what is shell?

A shell is a type of computer program called a command-line interpreter that let a Linux and Unix users control their operating systems with command line interfaces.

shells allow users to communicate efficiently and directly with their operating systems.

basic shell commands?

- $ls - lists all the files
- $pwd - gives present working directory.
- $cd - change directory
- $man - gives information about command.
- $exit - exits from the shell
- which - shows full path of command

shell built in commands?

Built in commands are contained with in the shell itself, means shell executes the command directly, without creating a new process.

Built in commands:

- break
- cd
- exit
- pwd
- export
- return
- alias
- echo
- print
- logout
- help
- man

Working with directories?

This module is brief overview of the most common to work with directories .

Pwd

Cd

Ls

mkdir and **rmdir** these commands are available on any Linux system.

This module also discuss **absolute** and **relative paths** and **path completion** in the **bash** shell.

Pwd :

The you are here sign in can be displayed with the **Pwd** command (present working directory).

Ex**: $ pwd**

Cd :

Cd means change directory

You can change your current directory with the cd command.

Ex: **$cd /etc**

Cd ~ :

This cd also a shortcut to get back your home directory. Just typing a target directory, will put you in home directory. Typing cd~ has the same effect.

Ex: **cd~**

Cd .. :

To go the parent directory(the one just above your current directory in the directory tree), type cd..

Ex: **$ cd ..**

```
$ pwd
```

Cd - :

Another useful shortcut of cd is to just type cd – to go the previous directory.

Ex: **$ cd –**

```
$ pwd
```

Absolute and relative paths:

You should be aware of **absolute and relative paths** in the file tree. When you type a path starting with a **slash (/),** then the root of the file tree is assumed. If you don’t start your path with a slash, then the current directory is the assumed starting point.

The screenshot below first shows the current directory **/home/paul**. From within this directory you have to type cd/home instead of cd home to go the /home directory.

Paul: this is an username(example)

```
paul@debian8$ pwd
/home/paul
```

```
paul@debian8$  cd home
bash:  cd:  home: no such file or directory
paul@debian8$ cd /home
paul@debian8$ pwd
/home
```

When inside **/home**, you have type **cd paul** instead of **cd /paul** to enter the subdirectory **paul** of the current directory /**home.**

```
paul@debian8$ pwd
/home
paul@debian8$  cd /paul
bash:  cd:  home: no such file or directory
paul@debian8$ cd paul
paul@debian8$ pwd
/home/paul
```

In case of your current directory is the **root directory** / , then both **cd /home** and **cd home** will get you in the **/home** directory.

```
paul@debian8$ pwd
/
paul@debian8$  cd home
paul@debian8$  pwd
/home
paul@debian8$ cd /
paul@debian8$ cd /homess
paul@debian8$ pwd
/home
```

This was the last screenshot with **pwd** statements. From now on the current directory will often be displayed in the prompt. Later in this book we will explain how we shell variable $**ps1** can configured to show this