hitbullseye

Difference between vi and vim Editors

Visual text editors include Vi. Line editors, which used to work on a single line of text at a time, predated visual editors. To utilize a line editor, you must instruct it to navigate to a specific line and specify the modification you wish to make, such as text addition or deletion.

The introduction of video terminals made visual editing feasible. Vi was developed to enable video editing with a moving cursor. The term "visual" is where the name Vi comes from. Bill Joy created the original version of vi in 1976.

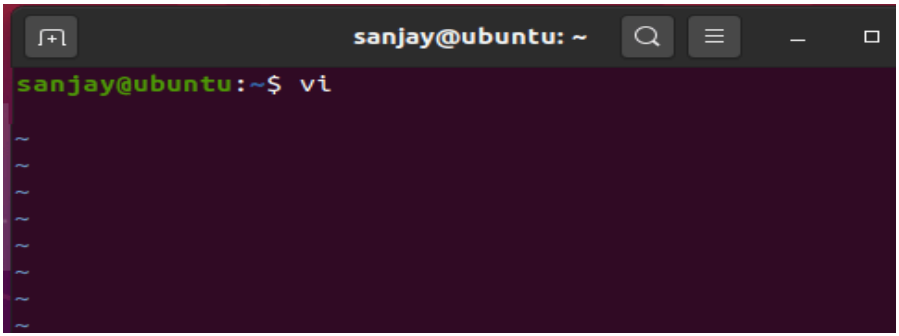
Line editors, instead of giving a complete overview of a text file, used to show only the current line that the user was working with. Vi was the first editor that worked in a mode where the complete text file was displayed which made it possible to move back and forward between lines.

Because of this, in the text editors' historical context, Vi was a big achievement that prepared a base for several modern visual text editors. One such a modern text editor is Vim. Vim is purely based on Vi. Vim stands for **Vi-Improved**, the user-friendly version of Vi.

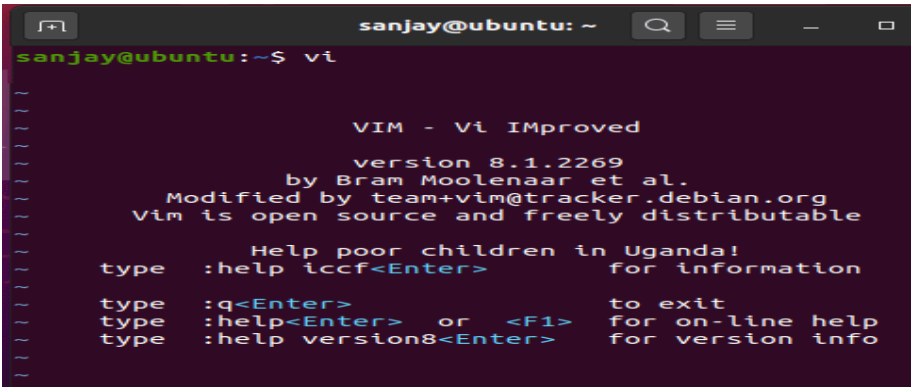
Vim is written by **Bram Moolenaar**. It is a substantial improvement over traditional Vi. Since Vi is quite difficult, most Linux distributions don't include real Vi; rather, they

include Vim and symbolically link (or alias) the Vim to the name Vi.

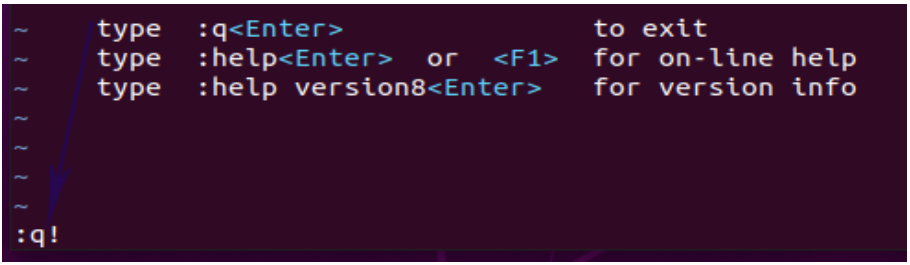
If the output of this command displays a bunch of tildes (~) marking empty lines and a line of status at the bottom of the screen, the traditional version of Vi is installed. The following image shows the output of this command when the traditional version of Vi is installed.



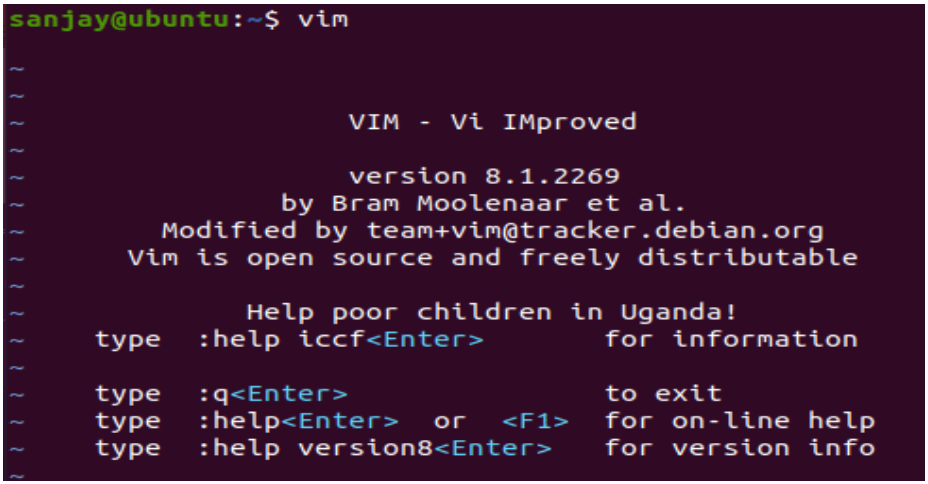
If the improved version of Vi is installed and linked with the traditional '**vi**' command, the output of this command will display information about it. The following image shows the output of this command when the improved version of Vi is installed.



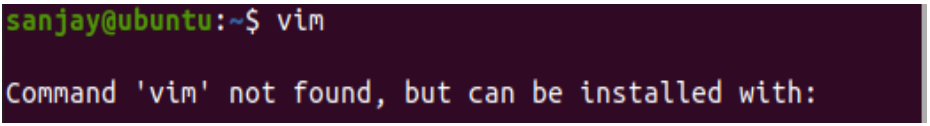
To quit from the output, press **Esc** key and type **:q!** and hit the **Enter** key.



Another simple way to know whether the Vim is installed or not is running the **vim** command. If command runs successfully, Vim is installed. If it returns the **'command not found'** error message, Vim is not installed. The following image shows the output of **vim** command when Vim is installed.



The following image shows the output of **vim** command when Vim is not installed.



If Vim is not installed, you can install it just like any other Linux package. For example, to install Vim on Ubuntu, use the following command.

```
$sudo apt install vim
```

The following image shows the sample output of the above command.

```
sanjay@ubuntu:~$ sudo apt install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  ctags vim-doc vim-scripts
The following NEW packages will be installed:
  vim
0 upgraded, 1 newly installed, 0 to remove and 247 not up
graded.
Need to get 1,238 kB of archives.
After this operation, 3,111 kB of additional disk space w
ill be used.
update-alternatives: using /usr/bin/vim.basic to provide
/usr/bin/vi (vi) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide
/usr/bin/view (view) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide
/usr/bin/ex (ex) in auto mode
sanjay@ubuntu:
```

Vi editor V/s Vim editor

As far as functionality is concerned, both editors work in the same manner. Which editor you choose is a matter of personal choice. If you are learning from scratch, you should learn the Vim editor instead of the Vi editor. Due to added features, learning and using Vim editor is much easier than the Vi editor. Since Vim is based on the Vi, when you will learn how to use the Vim editor, you will automatically learn how to use the Vi editor.

Differences between Vi and Vim editors

Vim adds the following features and functions to the existing functionality of Vi.

- It includes more features for the programming languages such as; syntax highlighting, code folding, text formatting, etc.
- It includes an inbuilt utility for comparing files.
- It includes the undo/redo facility.
- It supports external scripting languages.
- It can edit compressed files.
- It can edit remote files over the network protocol.
- It supports plugins for additional functionality.
- It supports screen splitting for editing multiple files simultaneously.

Vi - an editor of administrators

Administrators learn Vi mainly for the following reasons.

- Vi is the universal editor of Linux. No matter which Linux distribution you use, or which version of the distribution you use, it contains the Vi editor. Therefore, if an administrator knows how to use the Vi editor, he can edit text files on any Linux system.
- Vi is always available in every mode of Linux. In many situations, such as emergency mode and a remote session, Vi is the only available editor. If an administrator does not know how to use Vi, then

editing configuration files in such mode would be a nightmare for him.

- Vi is the lightest and fastest editor. Vim has a small footprint in RAM and on the CPU. For many tasks, Vi is easier to use than to load and use a heavy graphical text editor. Vi is designed for typing speed. A skilled vi administrator never has to lift his or her fingers from the keyboard while editing.

Some system administration commands are built on Vi. For example, **edquota** (a command that limits available disk space for users on your server), **visudo** (a command that sets permissions for the sudo command), and **crontab –e** (a command that schedules a task to run at a given moment in time). These commands are macros built on Vi. Learning the Vi helps to use and manage these commands more effectively.