

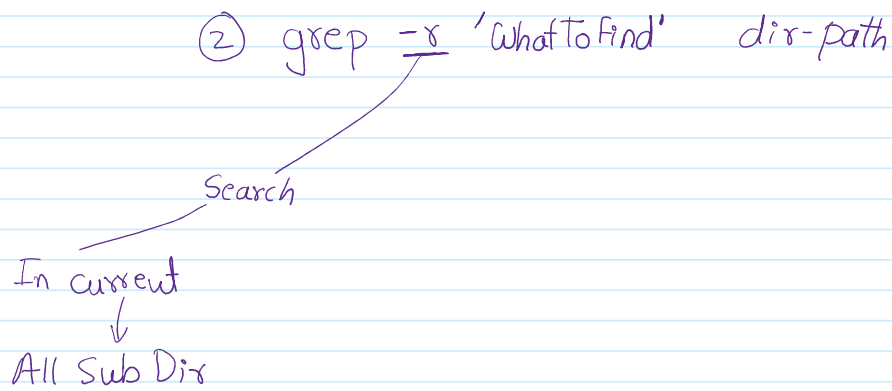
⑥ Text Processing

- 1] grep - Search Pattern.
- 2] sed - Text Transform
- 3] awk - Manipulate & Process Text data Column & Rows.
- 4] cut - Extract Column/Field.
- 5] sort - Sort the lines of a file

grep

① grep 'Pattern' Path 

② grep -r 'WhatToFind' dir-path



③ `syslog.gz` → `zgrep 'pattern' 'path.gz'`

④ `grep 'error' folderPath --exclude-dir=abc`

`syslog` → `error`

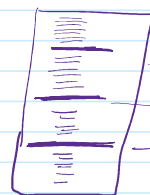
-A 2 ==
-B 3 ==

⑤ `grep -A 2 -B 3 'error' /var/log/syslog`

Sed

← X 1000 files X 50 Lines = ?

① `sed 's/oldtext/newtext/g' *.txt`



delete abc line.

② `sed '/pattern/d' file.txt`

③ `sed '2d' *.txt` → Delete line number 2.

③ sed '2d' *.txt → Delete line number 2.

```
$ awk '{print $1,$3}' file.txt
```

```
$ awk '/pattern/{print}' file.txt
```

Cut

```
$ cut -d',' -f2 file.txt
```

```
$ echo "Welcome to Linux Command" | cut -c2-5  
elco
```

```
$ echo "Welcome to Linux Command" | cut -c12-
```

Sort

```
cat > data.txt  
www  
rrr  
vvv  
bbb  
ccc  
sss  
aaa
```

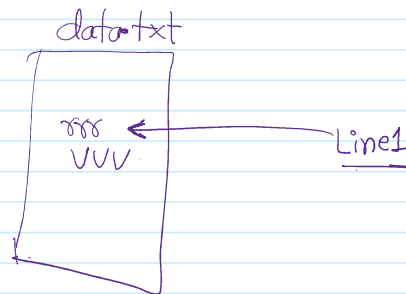
```
sort data.txt  
aaa  
bbb  
ccc  
rrr  
sss  
vvv  
www
```

```
sort -r data.txt  
www  
vvv  
sss  
rrr  
ccc  
bbb  
aaa
```

```
cat data.txt
www
rrr
vvv
www
bbb
ccc
rrr
sss
aaa
```

```
sort -u data.txt
aaa
bbb
ccc
rrr
sss
vvv
www
```

```
$ sed 's/rrr/qqq/g' data.txt > data1.txt
```



① sed '/pattern/i new-line' data1.txt

Diagram illustrating the components of the sed command:
- '/pattern/' is labeled 'what to find'
- 'i' is labeled 'Command (INSERT)'
- 'new-line' is labeled 'new data'

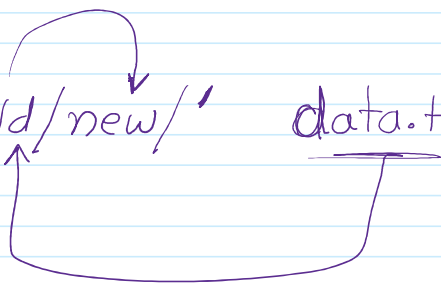
② sed '/pattern/a new-line' data2.txt

Diagram illustrating the components of the sed command:
- '/' is labeled 'append After'
- 'a' is labeled 'append After'

↘ append.
After
Line.

③

sed '2s/old/new/' data.txt

A hand-drawn diagram with two curved arrows. One arrow starts at the end of the command '2s/old/new/' and points back to the start of the same command. The second arrow starts at the end of the filename 'data.txt' and points back to the start of the command '2s/old/new/'.