**Section 3 - Duration : 2 hr 36 mins**

1.How to open and close the terminal

* Keyboard shortcut : ctrl+alt+T - open
* Keyboard shortcut : ctrl+T - close

2.Our first commands

* echo - printing words
* Cal - calendar
* Up arrow - last command
* History - Shows history
* Exit - Exit terminal

3.Terminals, commands and shells

* Commands are basic text which gets interpreted by shell.
* Terminal is the mediator in between command and shell.
* The terminal is nothing more than just a window into the shell
* Most common type is bash shell
* Shell is a program to interpret command.
* Commands are case-sensitive.

4.Understanding command structure

* Structure: <CommandName> <options> <inputs>.
* Shortform options use '-'.
* Longform options use '--'.

5.Using Linux Manuals Part 1

* Structure

|  |  |
| --- | --- |
| Section | contents |
| 1 | User commands |
| 2 | System calls |
| 3 | C Library functions |
| 4 | Devices and special files |
| 5 | File formats and Conventions |
| 6 | Games |
| 7 | Miscellaneous |
| 8 | System administration |

* System administrator is used by only administrator.

6.Using Linux Manual Part 2

* Command for manual : 'man' .
* '-k' option is used for manual searching of commands.
* Man <Section number> <Command>.
* If no section number is provided, it will search in section 1.
* 'q' to exit the manual.
* Synopsis shows how to use the commands.
* For ex:- $ man -k which

7.Using Linux Manual Part 3

* If the instructions is not present in manual we use 'help' command to find instructions.

8.Command Input and Output

* Standard data stream: Standard input, Standard Output, Standard error
* Standard error : where error and log messages go.
* Standard I/P : defaultly connected to keyboard.
* Standard O/P : defaultly connected to Terminal.
* Standard error : defaultly connected to Terminal.

9.Redirection Part 1

* Cat- print content of file.
* cat > output.txt - Type content and appends in the file output.txt.
* ctrl+c - to get back to command line exit the cat command.
* ">" - overwrites the file.
* ">>" - appends in the present file.

10.Piping

* A pipe is a form of redirection that is used to send the output of one command or program or process to another for further processing.
* " | " pipe in linux.
* Example : date | cut --delimiter " " --fields 1.

11.Tee command

* tee command reads standard input (stdin) and writes it to both standard output (stdout) and one or more files.
* It basically breaks the output of a program so that it can be both displayed and saved in a file.
* It does both the tasks simultaneously, copies the result into the specified files or variables and also display the result.
* Example : date | tee fulldate.txt | cut --delimiter = " " --field = 1
* the above command will display the output and save the file in home directory as fulldate.txt.
* Example : date | tee fulldate.txt | cut --delimiter = " " --field = 1 > today.txt
* the above command will create and save both fulldate.txt and today.txt file.

12.Xargs command

* It is used to build and execute commands.
* $ command -options arguments
* Example: date | xargs echo (or) date | xargs echo "hello"
* some commands only accept command line arguments in that case xargs can be used.

13.Aliases

* Aliases are custom made command name used for a user defined function.
* .bash\_aliases - Create this file to create custom commands
* Example - alias <command name>=' <commands>…<commands>'