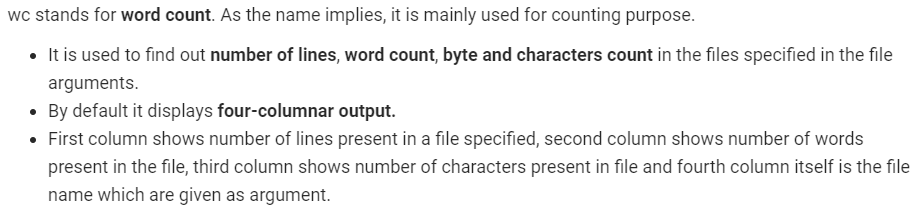
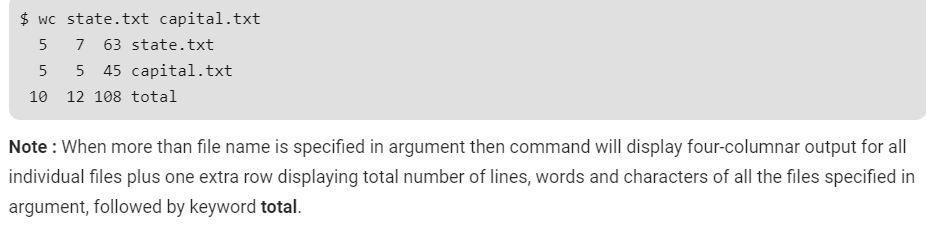
**wc command:**

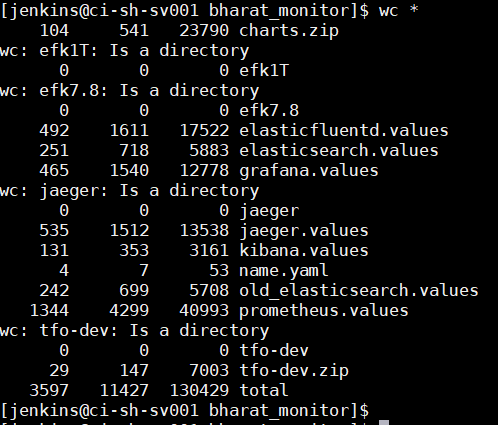


**Examples:**

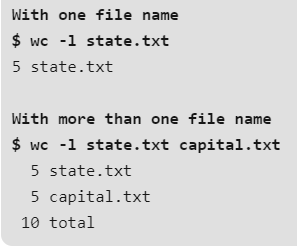


**Passing more than one file in argument:**

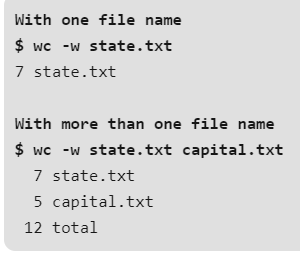




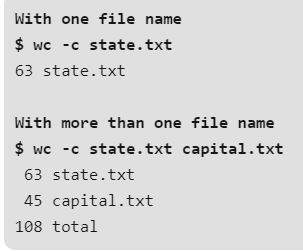
**-l:** This option prints the **number of lines** present in a file. With this option wc command displays two-columnar output, 1st column shows number of lines present in a file and 2nd itself represent the file name.



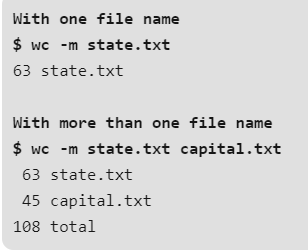
**-w:** This option prints the **number of words** present in a file. With this option wc command displays two-columnar output, 1st column shows number of words present in a file and 2nd is the file name.



**-c:** This option displays **count of bytes** present in a file. With this option it displays two-columnar output, 1st column shows number of bytes present in a file and 2nd is the file name.

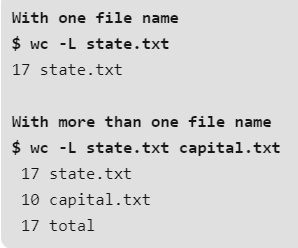


**-m:** Using -m option ‘wc’ command displays **count of characters** from a file.



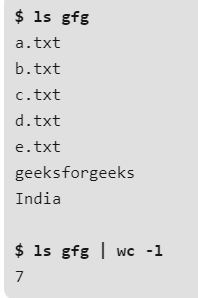
**-L:** The ‘wc’ command allow an argument -L, it can be used to print out the length of longest (number of characters) line in a file. So, we have the longest character line Arunachal Pradesh in a file state.txt and Hyderabad in the file capital.txt. But with this option if more than one file name is specified then the last row i.e. the extra row, doesn’t display total but it displays the maximum of all values displaying in the first column of individual files.

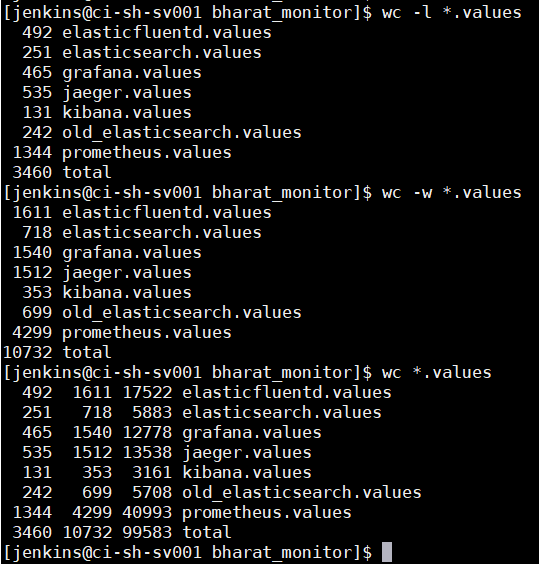
Note: A character is the smallest unit of information that includes space, tab and newline.



**To count all files and folders present in directory:**

As we all know ls command in unix is used to display all the files and folders present in the directory, when it is piped with wc command with -l option it display count of all files and folders present in current directory.



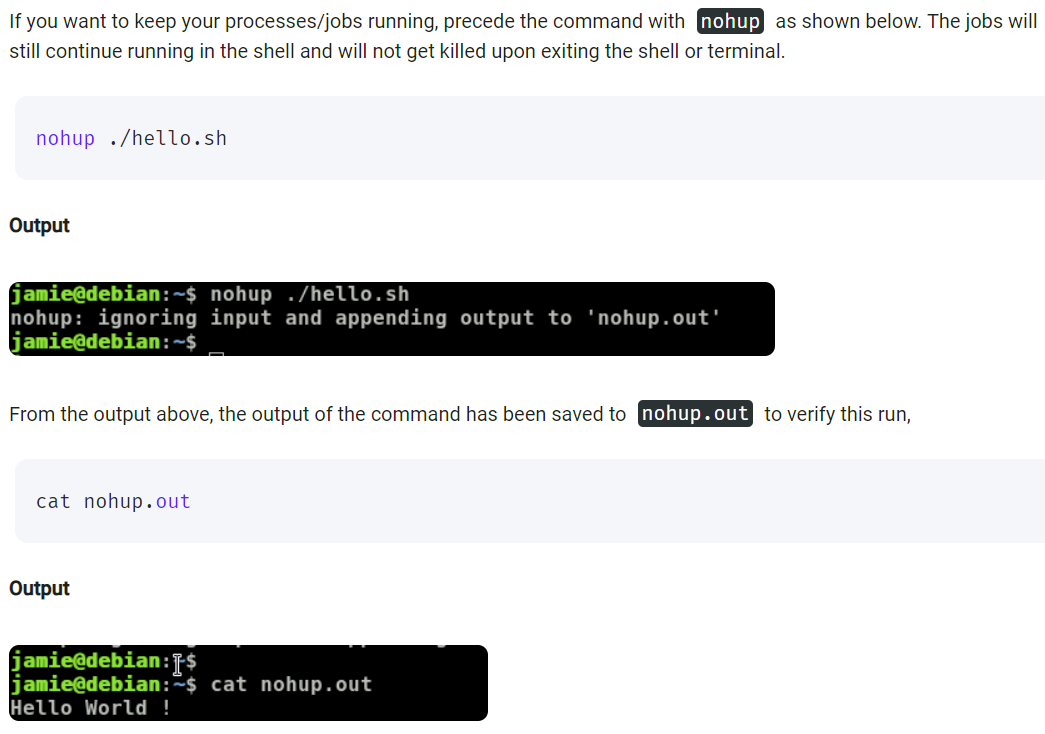


**nohup:**

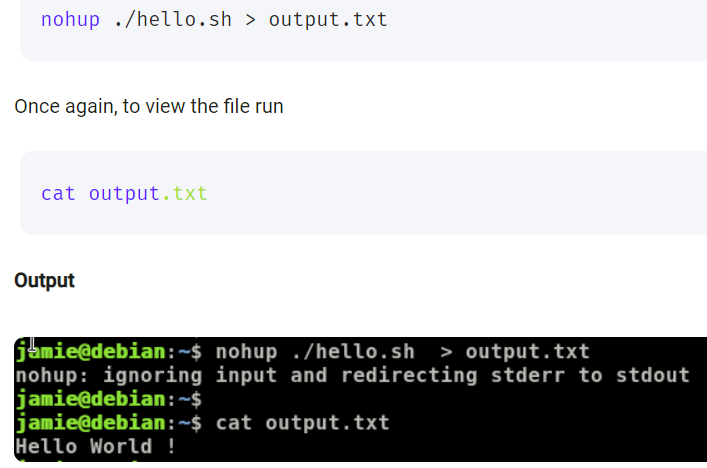
* Nohup, short for no hang up is a command in Linux systems that keep processes running even after exiting the shell or terminal.
* The meaning of nohup is ‘no hangup‘. Normally, when we log out from the system then all the running programs or processes are hangup or terminated. If you want to run any program after log out or exit from Linux operating system then you have to use nohup command. There are many programs that require many hours to complete. We don’t need to log in for long times to complete the task of the command. We can keep these type of programs running in the background by using nohup command and check the output later. Some example of using nohup command are memory check, server restart, synchronization etc. How you can use a nohup command on Ubuntu to run a program in the background is shown in this tutorial.

**Examples:**

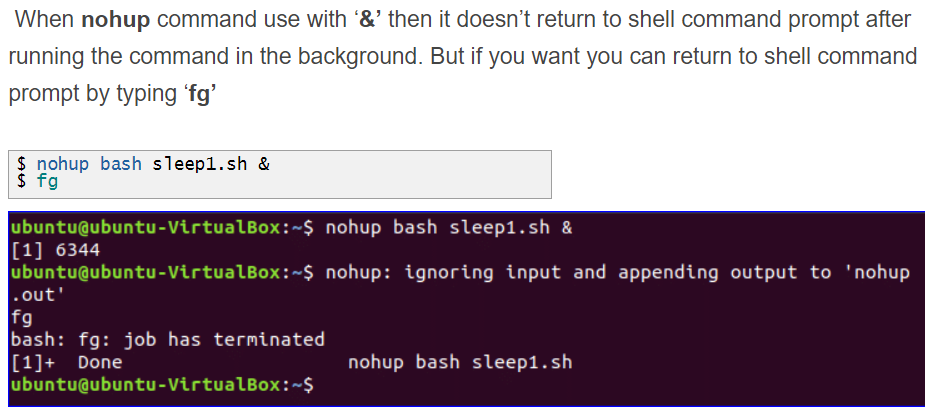
* When you run nohup command without ‘&’ then it returns to shell command prompt immediately after running that particular command in the background. In the following example, nohup run bash command without ‘&’ to execute sleep1.sh file in the background. The output of the nohup command will write in nohup.out the file if any redirecting filename is not mentioned in nohup command

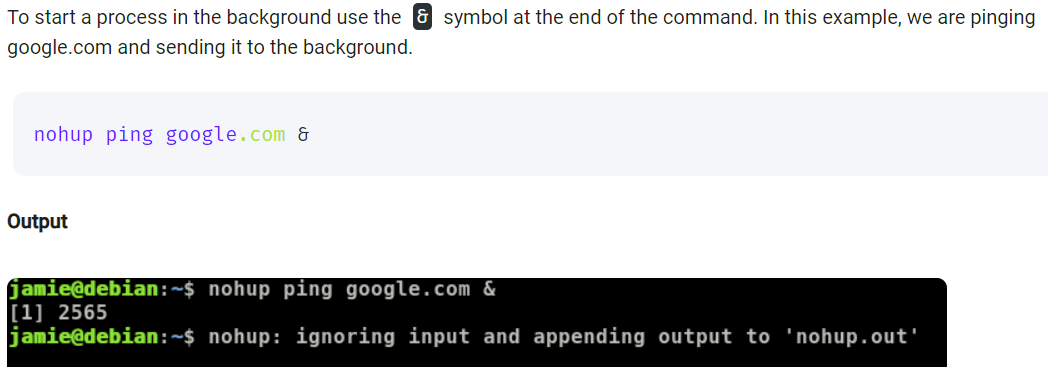


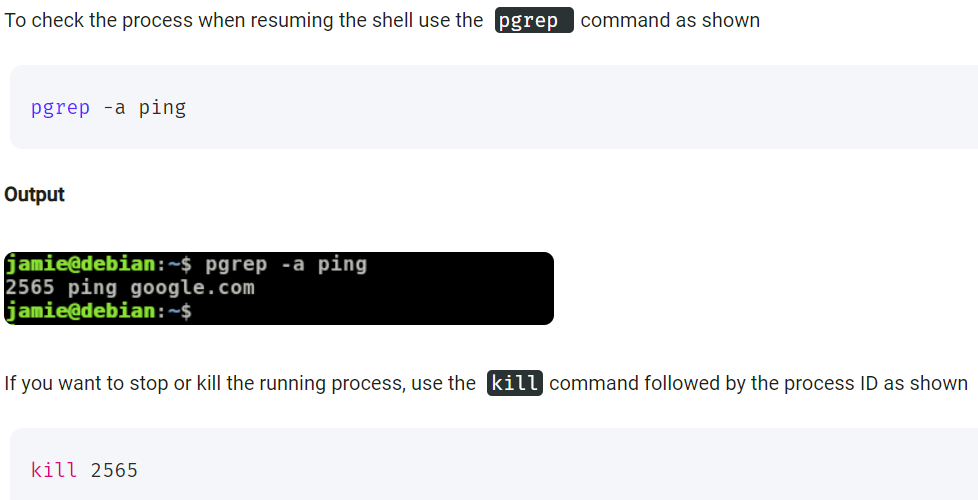
* If we want to store the output of nohup in separate file then use the below command.



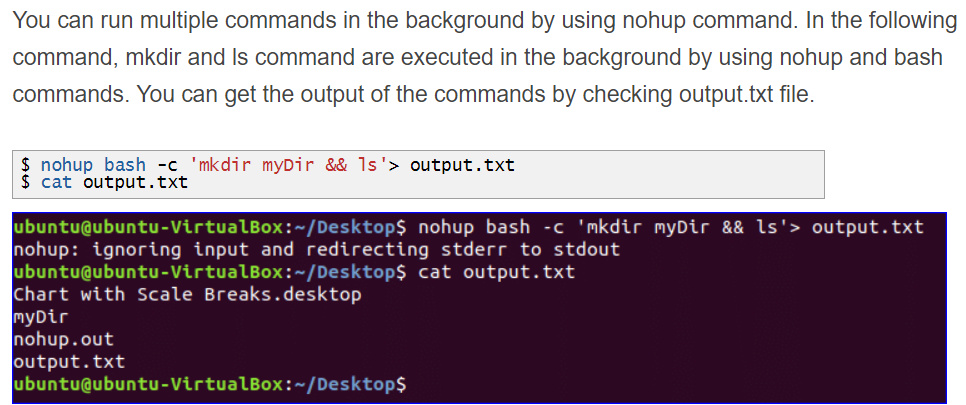
* Using the nohup command with “&” symbol as below.



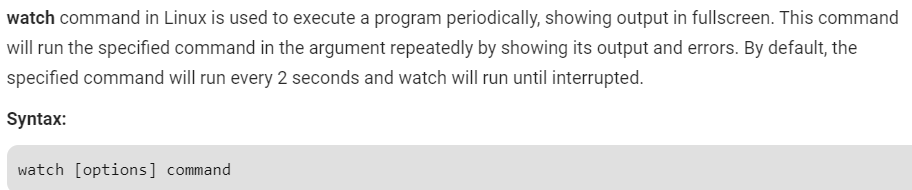




* Combining multiple commands in nohup as below.



**watch command:**



**Examples:**

**-d, –differences:** This option highlights the differences between successive updates. The options will be going to read the optional argument which changes highlight to be permanent, allowing the user to see what has changed at least once since the first iteration.

* **Watch -d free -m**

