**Faculty of Computing**



**Operating Systems**

**LAB # 03**

**Instructor**

Ayesha Akram

Faculty of Computing,

Riphah International University, Islamabad

**Linux Basic Commands (Part - I):**

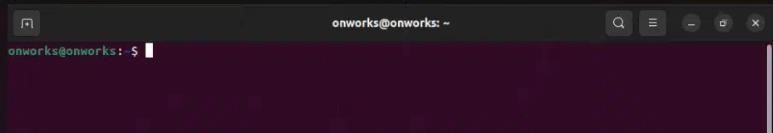
In the previous Lab we have learned to how to install **Linux** and setting up environment and some basic Linux commands. Today we will look upon some of the other basic commands.

# Linux directory structure

Linux is based on UNIX and hence it borrows its file system hierarchy from UNIX. You’ll find a similar directory structure in UNIX-like operating systems such as BSD and macOS.

## Linux Terminal

You must have used DOS commands **copy, format, dir** etc. Similarly Linux has such type of commands but with different syntax and format. You have to remember these commands because you will be using these commands frequently, and these commands will become base for your next practice. There is syntax for each command, using wrong syntax the command will not execute. Linux come with software that is called Terminal. When you run it, following screen appear with **user prompt.**

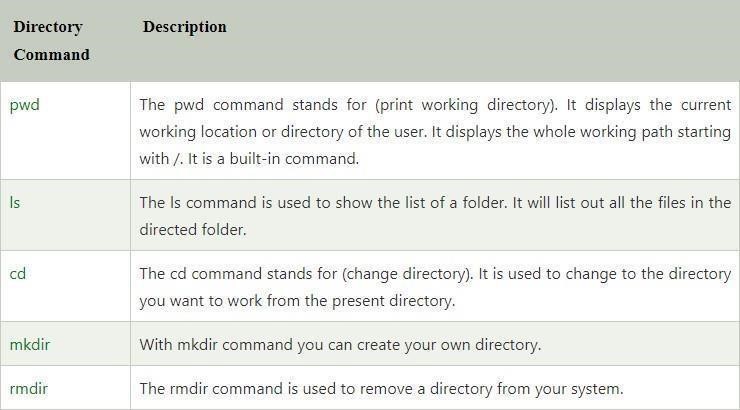


**[onworks@onworks: ~] $** is called user prompt. It consist of four elements:

1. **User Name:** Name of current login user (**onworks**)
2. **Machine Name:** Name of machine (**onworks**)
3. **Current Path**: This is path where currently user is in. (**~** shows users is currently in the **home path**)
4. **Sign: $** shows user can only perform **limited** commands. Whereas, with **#** sign when login as root user. Root user perform all commands.

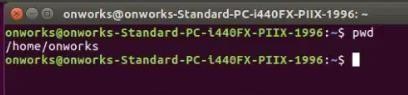
### Elementary File and Directory Commands

The elementary commonly used **directory commands** are **ls**, **cd, mkdir**, **pwd** and **rmdir**.



### Exploring the File Hierarchy

When you login, there is a special directory associated with your login name called your ***home directory***. Your home directory is actually your initial current working directory. The simplest way to find out where in the directory hierarchy your home directory is located is just to use the command **pwd** straight after you login. The **pwd** command tells you your *present working directory*.



**Note:**

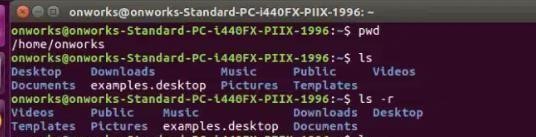
Linux commands are case-sensitive. All standard Linux commands use lower case letters only.

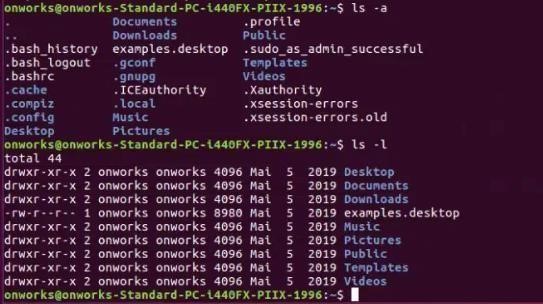
## Directory Commands

Most Linux users will use the **ls** command for listing files and folders within Linux. The **dir** command is often considered to be the Windows equivalent but it works in Linux in pretty much the same way. To get a list of all the files and folders in the current directory use the **dir** command.

## ls Command

|  |  |
| --- | --- |
| **Command** | **Description** |
| ls | List the file in the directory, just like dir command in DOS. |
| ls -a | Display all the files, and subdirectories, including hidden files. |
| ls -l | Display detailed information about each file, and directory. |
| ls -r | Display all the files in reverse order. |





1. **cd command**

In order to move your present working directory away from your home directory to somewhere else in the directory hierarchy, you use the cd (*change directory*) command. So, to change directory to the directory hierarchy’s *root* directory you would use the command.

|  |  |
| --- | --- |
| **Command** | **Description** |
| **cd** | Used to Change your current directory to another directory. |
| **cd /** | Move to root directory |
| **cd ..** | Move to one directory backward |
| **cd dir-name** | To change to any sub-directory under the current directory. |

Some standard Linux directories are given below:

**/home** Users’ home directory

**/etc** All system administrator commands, configuration files, and installation control files.

**/bin** The core set of system commands and programs. Most systems cannot boot (initially start) without executing some of the commands in this directory.

**/dev** The device files used to access system peripherals (for example, your terminal can be accessed from /dev/tty).

**/lib** The standard set of programming libraries linked with Linux programs.

**/tmp**  Temporary files created and used by many Linux programs.

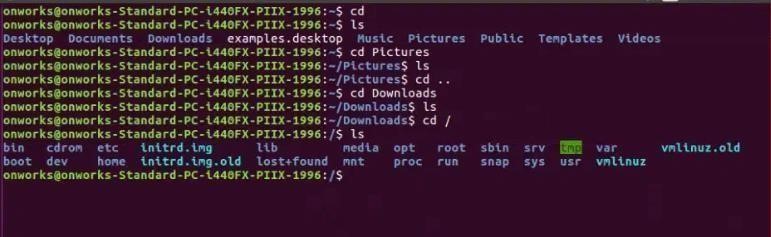
**/sbin**  The system program used to boot the system.

**/Var** Log files(Record of Web activity), spool files(temp files)

**/root**  The root user’s home directory.

|  |  |
| --- | --- |
| **/usr/bin** | Common commands and programs. |
| **/usr/doc** | Documentation |
| **/usr/games** | Games |
| **/usr/include** | Header files |
| **/usr/info** | Online documentation |
| **/usr/lib** | Shared libraries |
| **/usr/man** | Manual pages (help) |
| **/usr/sbin** | All system administration utilities |
| **/usr/share** | Shared information |
| **/usr/src** | Source code |

**Example**



1. **mkdir command**

|  |  |
| --- | --- |
| **Command** | **Description** |
| **mkdir dir-name** | Creates a new directory. **{dir-name}** specifies the name of the new directory. If the name doesn’t begin with a slash, the new directory is created as a subdirectory of the current working directory. If the name begins with a slash, the name defines the path from the root directory to the new directory. |

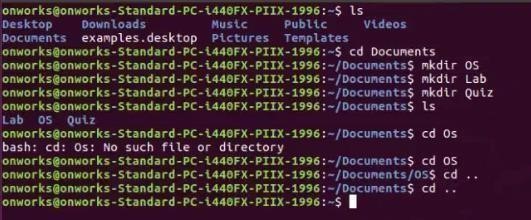
**Example 1:**



First we write **ls** (to check list of directories)

And then we write a command **mkdir Books** (which created new Directory with name Books)

### And again we use ls command to check the new created Directory



**How would you go back to the directory?**

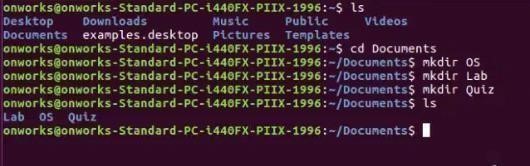
**Example 1**

After you move inside documents directory OR into sub-directory of documents. What you have to do simple type a command **cd ..**

**(There’s space between cd and ..)**   
Now you will be in your parent directory again. And the prompt becomes



**EXAMPLE 2:**

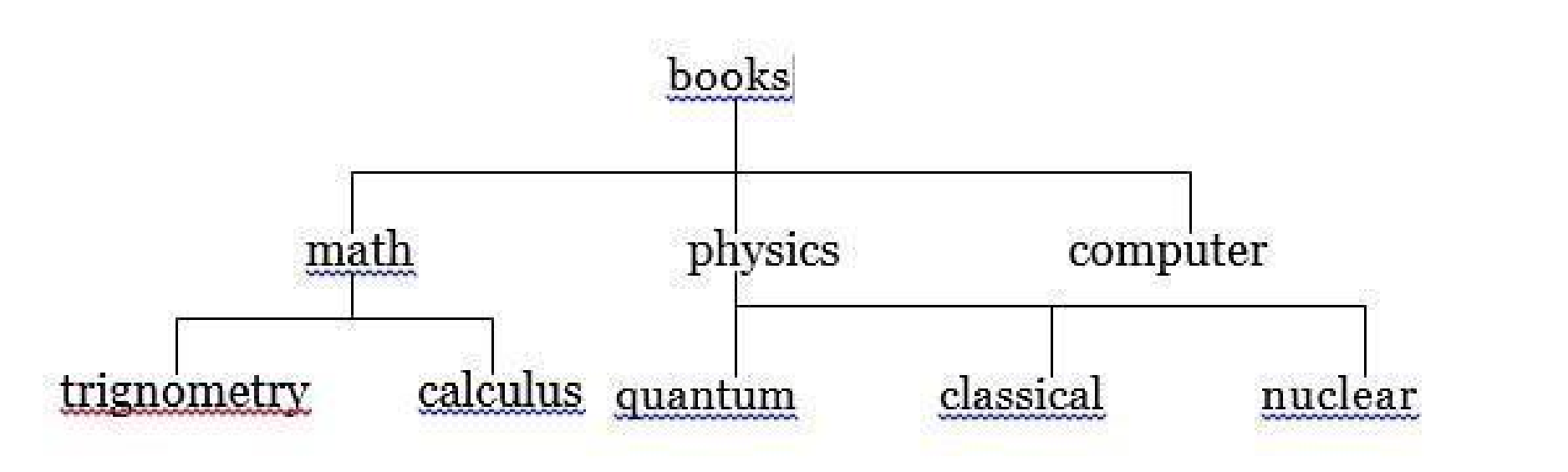


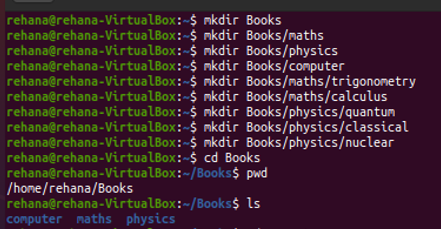
In this we **created a new directory under Document directory**. We created three new directories  **i.e. OS, Lab, Quiz**

And then we use command **ls** to check the sub-directories under Document Directory

**Example:**

Make the following directory hierarchy:





If I’m in directory classical, how would I find where I am? The command used for the purpose is **pwd.**

**Ans:** When I entered this command when I was in directory **classical** the following information was printed on my screen. The path printed was **absolute path.**

So when you lost in directory structure you could find the way to your current directory by command **pwd**.

Example:

**[user@localhost classical] $ pwd**

**/home/user/books/physics/classical**

Now you can create directories. *What is the way* ***removing*** *them?*

1. **rmdir command**

|  |  |
| --- | --- |
| Command | Description |
| rmdir dir-name    rm -r dir-name | Delete a Directory (if it is empty) rmdir will only work if the directory you are trying to remove does not contain any file. So first remove all files from the directory |
| Delete a Directory (if it is not empty) |

**Tasks:**

1. Create other directory chemistry under books, and move to chemistry directory.

Now you are quite away from your home directory. How would you go to your home directory directly? Your current location is:

books/chemistry

Go to Home directory directly.

1. Now you are in your home directory.

How will you go to chemistry directory directly?

What do you think **books/chemistry** is relative or absolute path?

1. How will you add a directory **graphics** under the directory computer while you are in **physics**’ subdirectory **classical**?
2. You are in books directory, from here try to remove sub-directory quantum under the directory physics. Now move to directory computer, from here remove sub-directory calculus under the directory math.
3. Which are the Linux Directory Commands? Explain the understanding of commands in your own words.