1. How to jump to a particular line in a file using vim editor?

Syntax: -

$ vim +<line number> <filename>

Suppose we want to jump to 10th line in ineuron.txt file

**$ vim +10 ineuron.txt**

1. How do you sort the entries in a text file in ascending order?

We can use the **sort**command.

**$ sort sample.txt**

1. What is the export command used for?

The **export** command is used to set and reload the environment variables. For example, if we want to set the Java path, then the command would be:

**$ export JAVA\_HOME = /home/user/Java/bin**

1. How do you check if a particular service is running?

**$ service <servicename> status**

1. How do you check the status of all the services?

**$ service --status-all**

1. How do you start and stop a service?

To start the service:

**$ service <servicename> start**

To stop the service:

**$ service <servicename> stop**

1. Explain the free command.

This command is used to display the free, used, swap memory available in the system.

**$ free**

1. Explain chmod command

You can change the permission of a file or a directory using the chmod command. There are two modes of using the **chmod** command:

1. **Symbolic mode**
2. **Absolute mode**

#### 1. Symbolic mode

The general syntax to change permission using Symbolic mode is as follows:

$ chmod <target>(+/-/=)<permission> <filename>

where <permissions> can be **r: read; w: write; x: execute**.

<target> can be **u : user; g: group; o: other; a: all**

'+' is used for adding permission

'-' is used for removing permission

'=' is used for setting the permission

For example, if you want to set the permission such that the user can read, write, and execute it and members of your group can read and execute it, and others may only read it.

Then the command for this will be:

**$ chmod u=rwx, g=rx, o=r filename**

#### Absolute mode

The general syntax to change permission using Absolute mode is as follows:

$ chmod <permission> filename

The Absolute mode follows octal representation. The leftmost digit is for the user, the middle digit is for the user group and the rightmost digit is for all.

|  |  |  |
| --- | --- | --- |
| 1. | Execute permission | – – x |
| 2. | Write permission | – w – |
| 3. | Execute and write permission: 1 (execute) + 2 (write) = 3 | – wx |
| 4. | Read permission | r – – |
| 5. | Read and execute permission: 4 (read) + 1 (execute) = 5 | r – x |
| 6. | Read and write permission: 4 (read) + 2 (write) = 6 | rw – |
| 7. | All permissions: 4 (read) + 2 (write) + 1 (execute) = 7 | rwx |

For example, if we want to set the permission such that the user can read, write, and execute it and members of your group can read and execute it, and others may only read it.

Then the command for this will be:

**$ chmod 754 filename**

1. Explain grep command

Grep stands for **Global Regular Expression Print.** The grep command is used to search for a text in a file by pattern matching based on regular expression.

**Syntax**: grep [options] pattern [files]

**Example**:

**$ grep -c "ineuron" sample.txt**

This command will print the count of the word “ineuron” in the “**sample.txt**” file.

1. What is the use of nano editor?

[Nano](apt:nano) is a simple terminal-based text editor. Though not as powerful as Emacs or Vim, it is easy to learn and use. Nano is ideal for making small changes to existing configuration files or for writing short plain text files.

It has an easy GUI(Graphical User Interface) which allows users to interact directly with the text in spite of switching between the modes as in vim editor.