

Search any line

Search any line that contains the word dracula regardless of case and with number line

```
grep -in 'dracula' ~/Documents/Books/dracula.txt
```

Search for all the lines that do not contain the word 'war'

```
grep -v 'war' ~/Documents/Books/war-and-peace.txt
```

Search and display only the matched string(pattern)

```
grep -o 'pride' ~/Documents/Books/war-and-peace.txt
```

Display a list of users with the/bin/bash login shell

```
grep -i "/bin/bash" /etc/passwd
```

Display your user's information as stored in the /etc/passwd

```
grep -i $USER /etc/passwd
```

Search for a given strings inside files in a given directory .

```
grep -iR 'conf' /etc/
```

Search and display the total number of times a given word appears in a file

```
grep -wc '/bin/bash' /etc/passwd
```

The (caret) symbol matches the empty string at the beginning of a line. Search for all the lines that start with a given word.

```
grep -ni '^dracula' ~/Documents/Books/dracula.txt
```

Search for all the lines that ends with the string "nonlogin"

```
grep -n '$nonlogin' /etc/passwd
```

-i Enables case insensitivity. (it wil match regardless of case) -n Displays line number for every line matched -E Treats the pattern (search criteria) as a basic regular expression -G Treats the pattern (search criteria) as a basic regular expression -v Inverts the search (looks for the opposite of the given criteria) -o Only displays the matched string -c Search and display the total number of times a pattern is mathced. -w Matches only given word (pattern) by itself -r, -R Matches recursively. Useful for searching files in a given directory.

```
awk awk + options {awk command} + file + file to save
```

## Basic Example

```
awk '{print $1}' ~/Documents/Csv/cars.csv
```

Print first field of /etc/passwd file (user command)

```
awk -F: '{print $1}' /etc/passwd
```

Print the last field of the /etc/passwd (login shell)

```
awk -F: '{print $NF}' /etc/passwd
```

Print the first and last field of the etc/passwd

```
awk -F: '{print $1, " = ", $NF }'
```

Print the first and 3 field with line numbers

```
awk -F: '{print NR,$1,$4}' /etc/passwd
```

Print the first and 4th field with a different field separator

```
awk -F: '{OFS="-"}{print $1,$4}' /etc/passwd
```

Start printing a file from a given line (exclude the first 2 lines)

```
awk 'NR > 3{ Print }' /etc/passwd
```

sed command

```
sed options +ic sed script + file
```

## Bas Example

Replacing a string in given file globally (replace false for true)

```
sed 's/false/true/g' ~/Documents/sample_files/Json/joke.john
```

To delete a particular line (line 5)

```
sed '3d' ~/Documents/sample_files/Code/helloWorld.py
```

To delete the last line

```
sed '$d' ~/Documents/sample_files/Code/helloWorld.py
```

To delete line from range x to y

```
sed '2,4d' ~/Documents/sample_files/Code/helloWorld.py
```

To delete from a given number to last line

```
sed '3,$d' ~/Documents/sample_files/Code/helloWorld.py
```

To delete pattern matching line in a file

```
sed '/fav/d' ~/Documents/sample_files/Code/helloWorld.py
```

To insert one blank lined after each line

```
sed G helloWorld.c
```

To inset two blank lines

```
sed 'G;G' helloWorld.c
```

to delete balnk lines and insert one blank line after each line

```
sed '/^\$d;G' helloWorld.c
```

Insert 5 spaces to the left of every line

```
sed 's/^/ /5' helloWorld.c
```

Explain how to use the pipe (|) for redirecting the output of a command to another. Include at least 3 examples

Usage command\_1 | command\_2 | command\_3 | .... | command\_N

## Basic Examples

Use grep to look for a string in a particular man page

```
man ls | grep "human readable" man ls| grep "^[[:space:]]*[[[:punct:]] man ls| grep "long" or "comma"
```

Explain how to save the output of a command to a file (>). Include at least 3 examples

Explain how to append the output of a command to a file. Include at least 3 examples