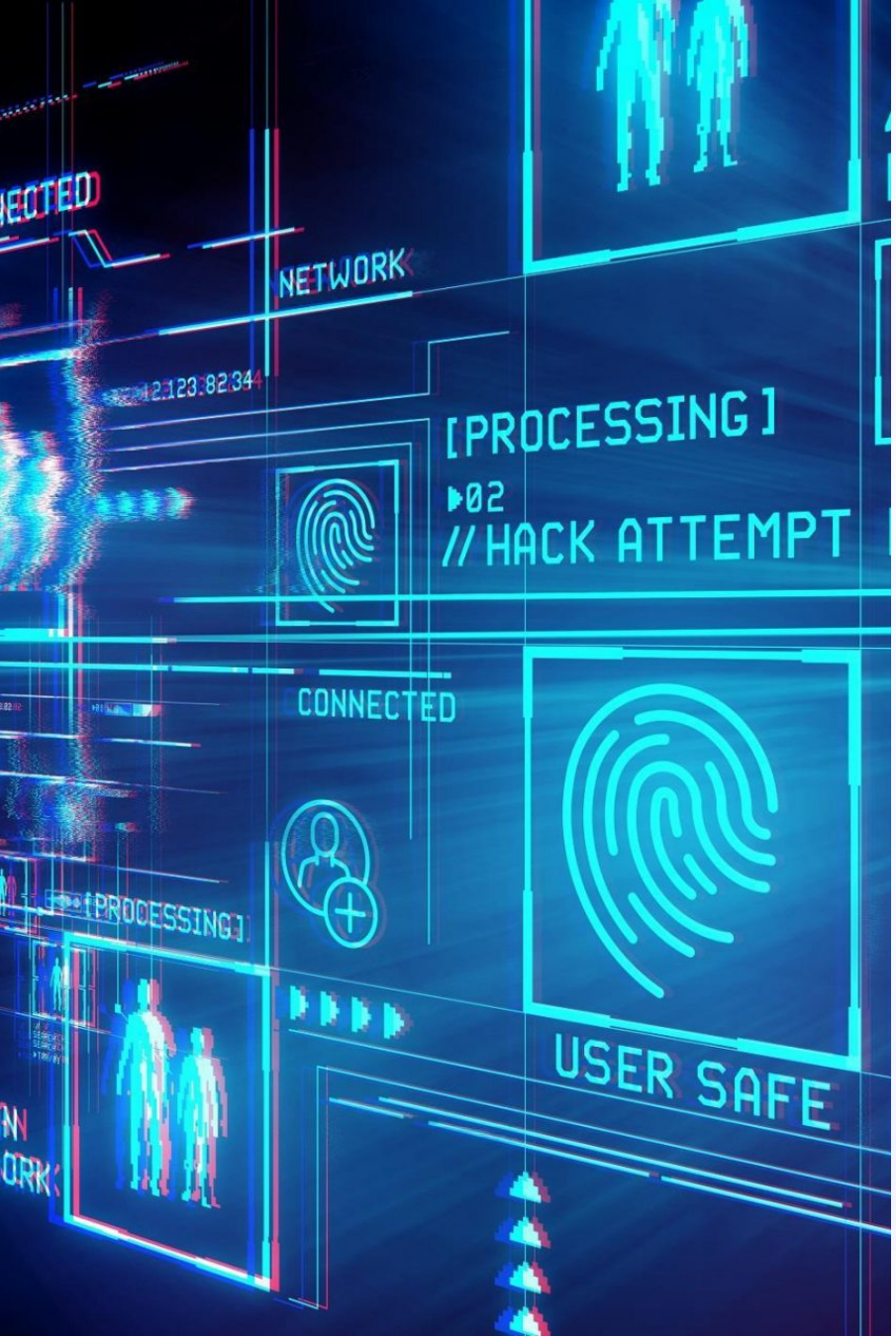


Text processing utilities

Linux offers a wide range of text processing utilities and commands that are extremely useful for manipulating and analyzing text data

by Sai Rohith

Made with Gamma

Exploring the versatile grep command

Search for a Word in a File: grep "search_term" file.txt

This command searches for the specified "search_term" in the "file.txt" and prints all lines containing the term.

Case-Insensitive Search: grep -i "pattern" file.txt

Use the `-i` option to perform a case-insensitive search, so it matches "Pattern," "PATTERN," "pattern," and so on in "file.txt."

Search Multiple Files: grep "pattern" file1.txt file2.txt

You can search for the same "pattern" in multiple files, listing them one after the other.

Invert Match (Exclude Lines): grep -v "pattern" file.txt

The `-v` option inverts the match, showing lines that do not contain the specified "pattern."

Recursive Directory Search: grep -r "pattern" /path/to/directory

`-r` searches for "pattern" recursively in all files under the specified directory.

The `grep` command is a powerful text search and pattern matching tool available in Unix-like operating systems, including Linux. Its name stands for "Global Regular Expression Print." `grep` is used for searching and filtering text based on patterns or regular expressions within one or more files or a stream of input.

```
sai@DESKTOP-PTH4AFE:~$ cat rc
hi my name is rc
iam the topper of our college
i got 98LPA
iam from aids
sai@DESKTOP-PTH4AFE:~$ cat rishi
hi iam rishinayan also known as 69
iam the handsome hunk of our college
and iam also from aids
RISHINAYAN
sai@DESKTOP-PTH4AFE:~$ grep "aids" rc
iam from aids
sai@DESKTOP-PTH4AFE:~$ grep -i "aids" rc
iam from aids
sai@DESKTOP-PTH4AFE:~$ grep -i "aids" rishi
and iam also from aids
sai@DESKTOP-PTH4AFE:~$ grep -i "rishinayan" rishi
hi iam rishinayan also known as 69
RISHINAYAN
sai@DESKTOP-PTH4AFE:~$ grep "aids" rishi rc
rishi:and iam also from aids
rc:iam from aids
sai@DESKTOP-PTH4AFE:~$ grep -v "aids" rc
hi my name is rc
iam the topper of our college
i got 98LPA
```

Made with Gamma

SED COMMANDS:

The `sed` command, short for "stream editor," is a powerful text processing tool available in Unix-like operating systems, including Linux. It is designed for performing text manipulation and transformation tasks on text streams or files. `sed` is particularly useful for tasks such as search and replace, text deletion, insertion, and more. It works by applying a series of commands to each line of input text, allowing you to modify the text based on specified patterns or regular expressions.

```
i got 98LPA
sai@DESKTOP-PTH4AFE:~$ sed 's/LPA/cr/' rc
hi my name is rc
iam the topper of our college
i got 98cr
iam from aids
sai@DESKTOP-PTH4AFE:~$ cat >> rishi
as iam from dept of aids placement is guaranteed
iam in aids dept club
sai@DESKTOP-PTH4AFE:~$ sed 's/aids/AI&DS/g' rishi
hi iam rishinayan also known as 69
iam the handsome hunk of our college
and iam also from AIaidsDS
RISHINAYAN
as iam from dept of AIaidsDS placement is guaranteed
iam in AIaidsDS dept club
sai@DESKTOP-PTH4AFE:~$ sed -i 's/aids/AI&DS/g' rishi
sai@DESKTOP-PTH4AFE:~$ cat rishi
hi iam rishinayan also known as 69
iam the handsome hunk of our college
and iam also from AIaidsDS
RISHINAYAN
as iam from dept of AIaidsDS placement is guaranteed
iam in AIaidsDS dept club
sai@DESKTOP-PTH4AFE:~$ sed -i 's/aids/AIDS/g' rishi
sai@DESKTOP-PTH4AFE:~$ sed -i 's/AIaidsDS/AIDS/g' rishi
sai@DESKTOP-PTH4AFE:~$ sed 's/AIaidsDS/AIDS/g' rishi
hi iam rishinayan also known as 69
iam the handsome hunk of our college
and iam also from AIAIDSDS
RISHINAYAN
as iam from dept of AIAIDSDS placement is guaranteed
iam in AIAIDSDS dept club
sai@DESKTOP-PTH4AFE:~$ sed -i 's/AIaidsDS/AIDS/g' rishi
sai@DESKTOP-PTH4AFE:~$ sed -n '/aids/p' rc
iam from aids
sai@DESKTOP-PTH4AFE:~$ sed -n '2,4p' rishi
iam the handsome hunk of our college
and iam also from AIAIDSDS
RISHINAYAN
sai@DESKTOP-PTH4AFE:~$ sed 's/$/ hero/' rc
hi my name is rc hero
```

Find and Replace Text (1st Occurrence):

sed 's/old_text/new_text/' file.txt

Find and Replace Text (All Occurrences):

sed 's/old_text/new_text/g' file.txt

Print Lines Matching a Pattern:

sed -n '/pattern/p' file.txt

Print Specific Line Numbers:

sed -n '2,5p' file.txt

Append Text to End of Each Line:

sed 's/$/ additional_text/' file.txt

Insert Text at the Beginning of Each Line:

sed 's/^/prefix_text/' file.txt

Transform to Uppercase and lowercase:

sed 's/.*/\U&/' file.txt

sed 's/.*/\L&/' file.txt

Made with Gamma

AWK COMMANDS:

The `awk` command is a versatile text processing tool that is widely used in Unix-like operating systems, including Linux. It is named after its creators: Alfred Aho, Peter Weinberger, and Brian Kernighan. `awk` is designed for processing and manipulating text data, making it a powerful tool for tasks such as text extraction, transformation, and reporting.

Print Specific Columns from a CSV File:

```
awk -F ',' '{print $1, $3}' file.csv
```

Calculate the Sum of a Column:

```
awk '{sum += $1} END {print sum}' file.txt
```

Print Lines Based on a Condition:

```
awk '$2 > 50 {print}' data.txt
```

Count Lines Matching a Pattern:

```
awk '/pattern/ {count++} END {print count}' file.txt
```

Print Unique Values in a Column:

```
awk '{print $1}' file.txt | sort | uniq
```

Print the Number of Fields in Each Line:

```
awk '{print NF}' file.txt
```

Calculate and Print Averages:

```
awk '{sum += $1} END {print "Average:", sum/NR}' file.txt
```

```
vnr@vnrvjiet01:~$ ls
dir1  emp.txt  f1  f2
vnr@vnrvjiet01:~$ awk '{print}' emp.txt
ajay manager account 45000
sunil clerk account 25000
varun manager sales 50000
amit manager account 47000
vnr@vnrvjiet01:~$ awk '/manager/{print}' emp.txt
ajay manager account 45000
varun manager sales 50000
amit manager account 47000
vnr@vnrvjiet01:~$ awk '{print $1, $4}' emp.txt
ajay 45000
sunil 25000
varun 50000
amit 47000
vnr@vnrvjiet01:~$ awk '{print NR, $0}' emp.txt
1 ajay manager account 45000
2 sunil clerk account 25000
3 varun manager sales 50000
4 amit manager account 47000
vnr@vnrvjiet01:~$ awk '{print $1, NF}' emp.txt
ajay 4
sunil 4
varun 4
amit 4
vnr@vnrvjiet01:~$ awk '{print $1, $NF}' emp.txt
ajay 45000
sunil 25000
varun 50000
amit 47000
vnr@vnrvjiet01:~$ awk 'NR==2, NR==4 {print NR, $0}' emp.txt
2 sunil clerk account 25000
3 varun manager sales 50000
4 amit manager account 47000
vnr@vnrvjiet01:~$
```

Made with Gamma

CUT COMMANDS:

The `cat` command, short for "concatenate," is a fundamental command-line utility in Unix-like operating systems, including Linux. Its primary purpose is to display the contents of one or more text files or concatenate the content of multiple files and display the result on the terminal. While its primary function is to concatenate and display text, it is often used for various other tasks as well.

```
vnr@vnrvjiet01:~$ cut -c 2,5,7 state.txt
nr
rah
sm
ir
hti
vnr@vnrvjiet01:~$ cut -c 1-7 state.txt
Andhra
Arunach
Assam
Bihar
Chhatti
vnr@vnrvjiet01:~$ cut --complement -c 5 state.txt
Andha Pradesh
Arunchal Pradesh
Assa
Biha
Chhatisgarh
vnr@vnrvjiet01:~$ cut -f 1 state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
vnr@vnrvjiet01:~$ cut -d " " -f 1 state.txt
Andhra
Arunachal
Assam
Bihar
Chhattisgarh
```

Cut Specific Columns from a Tab-Delimited File:

cut -f 2,4 file.txt

Cut Specific Columns from a CSV File:

cut -d ',' -f 1,3 file.csv

```
vnr@vnrvjiet01:~$ ls
dir1  emp.txt  f1  f2  state.txt
vnr@vnrvjiet01:~$ cat state.txt
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chhattisgarh
vnr@vnrvjiet01:~$  cut -b 1,2,3 state.txt
And
Aru
Ass
Bih
Chh
vnr@vnrvjiet01:~$ cut -b 1-3,5-7 state.txt
Andra
Aruach
Assm
Bihr
Chhtti
```

Cut Characters from a Line:

echo "Hello, World" | cut -c 1-5

Cut Fields Based on a Different Separator:

echo "apple|banana|cherry" | cut -d '|' -f 2

Cut Fields by Setting a Delimiter Range:

echo "abcdefghij" | cut -c 3-7

Made with Gamma

SORT COMMANDS:

The** sort **command is a Unix and Linux command-line utility used to sort lines of text files or standard input. Its primary function is to arrange the lines of text in ascending or descending order based on various criteria, such as alphabetical order or numerical order. The sorted output can be displayed on the terminal or redirected to a file.

```
vnr@vnrvjiet01:~$ cat names.txt
abhishek
Rohith
ravi
pooja
Nithya
INS
vnr@vnrvjiet01:~$ sort names.txt
INS
Nithya
Rohith
abhishek
pooja
ravi
vnr@vnrvjiet01:~$ sort -r names.txt
ravi
pooja
abhishek
Rohith
Nithya
INS
```

Sort a File Alphabetically:

sort file.txt

Sort a File Numerically:

sort -n numbers.txt

```
vnr@vnrvjiet01:~$ cat > num.txt
50
26
47
85
14
vnr@vnrvjiet01:~$ sort num.txt
14
26
47
50
85
vnr@vnrvjiet01:~$ sort -r num.txt
85
50
47
26
14
```

Sort a File in Reverse Order:

sort -r file.txt

Sort and Remove Duplicate Lines:

sort -u file.txt

```
vnr@vnrvjiet01:~$ cat num.txt
50
26
47
85
14
vnr@vnrvjiet01:~$ sort -n num.txt
14
26
47
50
85
vnr@vnrvjiet01:~$ sort -nr num.txt
85
50
47
26
14
```

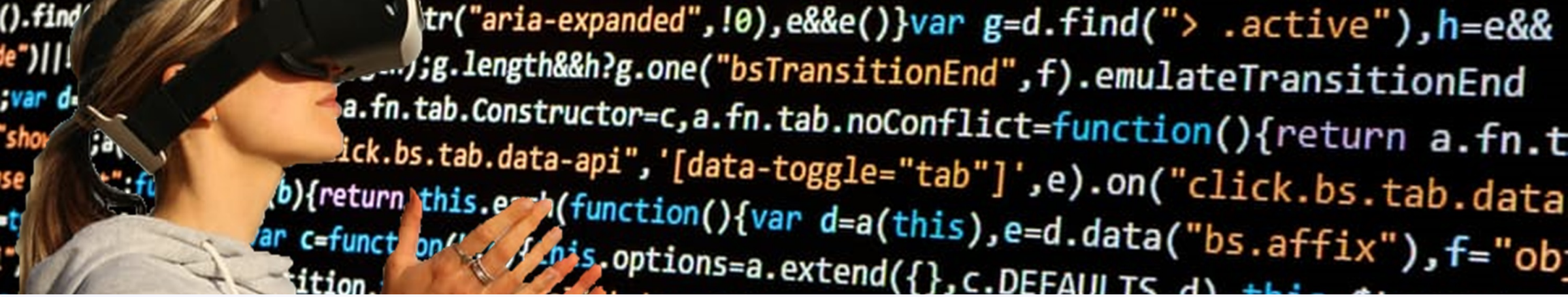
Sort a CSV File by a Specific Column:

sort -t ',' -k 2,2 -n data.csv

Sort and Output to a New File:

sort input.txt > sorted.txt

Made with Gamma



Conclusion:

In conclusion, text processing utilities are essential tools in the Unix and Linux command-line environment for handling and manipulating text data efficiently. These utilities offer a wide range of capabilities to perform tasks such as searching, extracting, transforming, and analyzing text.

In summary, text processing utilities are foundational to the Unix and Linux command-line ecosystem, offering powerful and flexible ways to manipulate and extract information from text data. Mastering these utilities can significantly improve one's ability to work with text-based information efficiently.

Made with Gamma