**LS COMMAND**

ls -g 🡪 donot list the owner

ls -G 🡪 in a long listing dont print group names

ls -H 🡪 follow symbolic links listed on the command line

ls -i 🡪 prints the index no. of file

ls -L 🡪shows info of the file

ls -m 🡪 fill width with a comma separated list of entries

ls -n 🡪 list numeric user and group ids

ls -N 🡪 print raw entry names

ls -o 🡪 do not list group info

ls -p 🡪append / indicator to directories(put / in dir file)

ls - Q or ls --quote-filename🡪 enclose entry names in double quotes (" ")

ls -r or ls –reverse🡪reverse order while sorting (sorting in reverse order)

ls -R🡪list subdirctories recursively

ls -s or ls –size🡪print the allocated size of each file, in blocks

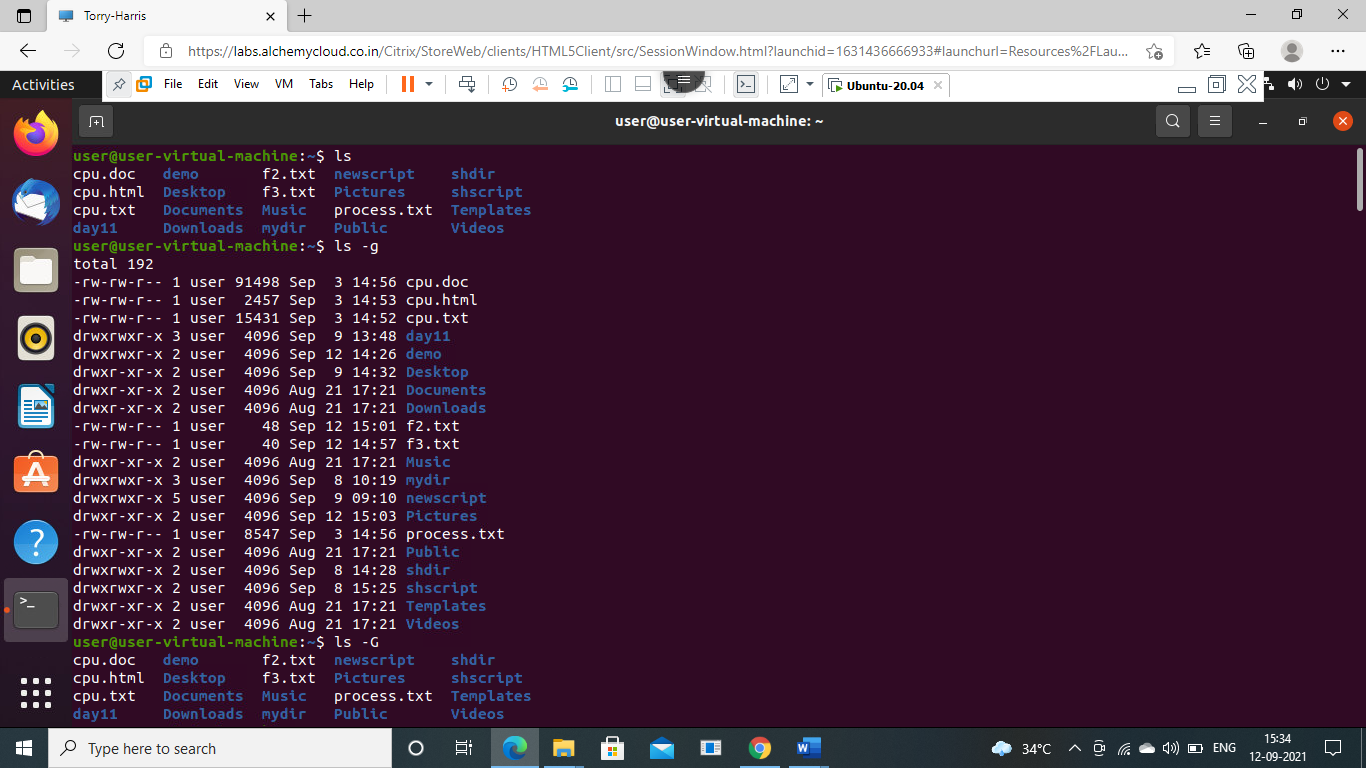
ls -S🡪sort by file size

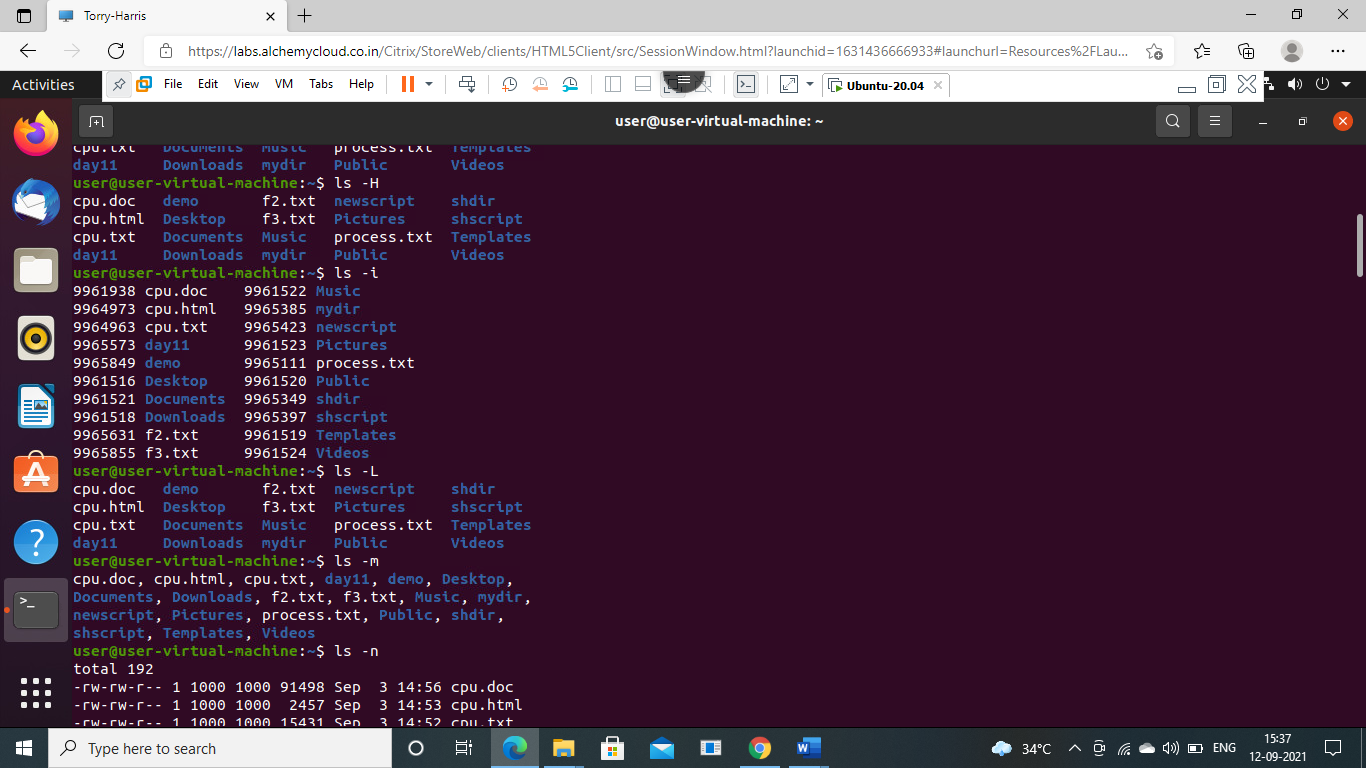
ls -t 🡪sort by modification time

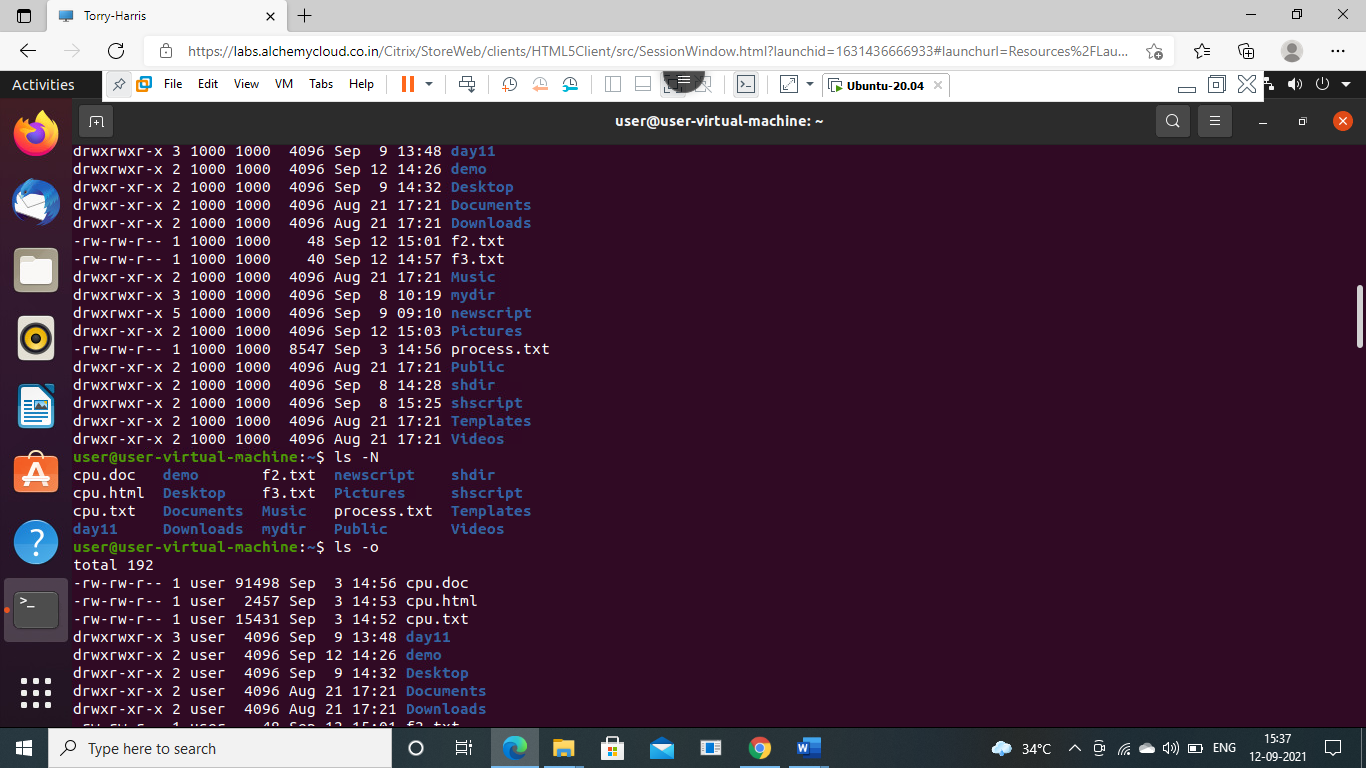
ls -T 🡪assume tab

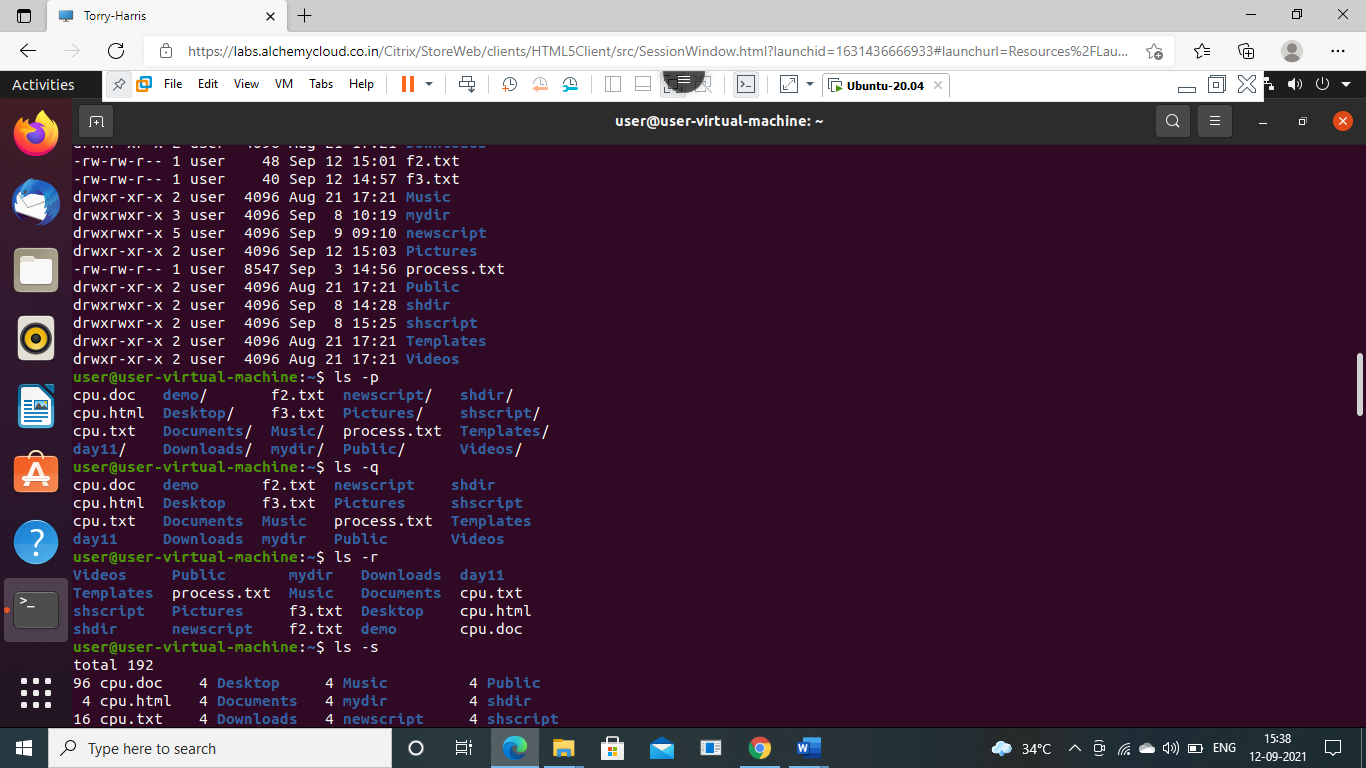
ls -v 🡪natural sort numbers within text

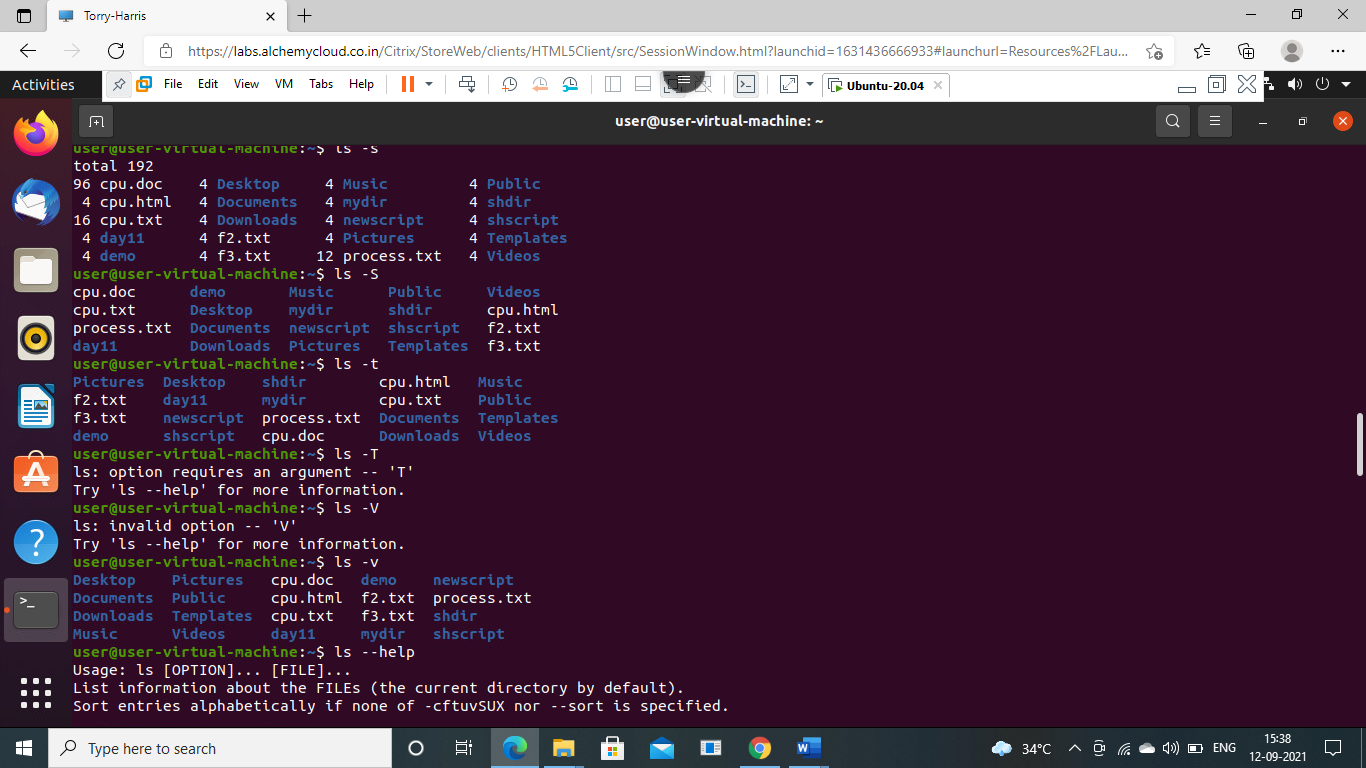
ls --help 🡪display this help and exit

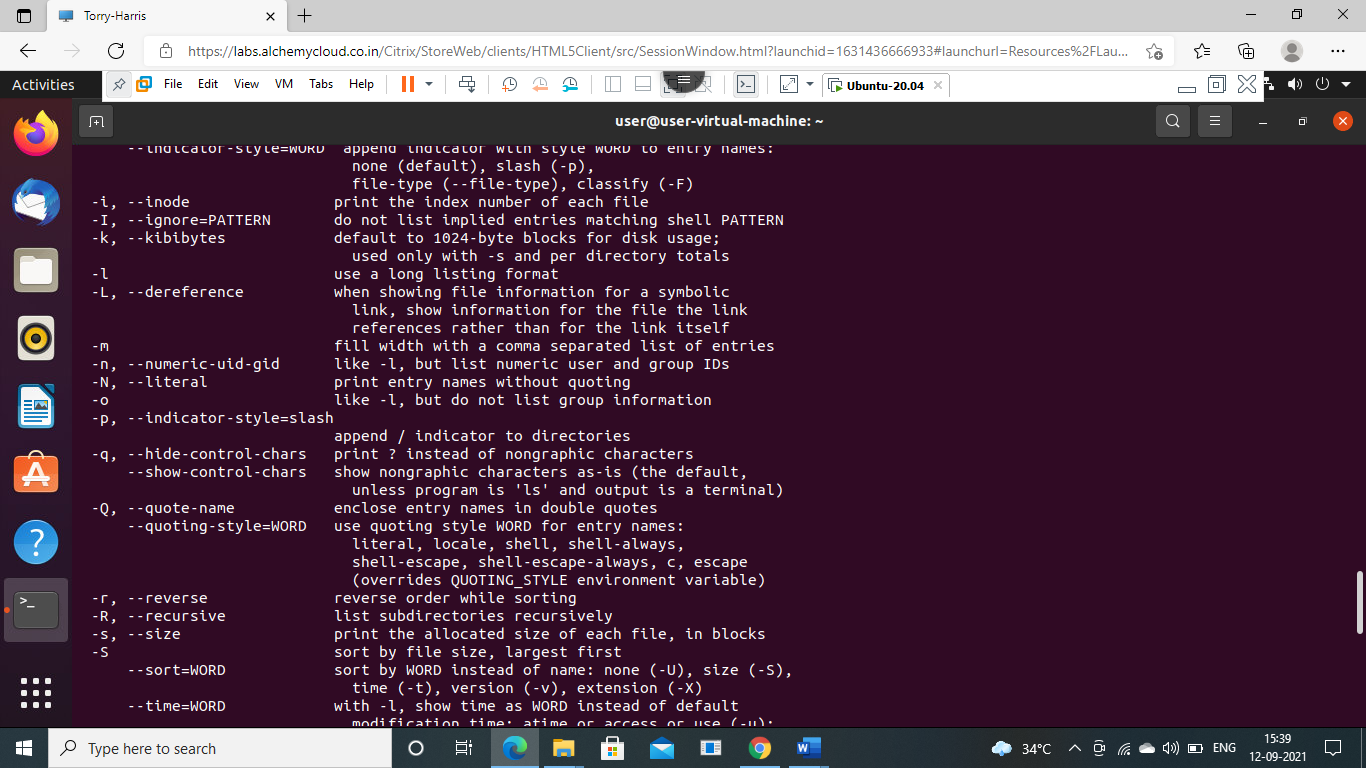












**CUT COMMAND**

cut -b 1,2,3 file.txt 🡪 List without ranges

cut -b 1-3,5-7 state.txt 🡪 List with ranges

cut -b 1- state.txt 🡪 from 1st byte to end byte

cut -b -3 state.txt 🡪from 1st byte to 3rd byte

cut -c4 file.txt 🡪 to print characters in a line by specifying the position of the characters.

cut -c4,6 file.txt 🡪 more than one character at a time

cut -c4-7 file.txt 🡪 print a range of characters in a line by specifying the start and end position of the characters.

cut -c-6 file.txt 🡪 first six characters in a line, omit the start position and specify only the end position.

cut -c10- file.txt 🡪 To print the characters from tenth position to the end, specify only the start position and omit the end position.

cut -c- file.txt 🡪 If you omit the start and end positions, then the cut command prints the entire line.

cut --complement -c 5 state.txt 🡪 except 5th char all will be there

cut -d (delimiter) 🡪f (field)

cat marks.txt

cut -d- -f2 marks.txt 🡪it shows only the 2nd field.

cut -d- -f2,3 marks.txt 🡪 it shows only the 2nd and 3rd fields.

cut -d- -f1- marks.txt 🡪 it shows only fields from 1st field to the last.

cut -d- -f-3 marks.txt 🡪 it shows only fields from starting field to the 3rd field.

cut -d- -f1-3 marks.txt 🡪 it shows only fields from 1st field to the 3rd field.

complement

cut -complement -d- -f2 marks.txt 🡪 it shows all the fields except field 2.

cut -complement -d- -f2,3 marks.txt 🡪 it shows all the fields except field 2 and 3.

cut -complement -d- -f1- marks.txt 🡪 it does'nt shows the fields from field 1 to the end.

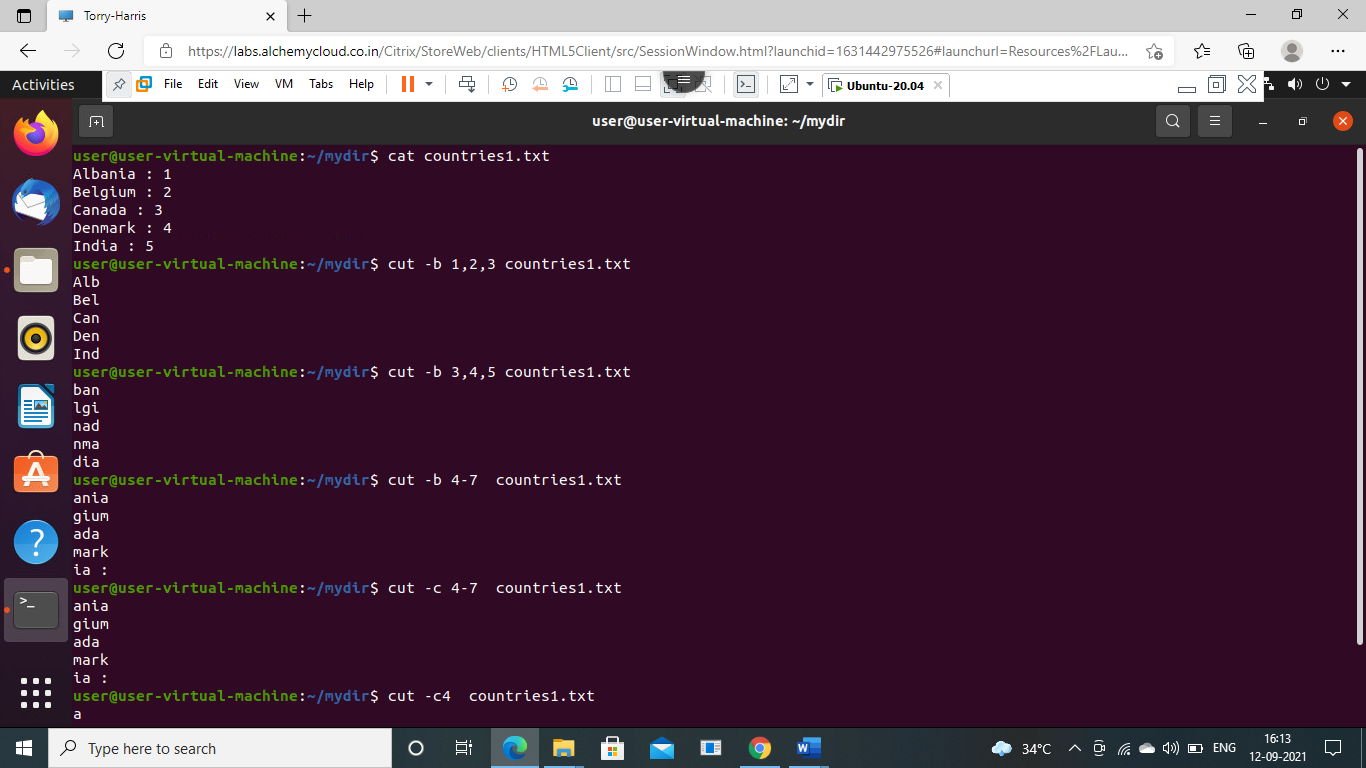
cut -complement -d- -f-3 marks.txt 🡪 it shows all the fields except field from starting field to 3.

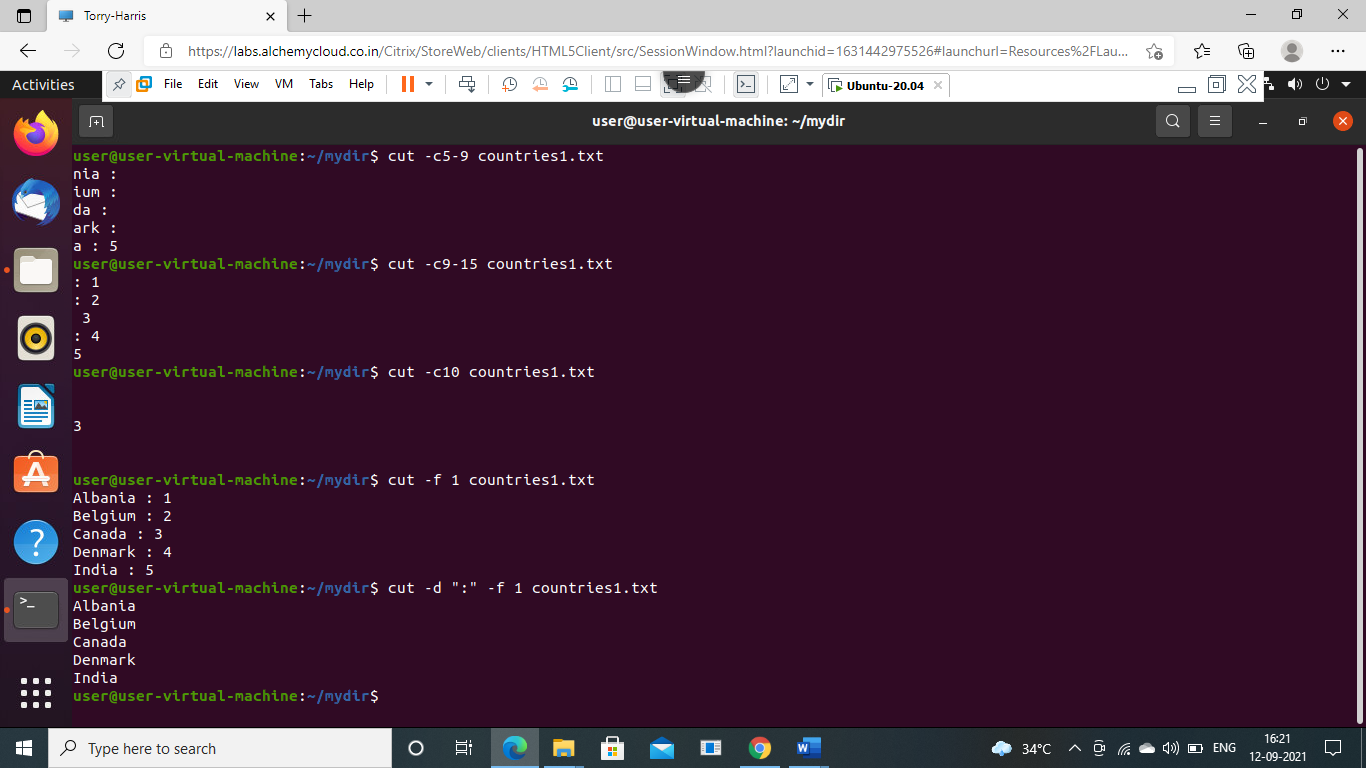
cut -complement -d- -f1-3 marks.txt 🡪 it shows all the fields except field from 1st field to 3rd field.

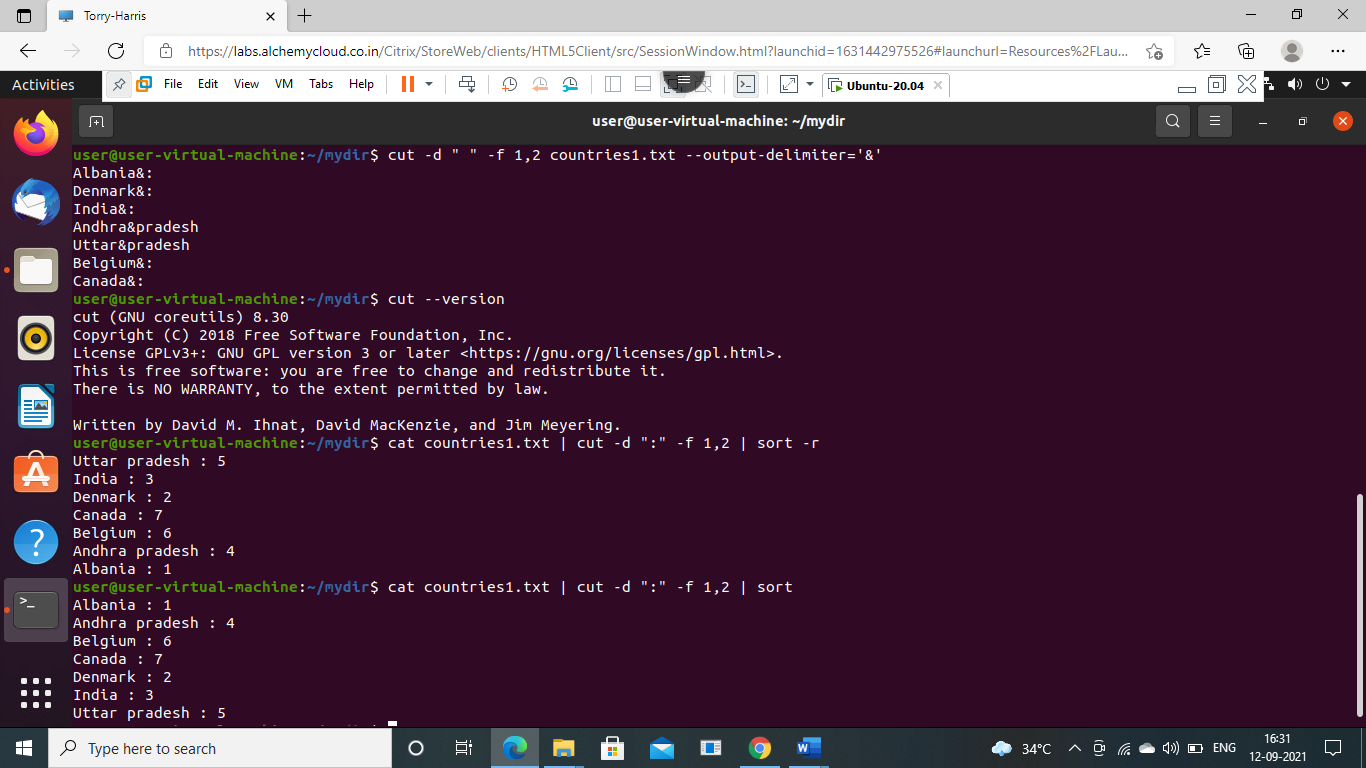
output-delimiter

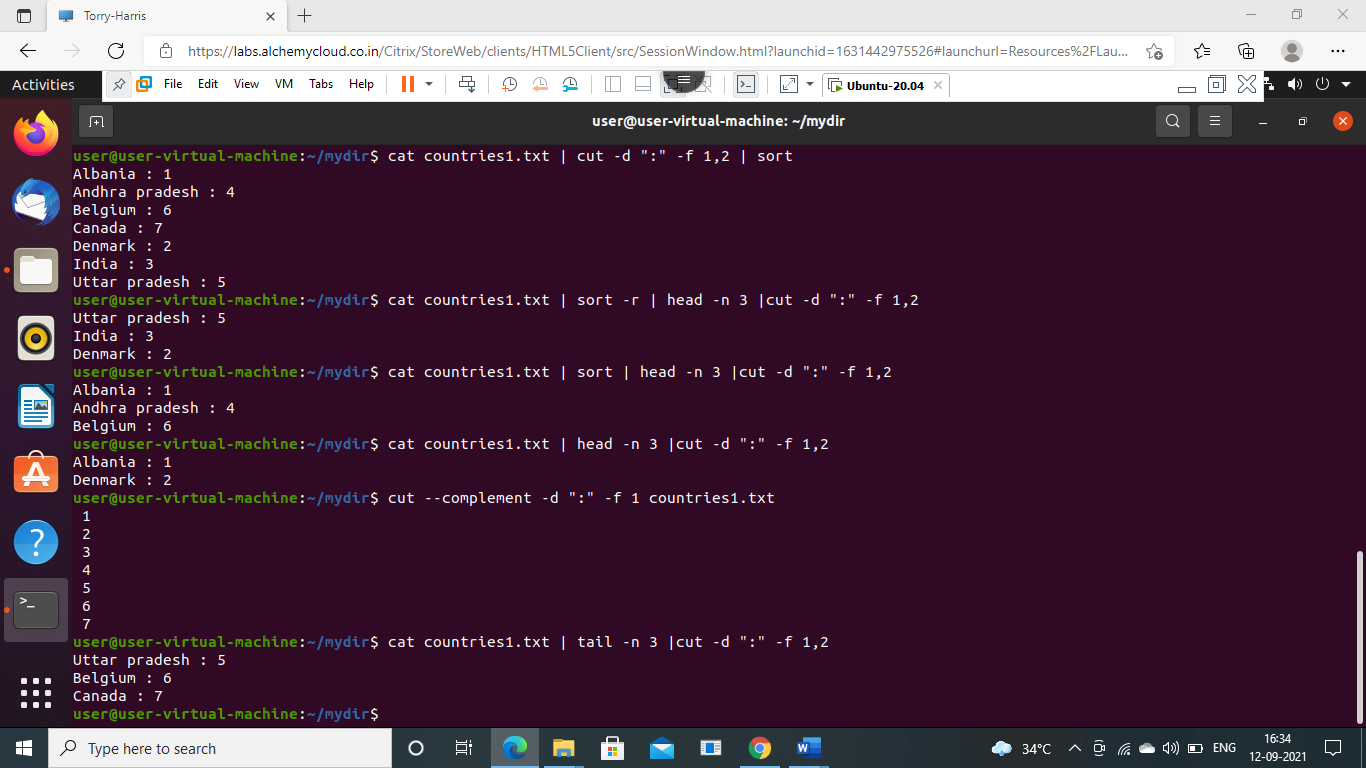
cut -d- -f2 marks.txt --output-delimiter="%" 🡪 it changes the delimiter '-' to '%'

cut -d% -f2 marks.txt --output-delimiter="\*" 🡪 it changes the delimiter '%' to '\*'









**SORT COMMAND**

Sort file.txt🡪Sort the content in ascending order

Sort -r 🡪 file.txt-display the sort in reverse

Sort -k2 🡪 file.txt-Sort with respect to 2nd field

Sort -c 🡪 file.txt-check the content in the file is sorted

Sort -u 🡪 file.txt-sort and removes duplicate

Sort -n🡪 file.txt-sort the number

Sort -nu🡪 file.txt-sort the number and removes duplicate

Sort -M 🡪 file.txt-sort the months

Sort -o 🡪output.txt file.txt-sorts and saves the output in output file

Sort -t “,” –k 1 🡪file.txt-sort in 1st field for delimiter in the content

Sort -t “,” –k2nr 🡪file.txt-sort in 2st field for delimiter in the content and reverse it

