Kavyashree . N
MCA 'A' section

PROCESS MANAGEMENT

A process means program in execution. It generally takes an input, process it and gives the appropriate output. There are 2 types of processes

1. Foreground processes:

These are the processes which are to be executed or initiated by the user or programmer, they cannot be initialized by system services. Such processes take the input from the user and return the output. While these processes are running we cannot directly initiate a new process from the same terminal.

2. Background process:

These are the processes to be executed or initiated by the system itself or by users, though they can even be managed by users. These processes have a unique PID or process if assigned to them and we can initiate other processes within the same terminal from which they are initiated

Example for foreground processor

1. sleep 5

This command will be executed in the terminal and we would be able to execute another command after the execution of the above command

2. To stop a process in between the execution, the user has to press Ctrl + Z to force stop the process

```
sleep 100

^Z
[1]+ stopped          sleep 100
```

3. To get the list of jobs which are running or stopped

```
Jobs

[1]+ stopped          sleep 100
```

4. To run all the pending and force stopped jobs in the background.

```
bg
Jobs
[1]+ stopped          sleep 100

bg
[1]+ sleep100 &

Jobs
[1]+ running          sleep 100 &
```

5 To get details of a process running in background

```
ps -et | grep sleep
```

6. To run all the pending and force stopped jobs in the foreground.

```
fg
```

User Management.

A user is an entity that can manipulate files and perform several other operations. Each user is assigned a ID which is unique for each user in the operating system.

1. To list out all the users in Linux, use awk command with -F option. Here we are accessing a file and printing only first column with the help of print $1 and awk

```
awk -F':' '{print $1}' /etc/passwd
```

2. Using id command, you can get the ID of any username. Every user has an id assigned to it and the user is identified with the help of this id. By default, this id is also the group id of the user

```
id username
```

3. The command to add a user. useradd command adds new user to the directory. The user is given the ID automatically. The username of the user will be as provided by us in the command

```
sudo ~~add~~ useradd username
```

4. Using passwd command to assign a password to a user. After using this command we have to enter the new password for the user and then the password gets updated to the new password

```
passwd username.
```

* Group creation.

syntax : group add group-name

eg. groupadd Group 1

* setting password to group

```
gpasswd Group 1
New password
Re-enter password
```

* To add a user to an existing group

```
usermod -G group-name username
```

* To delete a user from a group

```
gpasswd -d username groupname
```

* To delete the whole group

```
groupdel groupname
```

FILE PERMISSIONS

Linux have three types of permissions defined.

1. Read (r) : The read permission allows you to open and read the content of a file. But you can't do any editing or modification in the file

2 Write (w): The write permission allows you to edit, remove or rename a file. For instance, if a file is present in a directory, and write permission is set on the file but not on the directory, then you can edit the content of the file but can't remove, or ~~ren~~ rename it.

3. Execute (x) : In Unix type system, you can't run or execute a program unless execute permission is set . But in Windows there is no such permission available

* To add permissions

```
chmod + rwx filename
```

* To remove permissions

```
Chmod - rwx directory name
```

* To allow executable file permissions

```
Chmod +x filename
```

* To take out write and executable permissions

```
chmod -wx filename.
```

Absolute method

Octal	Binary	File mode
0	0 0 0	- - -
1	0 0 1	- - x
2	0 1 0	- w -
3	0 1 1	- w x
4	1 0 0	r - -
5	1 0 1	r - x
6	1 1 0	r w -
7	1 1 1	r w x