CSE 301L: Operating Systems Lab

INTRODUCTION TO LINUX COMMANDS

INTRODUCTION TO LINUX

- Linux is an operating system that works just like Windows.
- Linus Torvalds released the Linux kernel as free, open-source software (1991).
- Open source means that the code is fully visible, and can be modified and redistributed
- Graphical User Interface(GUI) or Command line(shell) can be used to interact with Linux system

BASIC LINUX COMMANDS

- Linux commands can be given through terminal window.

- To open terminal window use

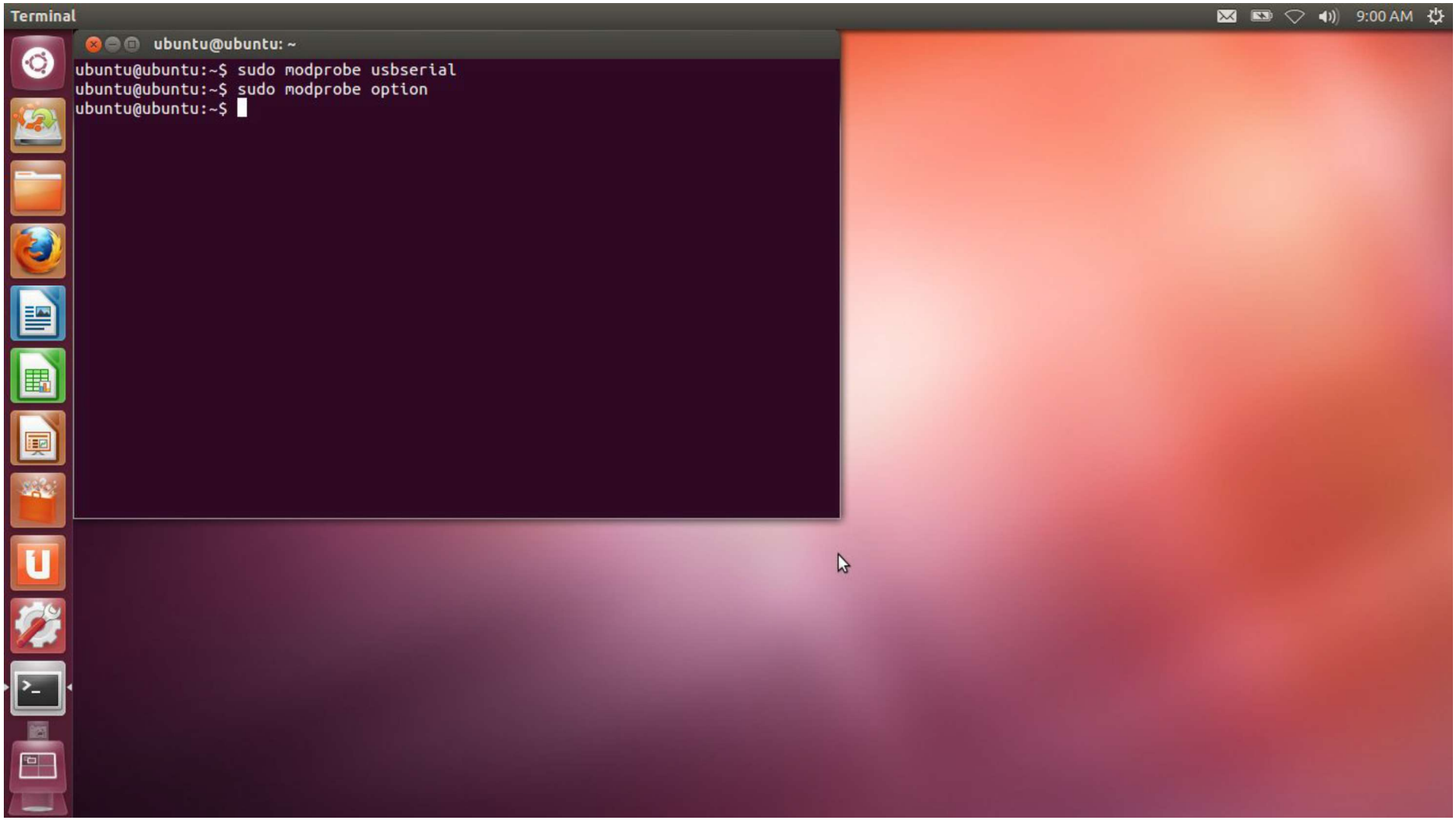
  ALT + CTRL + T

- To close the terminal

  CTRL + D

- Note : all Linux commands are case sensitive

BASIC LINUX COMMANDS



Linux Terminal

BASIC LINUX COMMANDS

- **ls**
  - It provides a list of the names of the objects (i.e., files and directories) in the current directory

BASIC LINUX COMMANDS

mkdir

- Stands for make directory
- To create new directory
- Syntax : mkdir directory_name(s)
- Eg :- mkdir students
  mkdir cricket tennis football

rm

- Remove files or directories
- Syntax : rm file_name(s)
- Eg :- rm students
  rm cricket tennis football
  rm *.txt
  rm lect*

BASIC LINUX COMMANDS

- **rmdir**
  - Stands for remove directory
  - Another way to remove empty directories.
  - Syntax : rmdir directory_name(s)
  - Eg :- rmdir students
    rmdir cricket tennis football
- **vi**
  - We can edit an existing file or create a new file from scratch.
  - Syntax : vi filename
  - Eg :- vi resume.txt
  - Esc and typing **:q!** (exit without saving)
  - Esc and typing **:wq**

BASIC LINUX COMMANDS

- **rm -r**
  - Stands for remove directory
  -

BASIC LINUX COMMANDS

- **pwd**
  - Stands for present working directory
  - Shows the name and location of the current directory

BASIC LINUX COMMANDS

- **cd**
  - Stands for change directory
  - Syntax cd path_name
  - Eg :- cd football
- **head**
  - Output the first part of the file
  - Eg :- head -5 resume.txt
  - Displays the first 5 lines from the file resume.txt
- **tail**
  - Output the last part of the file
  - Eg :- tail -3 resume.txt
  - Displays the last 3 lines from the file resume.txt

BASIC LINUX COMMANDS

- **cat**

  - Concatenate files and print the content.
  - Eg :- cat file1.txt file2.txt file3.txt
  - Displays the concatenated contents from all the three files

- **wc**

  - Count the number of lines, words or bytes of a file.
  - Eg :- wc -l file1.txt (No of lines)
    wc -c file1.txt (No of characters)
    wc -w file1.txt (No of words)

BASIC LINUX COMMANDS

cp

- Stands for copy
- Can be used to copy files or directories
- Copies become independent of originals (i.e., a subsequent change in one will not affect the other)
- Syntax : cp old_name new_name
- Eg :- cp resume.txt biodata.txt
  cp teachers professors
- Multiple files can be copied to a single directory in a single step.
- Eg :- cp file1 file2 file3 files

BASIC LINUX COMMANDS

- **mv**

  - Stands for move
  - Can be used to rename or move files and directories.
  - Syntax : mv old_name new_name
  - Eg :- mv resume.txt curriculumvitae.txt
    mv teachers employees
  - Files or directories can be moved to another directories with old/new name.
  - Eg :- mv file1 employees/file1
    mv file1 employees/newfile1

BASIC LINUX COMMANDS

zip

- Used to package and compress files
- Syntax : zip new_dir_name filename(s)
- Eg :- zip details.zip f1.txt f2.txt f3.txt

unzip

- Used to extract compressed files from zip archive.
- Syntax : unzip details.zip
- Eg :- zip details.zip

BASIC LINUX COMMANDS

- **date**

  - Print the system date and time
  - Eg:- date
  - [display the system date and time]

BASIC LINUX COMMANDS

- **grep**
  - Print lines of matching a pattern
  - Syntax : grep keyword filename
  - Syntax : grep apple fruits.txt
  - Display all line in file fruits.txt that contain the word apple

BASIC LINUX COMMANDS

- **last**
  - Display the details of last logged in users
- **last reboot**
  - Displays the last reboot time
- **free**
  - Displays the used and free memory in the system.

LINUX COMMANDS FOR C-PROGRAMMING

- **To create a c-file**
  - Use gedit filename.c
  - Type the c-program in the editer
  - Save the file.
- **To compile a c-file**
  - Use **cc** command
  - Syntax :- cc filename.c
  - Eg :- cc pgm1.c
- **To execute a compiled file**
  - Last compiled file will be named as **a.out**, which will be executable.
  - To execute the file use ./a.out

pgm1.c

```
#include<stdio.h>
int main()
{
printf("WELCOME TO IIITDM");
Return(0);
}
```

PRACTICE QUESTIONS

1. List all the files and directories
2. Check system date and time
3. Create a Parent called **Sports**
4. Create text files called **Team1.txt, Team2.txt** and **Team3.txt** in the **Sports** directory.
5. Create a sub-directory **Cricket** in **Sports** directory
6. Create a sub-directory **Hockey** in **Sports** directory
7. Create a sub-directory **Football** in **Sports** directory
8. Rename the directory **Hockey** with **Badminton**
9. Create one text file **Sample.text** in **Hockey, Football** and **Badminton** sub-directories.
10. Move to Badminton directory and change the name of **Sample.txt** to **BadmintonTeam.Txt**

PRACTICE QUESTIONS

11. Display the number of lines in **Team1.txt**
12. Display the first 3 lines of **Team2.txt**
13. Display the concatenated contents from the files **Team2.txt** and **Team3.txt**
14. Display the number of lines in **Team1.txt**
15. Remove all the text files in **Sports** directory
16. Copy all the contents of **Hockey** to another directory **Game**
17. Remove the **Hockey** directory