOG\_FILE "

echo "...": This command outputs the formatted string containing the timestamp, directory, and usage percentage.

>> "$LOG\_FILE": The >> operator appends the output to the file specified by LOG\_FILE. If the file does not exist, it will be created.

if [ " $USAGE " -gt " $THRESHOLD " ]; then

The -gt operator is used for numeric comparison. However, there should not be spaces around the variable names for proper evaluation.

Step 2: Make the Script Executable

chmod +x Disk\_Monitor.sh

Step 3:Test the Script

./Disk\_Monitor.sh



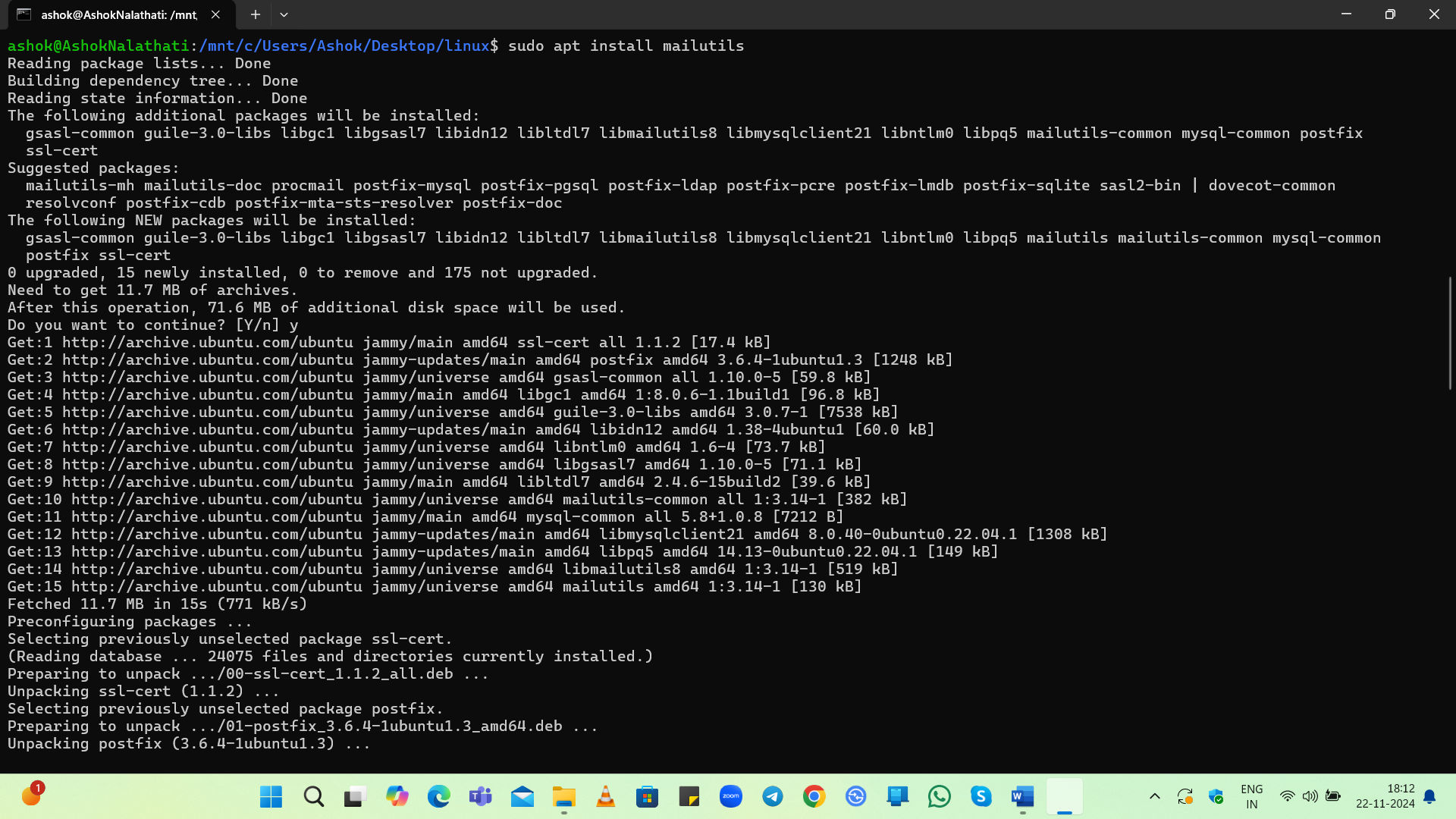
To check logs:

Cat /mnt/c/Users/Ashok/Desktop/linux



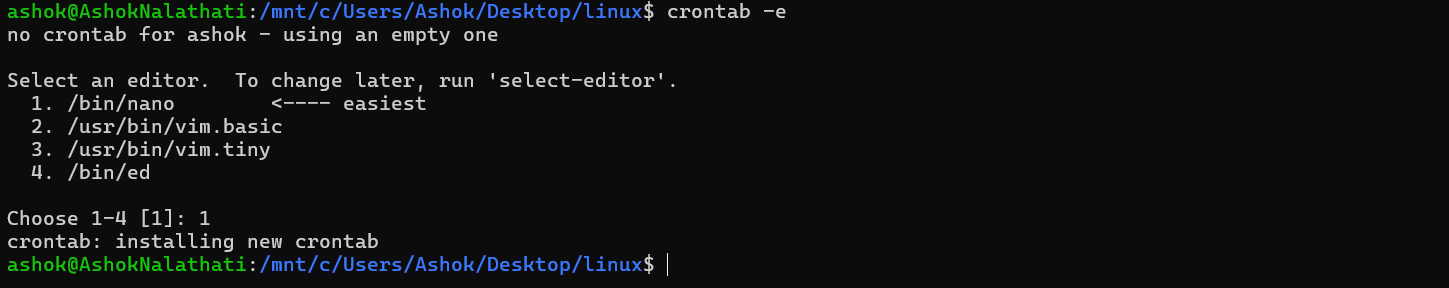
Step 4: Configure Email Notifications

sudo apt install mailutils



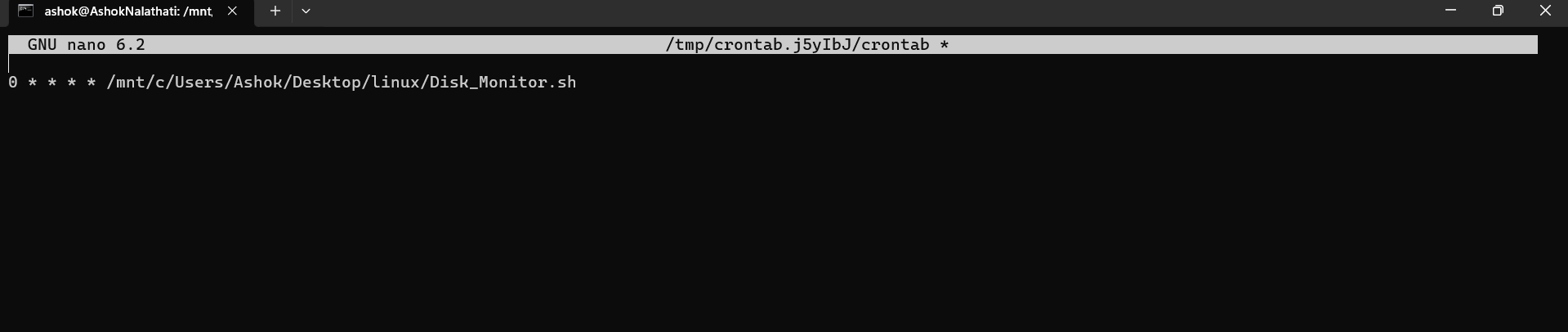
Step 5: Automate the Script

crontab -e



Add an entry to run the script every hour:

0 \* \* \* \* /mnt/c/Users/Ashok/Desktop/linux/Disk\_Monitor.sh



Save and exit. The script will now run automatically every hour.