Going around your machine and commands – Part 2
30-October-2019

Before you install any package, first get the latest description of packages from the source using the following command.

```
sudo apt-get update
```

After the package database is updated, you can upgrade all the packages you installed to the latest version using the following command.

```
sudo apt-get upgrade
```

You can now install packages that you need usin the following command. Example shows how to install a package called fortune. The command “fortune” provided by this package is funny. Run it and see for yourself.

```
sudo apt-get install fortune
```

The command “/usr/bin/info” shows the documentation about Linux in a structured manner. You can use the arrow keys to navigate to each section, press enter to browse. Use “<” to go back and “q” to quit. The experience of navigation using “info” command is like a text-based browser.

Linux does not pay attention to the file extensions. The characters that come after the dot have no special meaning. You can rename a file from “something.jpg” to “something.png” and the system will not bother. Use the command “/usr/bin/file something.jpg” to see what the system says. Change the file name using “mv something.jpg something.png” and run the command “/usr/bin/file something.png” and see that the system reports the file to be of the same type. At the beginning of the file, there are few magic characters that are read by the tool “/usr/bin/file” to determine what type of a file it is. The name does not matter.

Use this utility “/usr/bin/file” on files in the directories “/usr/bin/” or “/bin” to see which of them are ELF executables and which are shell scripts. If a file is a shell script, you can use “cat” or “less” or “more” to view the contents of that command. You would notice that all shell scripts come in a particular format where the first line starts with “#!/bin/sh” which tells the system that the script should be run using the shell “/bin/sh”. Lines that start with “#” are comments for documentation purpose.

Use the command “/usr/bin/cal” to view the calendar of the current month. Use it with month and year as arguments to see the calendar of any month or year – either in past or future. Eg., use “cal Aug 1947” to see the calendar of the August month of the year 1974. Check out which day of the week you were born using this command.

Use the command “/usr/bin/ncal” to display the calendar with the week days in vertical stacking as opposed to horizontal stacking used in the output of the command “cal”. Rest of the features are same.

Use the command “date” to see the current time and date. You can customize the format being displayed too. See the following example:

```
date +”%A %d %B %Y”
```

See for yourself how the output looks like.

Homework:

1. Write a command that displays the weekday of your friends birthday in full form as the output. Eg., if your friend was born on a Friday, the output of the command should be Friday.
2. Create images using the tool “gimp” and export to different formats like png, tiff, jpg, gif and bmp. Use the “file” command to see what the system knows about these formats.
3. Create a command that lists all the shell scripts in the directory “/bin”.