

# Notes 8

---

## awk

### Description

- A powerful text processing tool used to manipulate and extract data from text files or streams based on patterns.

### Usage

- `awk + 'pattern {action}' + filename`

### Examples

- Print the first column of a file:
  - `awk '{print $1}' file.txt`
- Print lines containing "error":
  - `awk '/error/ {print}' logfile.txt`
- Calculate sum of the second column:
  - `awk '{sum += $2} END {print sum}' numbers.txt`

## sed

### Description

- Stream editor used to perform basic text transformations on input streams or files.

### Usage

- `sed + 'command' + filename`

### Examples

- Replace all instances of "foo" with "bar":
  - `sed 's/foo/bar/g' file.txt`
- Delete the second line:
  - `sed '2d' file.txt`
- Print only lines that contain "hello":
  - `sed -n '/hello/p' file.txt`

## less

### Description

- A pager program used to view the contents of a file one screen at a time, allowing both forward and backward navigation.

### Usage

- `less + filename`

## Examples

- Open a file for viewing:
  - `less file.txt`
- Search within the file (after opening):
  - Press `/` and type the search term, e.g., `/keyword`
- View a large log file:
  - `less /var/log/syslog`

>

## Description

- Redirects the standard output of a command to a file, overwriting it.

## Usage

- `command > filename`

## Examples

- Save the output of `ls` to a file:
  - `ls > filelist.txt`
- Overwrite a file with a string:
  - `echo "Hello" > greet.txt`
- Redirect output of a script:
  - `./myscript.sh > output.log`

>>

## Description

- Redirects standard output to a file, appending the content rather than overwriting.

## Usage

- `command >> filename`

## Examples

- Append current date to a file:
  - `date >> log.txt`
- Append new user to a list:
  - `echo "user123" >> users.txt`
- Log the output of a script without deleting previous logs:
  - `./myscript.sh >> log.txt`

|

## Description

- A pipe operator that takes the output of one command and sends it as input to another command.

## Usage

- `command1 | command2`

## Examples

- View only lines containing "root" from a user list:
  - `cat /etc/passwd | grep root`
- Count the number of files in a directory:
  - `ls | wc -l`
- View top 5 most used commands:
  - `history | awk '{print $2}' | sort | uniq -c | sort -nr | head -5`