Basic Vim Commands

VI Editor
Vi Editor is a widely used text editor in Unix/Linux systems and is known for its efficiency and flexibility.
1. Create/Edit a file
To create/edit a file in the Vim editor run the below command
Syntax:
vim filename
Example:
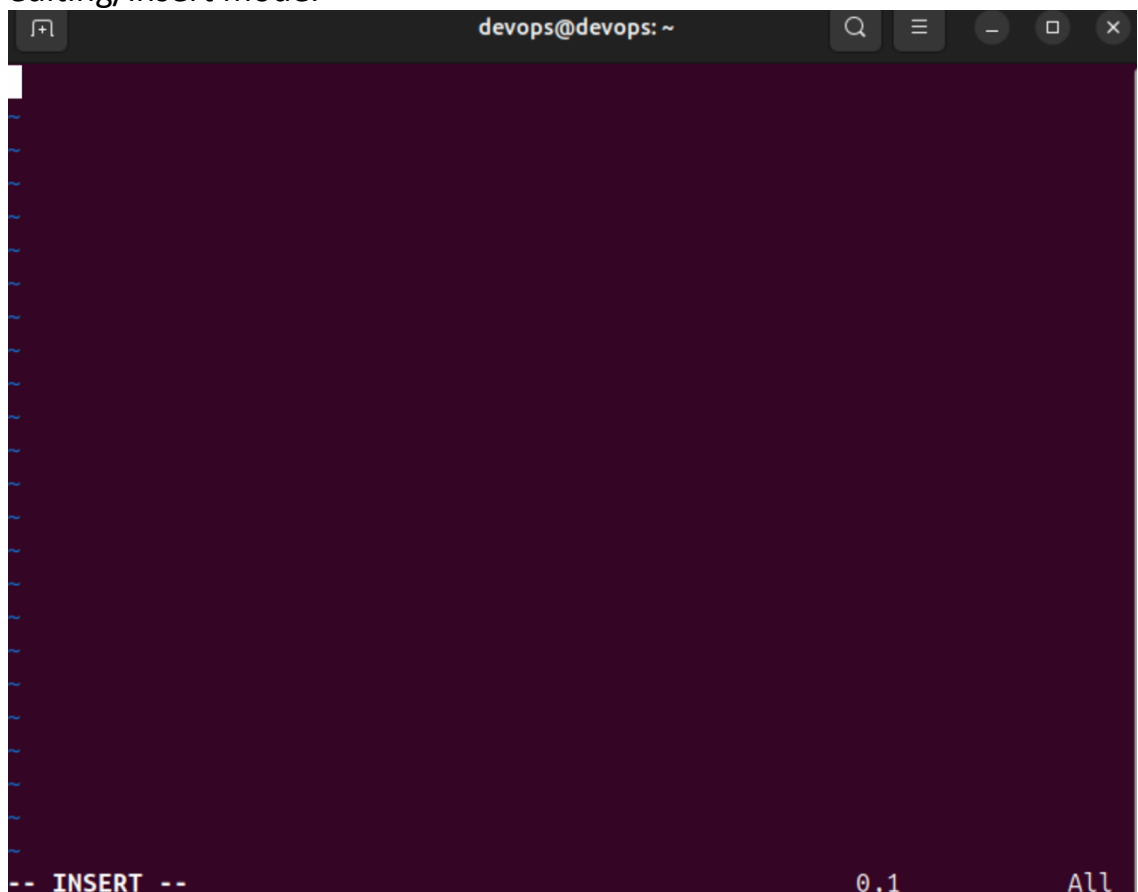vim new.txt
Replace [**filename**] with the name of the file you want to create or edit. In this case, the filename is **new.txt**

2. Changing modes in the Vim editor
There are Six modes in Vim editor.
- **Editing/Insert Mode:** To make any changes in a file in Vim editor user has to first enter editing/insert mode, Press **'i' or 'a'** to enter editing/insert mode.

Entering in Insert Mode
- **Command Mode:** To run any command in Vim editor user has to enter command mode, if you are in editing/insert mode

Press **Esc** then '**:**' followed by the command. **For example:** set the number
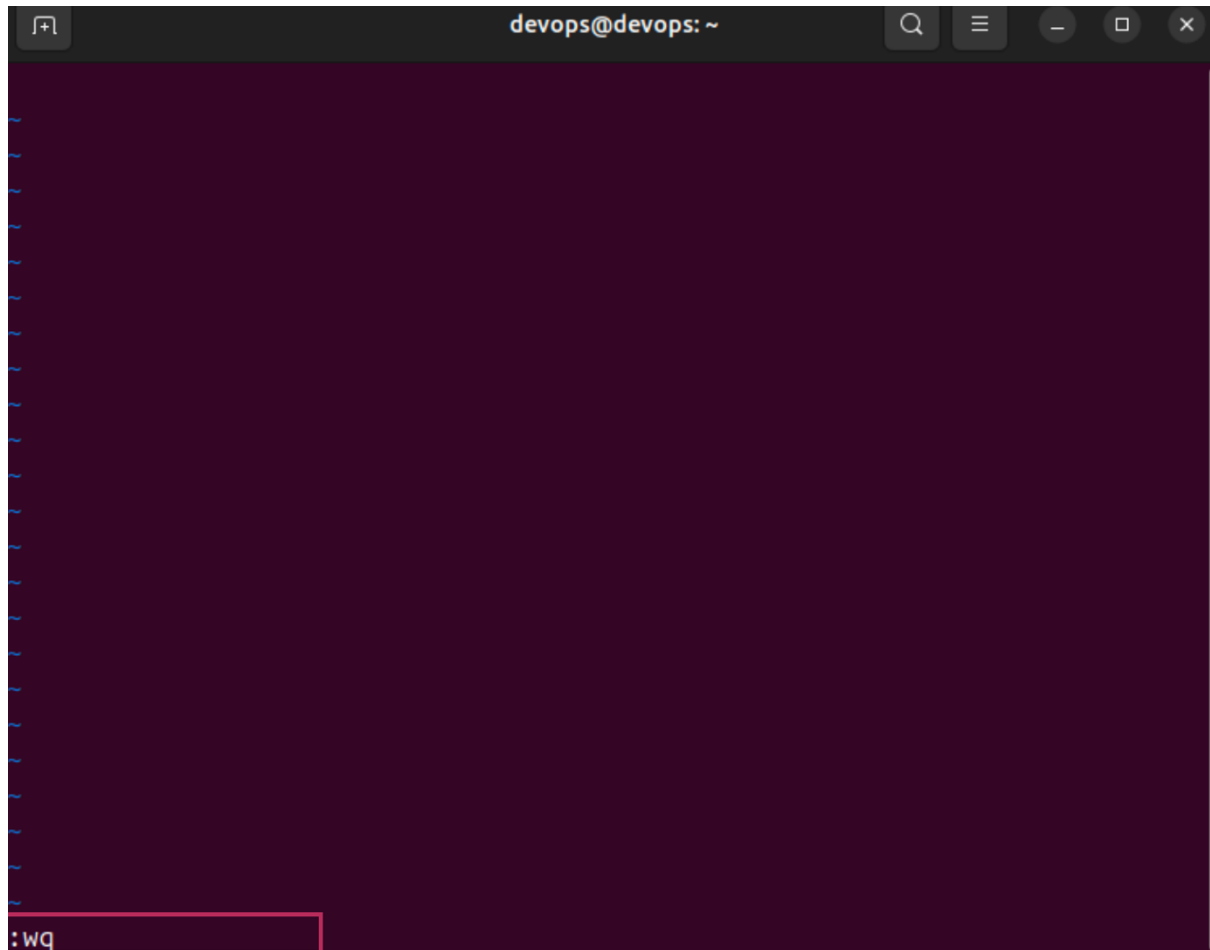
3. Quit the Vim editor
To quit the Vim editor enter command mode and use the following command,

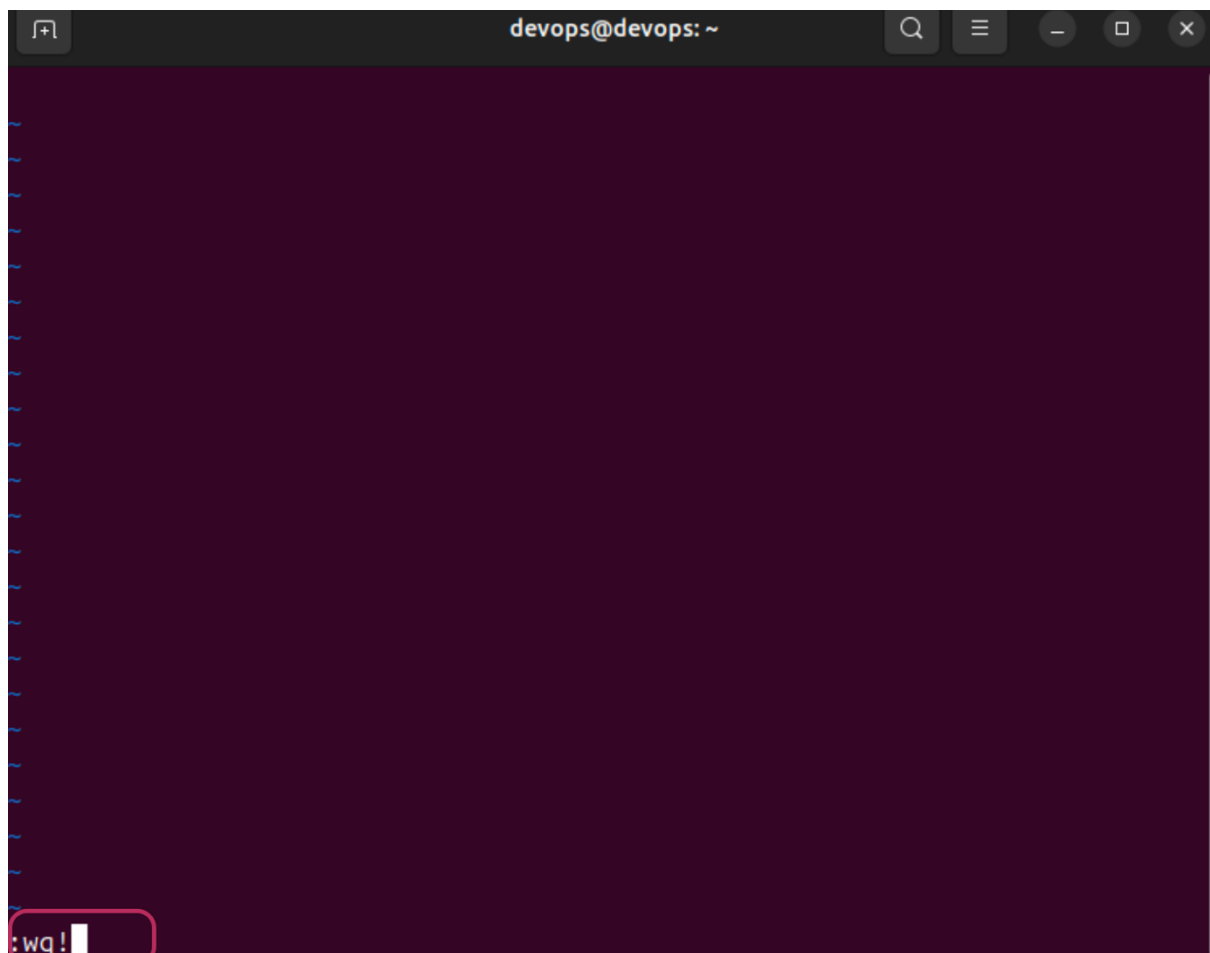| Command	Description
:q	Quit the editor
:q!	Quit without saving changes i.e. discard changes.
:wq	Save the changes and quit the editor

Example/Screenshot:
Quitting Vim Editor
4. Save the changes
To save the changes to the file in the Vim editor enter command mode and use the following command,

Command	Description
:w	Write to file called [file_name] (save as).
:w!	Overwrite to file called [file_name] (save as forcefully).
:wq	Save the changes and quit the editor

5. Navigation in Vim editor

We have covered the navigation in Vim editor in a detailed article at GFG which you can check here.

Command	Description
k	Moves the cursor up one line.
j	Moves the cursor down one line.

h	Moves the cursor to the left one-character position.
l	Moves the cursor to the right one-character position.

6. Delete a line

To delete a single line Press the **Esc** key if you are in insert/editing mode, go to the file you want to delete Press '**dd**' and then the line gets deleted.
To delete a range of lines use the following command,
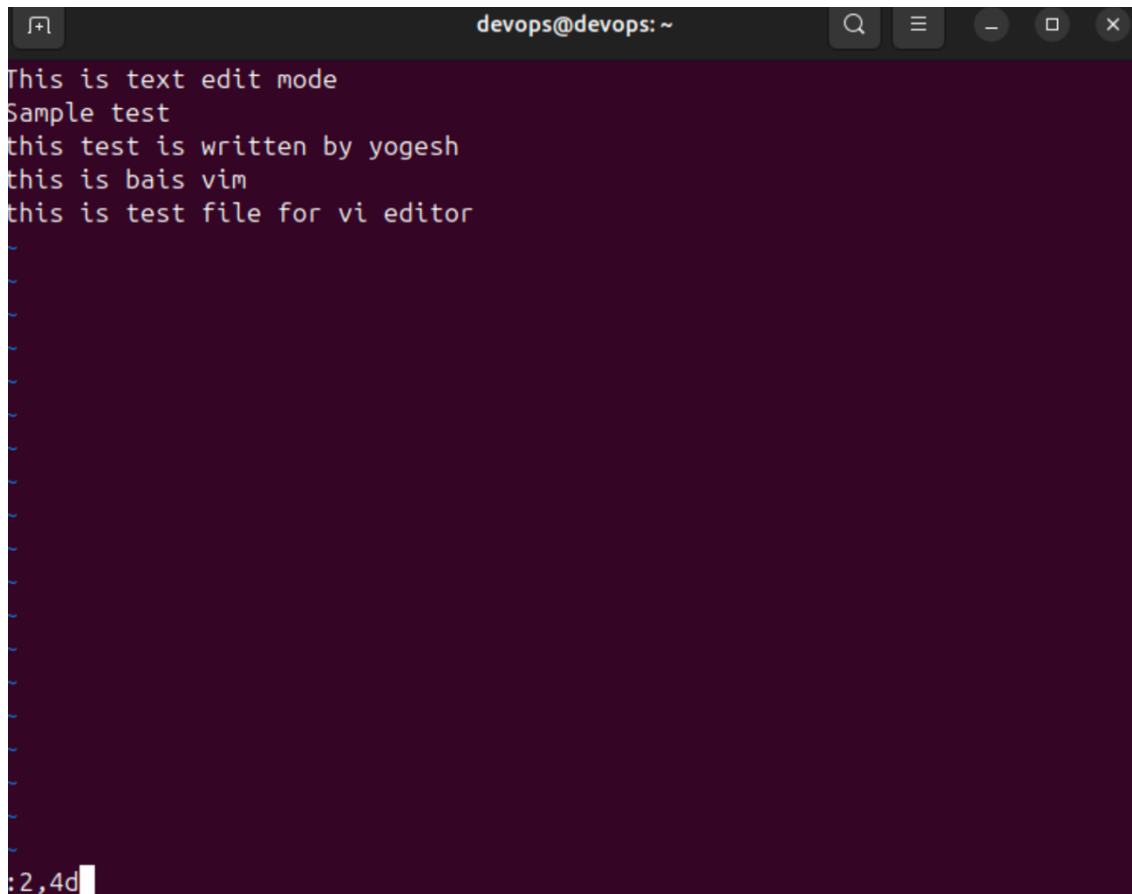
Syntax:
:[start],[end]d

Example::2,4d In this command the editor will delete the lines from **3 to 5** including the extremes.

Example Screenshot:

```
This is text edit mode
Sample test
this test is written by yogesh
this is bais vim
this is test file for vi editor
~
~
~
~
~
~
~
~
~
~
~
~
:
```

```
This is text edit mode
Sample test
this test is written by yogesh
this is bais vim
this is test file for vi editor

:2,4d
```
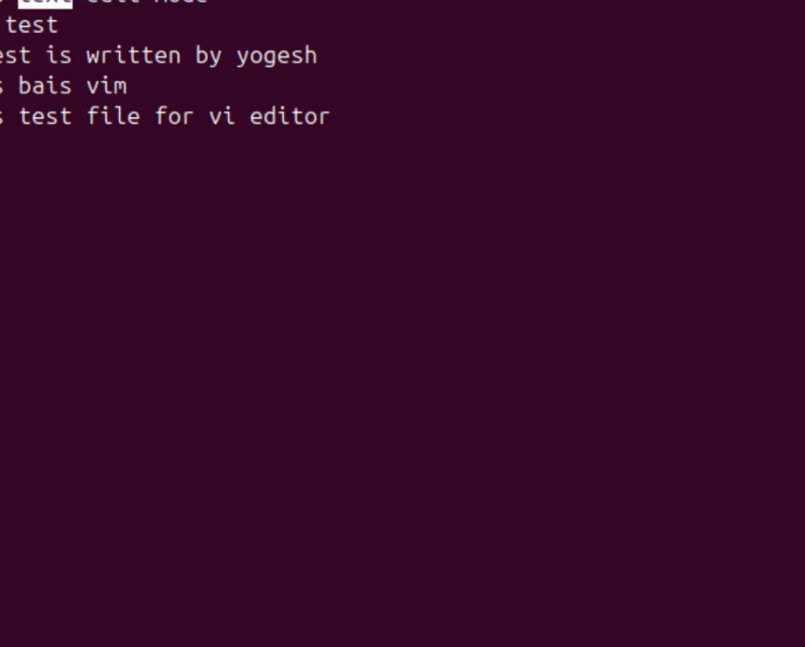
Deleting a Line

7. Search and Replace
To find and replace words in the vim editor we use **substitute** or **:s** command syntax of the command is as follows:
:[range]s/{pattern}/{string}/[flags] [count]
The command searches the pattern in [range] lines and replaces
the **[pattern]** with **[string]**. If **[range]** is not mentioned then the command will replace the words in the current line with **[string]** only.
Example:

```
This is text edit mode
Sample test
this test is written by yogesh
this is bais vim
this is test file for vi editor
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:s/text/test
```

```
This is text edit mode
Sample test
this test is written by yogesh
this is bais vim
this is test file for vi editor
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:%s/test/were/g
```

8. Display Line Numbers

To display **absolute line numbers** use any of the following commands,
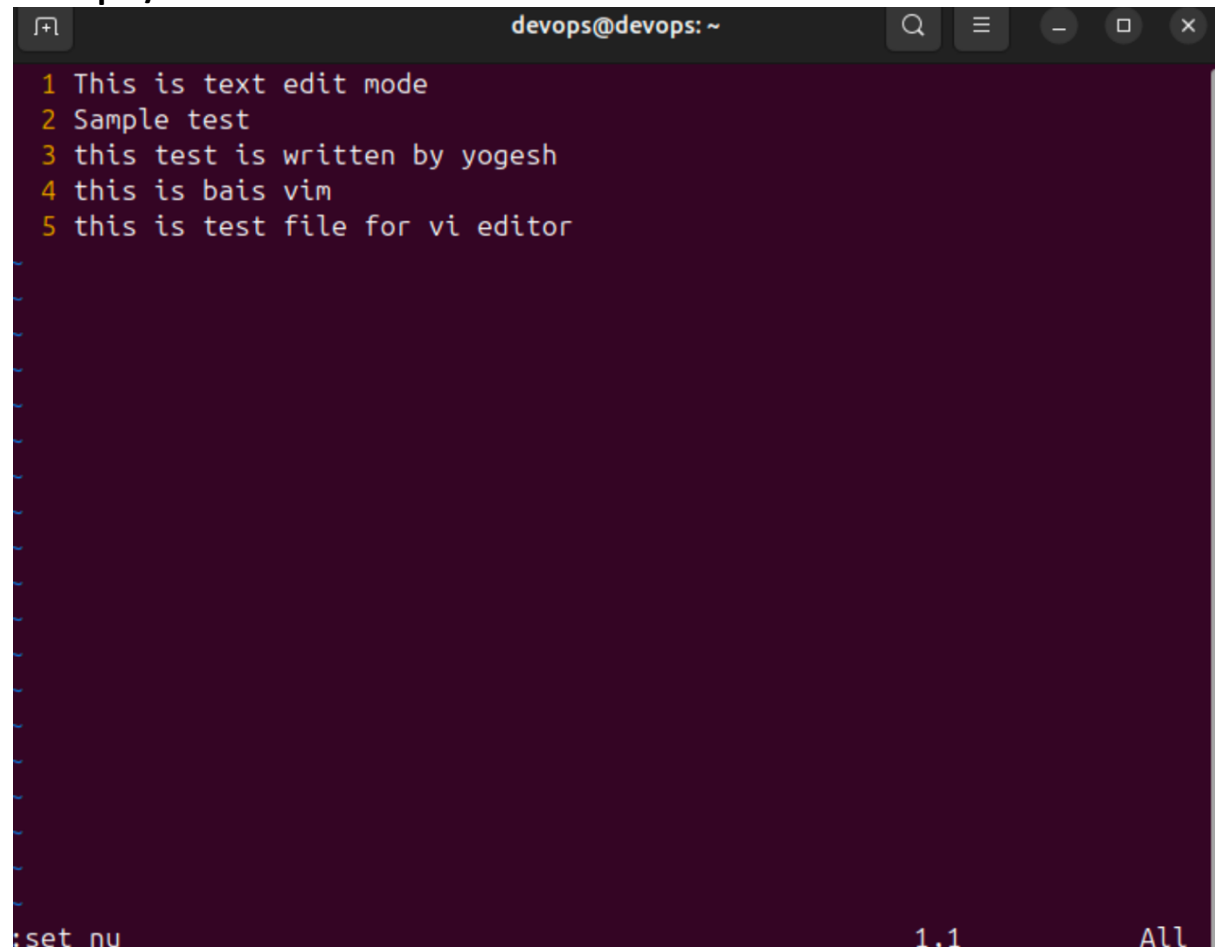
Syntax:

:set number

or

:set nu

Example/Screenshot:

To display **relative line numbers**: In relative line numbers the current line is shown as 0 and the lines above and below are incremented by 1,
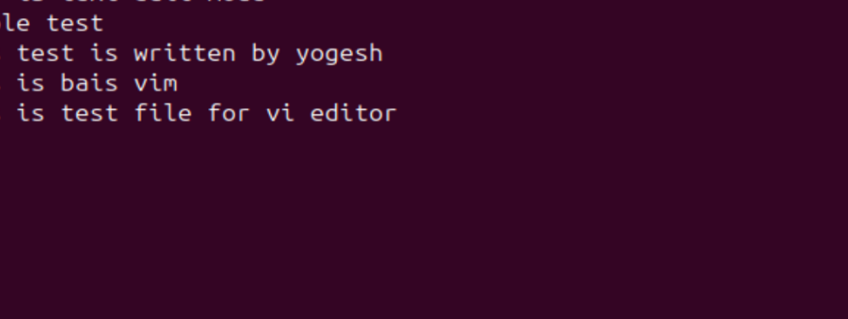
Syntax:

:set relativenumber

or

:set rnu

Example/Screenshot:

```
 devops@devops: ~                    Q  ≡   —   □   ×

3 This is text edit mode
2 Sample test
1 this test is written by yogesh
0 this is bais vim
1 this is test file for vi editor












:set rnu                          4,1              All
```

9. Hide Line numbers

To hide absolute line numbers use any of the following commands,

Syntax:

:set nonumber
or
:set nonu

Syntax:

:set norelativenumber
Alternatively,
:set nornu