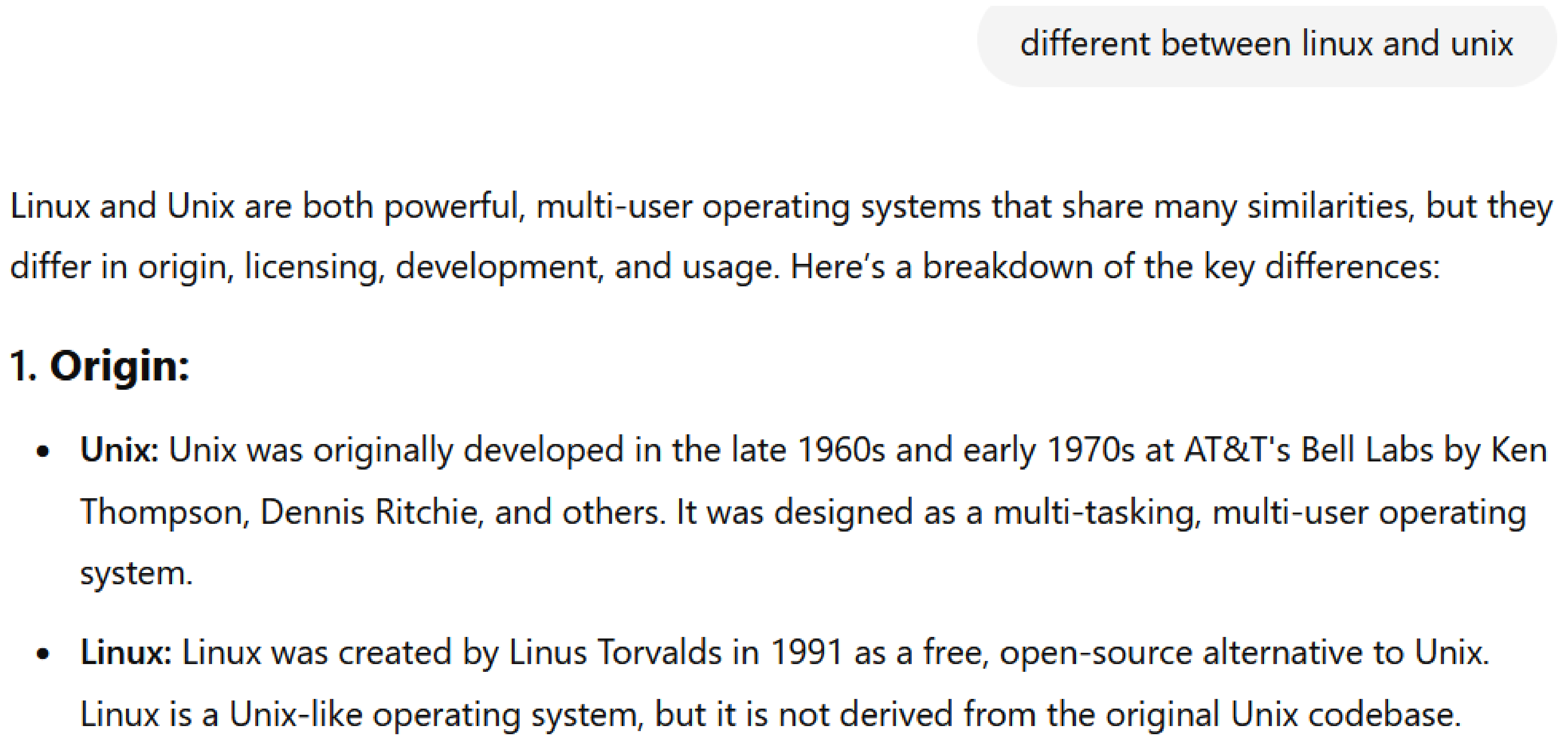
Linux vs Unix



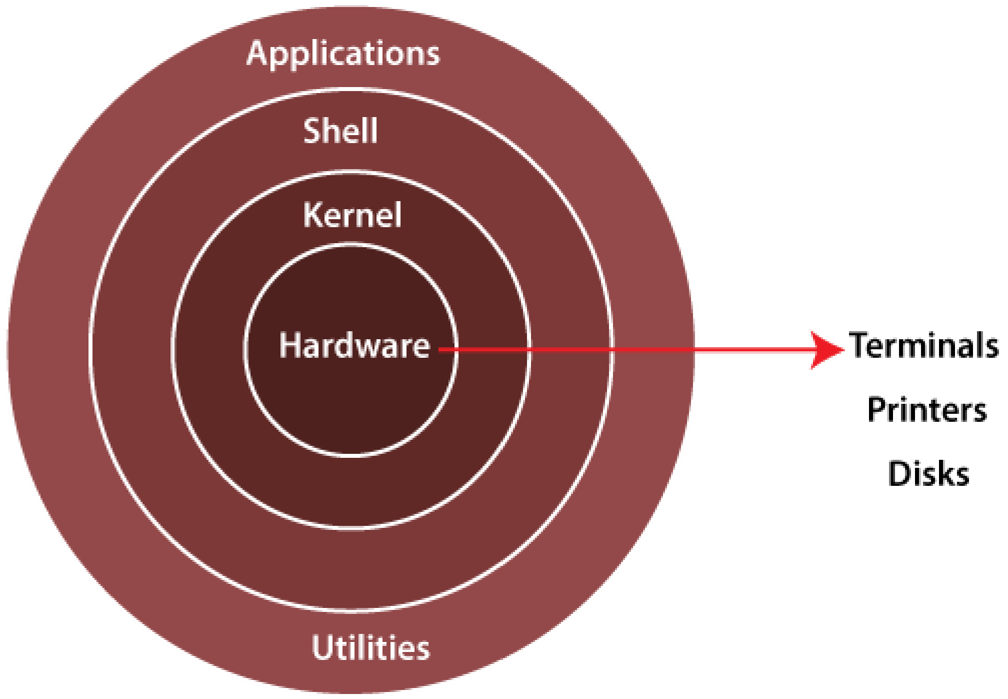
different between linux and unix

Linux and Unix are both powerful, multi-user operating systems that share many similarities, but they differ in origin, licensing, development, and usage. Here's a breakdown of the key differences:

1. Origin:

- **Unix:** Unix was originally developed in the late 1960s and early 1970s at AT&T's Bell Labs by Ken Thompson, Dennis Ritchie, and others. It was designed as a multi-tasking, multi-user operating system.
- **Linux:** Linux was created by Linus Torvalds in 1991 as a free, open-source alternative to Unix. Linux is a Unix-like operating system, but it is not derived from the original Unix codebase.

Linux/Unix Architecture

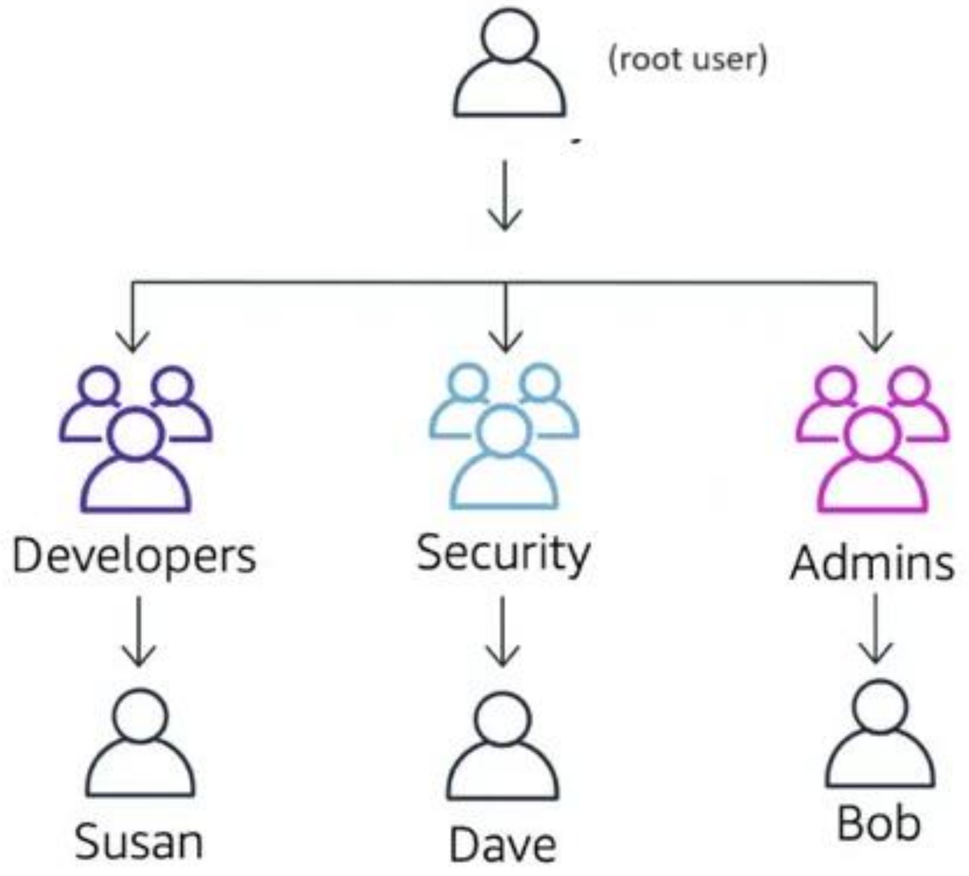


kernel

The kernel is the core component of the Linux operating system. It is responsible for managing the system's resources and allowing software applications to interact with the hardware.

Example: Resource Management, System Calls, Security and Permissions,

User, group and root



permission

Mode of reading a file

Read

write

execute

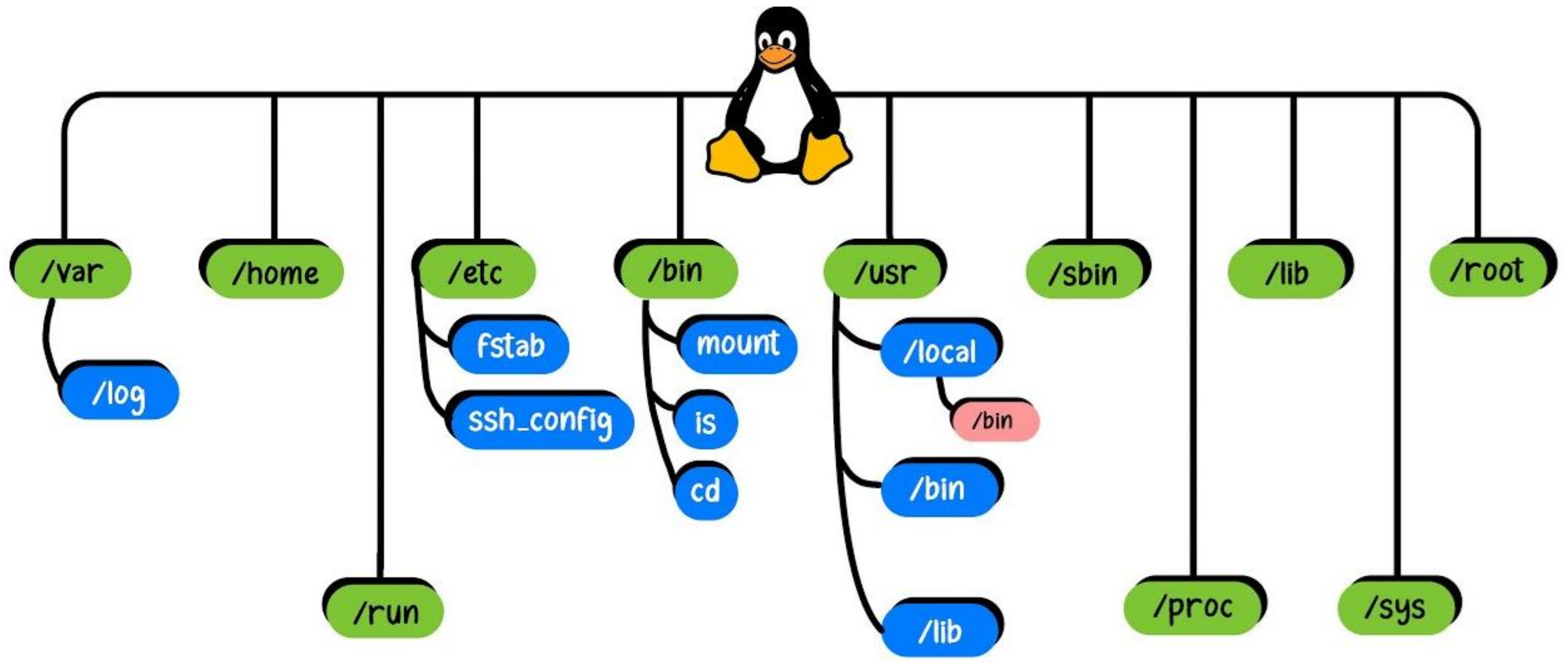
user_group_owner rwx_rwx_rwx [777]

Working with user and group

Command to list users > awk -F: '$3 >= 1000 {print $1}' /etc/passwd

Command to list group > cut -d: -f1 /etc/group

File system tree



Linux installation or server creation

Example: creation of EC2 in Amazon

ls , date and cal command

ls- list

date- printing date

cal- print calender

Working with directory

mkdir, rmdir, mkdir -p <directory name>

cp,mv, pwd

Working with file

cat, less, more, tail, head

Comparing file

```
diff <file1> <file2>
```

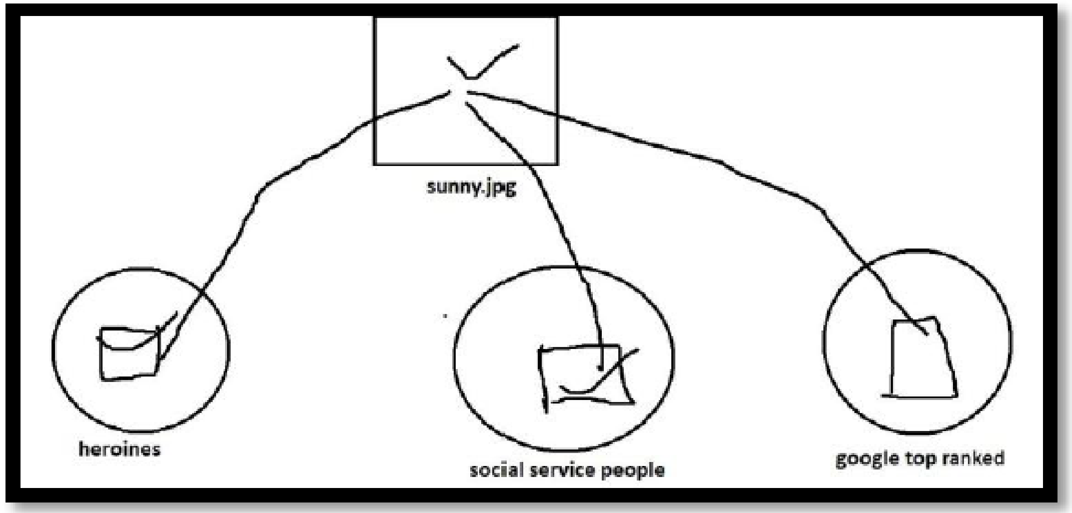
Symlink

- **There are 2 types of link files**

    1) Hard Link files

    2) Soft Link files

1) Hard Link Files:



It is just another name of the same exact file.
We can create hard link file by using ln command.
ln originalfile hardlinkfile

Eg: ln file1.txt file2.txt
Here file1.txt is original file and file2.txt is hard link file.

Important conclusions about hard link file:

1) Both original file and hardlink file have same inode number, same size, same timestamp.
2) If we delete original file, then there is no effect on hardlink file.

2) Soft Link File:

- A softlink is a pointer to another file. It is just like windows shortcut.
- It is also known as symbolic link.
- We can create soft link file by using ln command but with -s option.
- ln -s originalfile softlinkfile

- Eg: ln -s file1.txt   file2.txt
- Here file1.txt is original file and file2.txt is link file.

Important conclusions about softlink file:

1) Original file and softlink file have different inode numbers, different file sizes and different timestamps.
2) Usually softlink file has smaller file size than original file size.
3) If we delete original file then softlink files will become useless.

Downloading data

```
wget <url>
curl <url>
```

Word count

Count word,

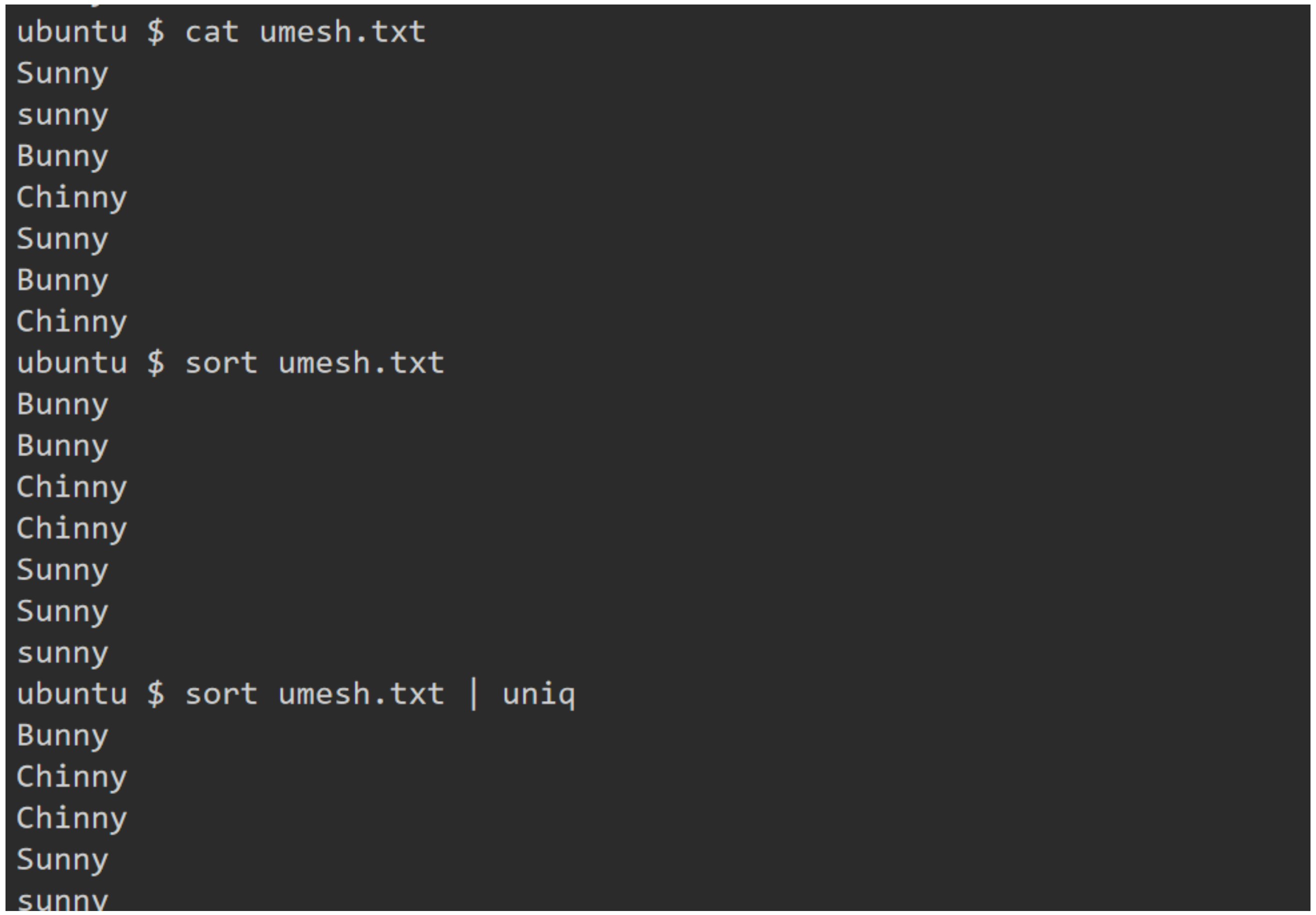
Count line,

Count a letter

Sorting containt of file

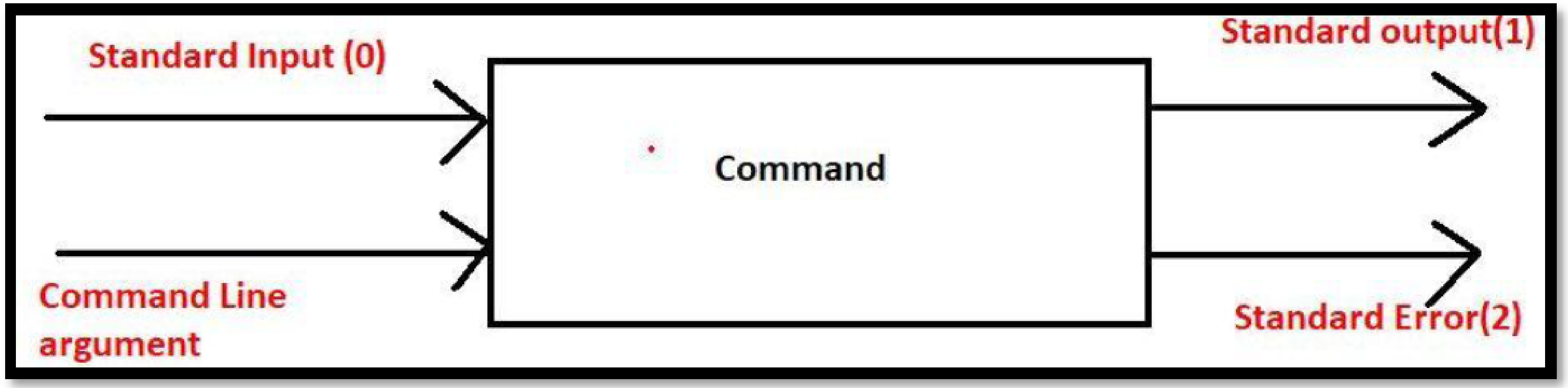
1 9 10 5 3 4 2 6 8 7 > 1 2 3 4 5 6 7 8 9 10

Removing duplicate by uniq command



```
ubuntu $ cat umesh.txt
Sunny
sunny
Bunny
Chinny
Sunny
Bunny
Chinny
ubuntu $ sort umesh.txt
Bunny
Bunny
Chinny
Chinny
Sunny
Sunny
sunny
ubuntu $ sort umesh.txt | uniq
Bunny
Chinny
Chinny
Sunny
sunny
```

Input and Output of Commands and Redirection



Commands can take input, perform required operation and produces some output. While executing command if anything goes wrong then we will get error message.

Command can take input either from standard Input or from command line arguments. Command will produce results to either Standard output or Standard Error.

Standard Input, Standard Output and Standard Error are Data Streams and can flow from one place to another place. Hence redirection and piping are possible.

Command Line arguments are static and these are not streams. Hence redirection and piping concepts are not applicable to command line arguments.

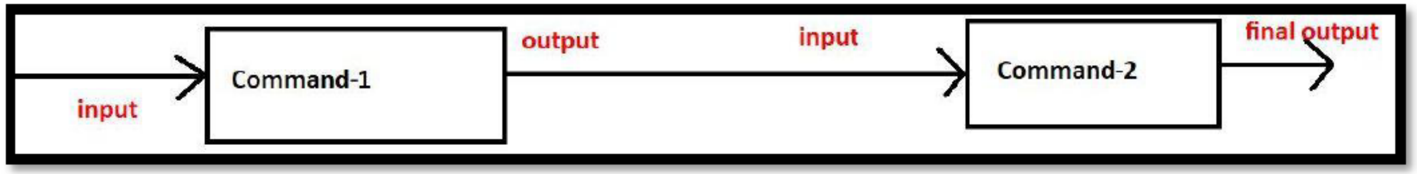
Standard input/output/Error

These data streams are associated with some numbers.
Standard Input associated with 0.
Standard Output associated with 1.
Standard Error associated with 2.

Bydefault Standard input connected with keyboard, Standard output and Standard Error connected with Terminal. But we can redirect.

piping

Sometimes we can use output of one command as input to another command. This concept is called piping.
By using piping, multiple commands will work together to fulfill our requirement.



We can implement piping by using vertical bar (|).

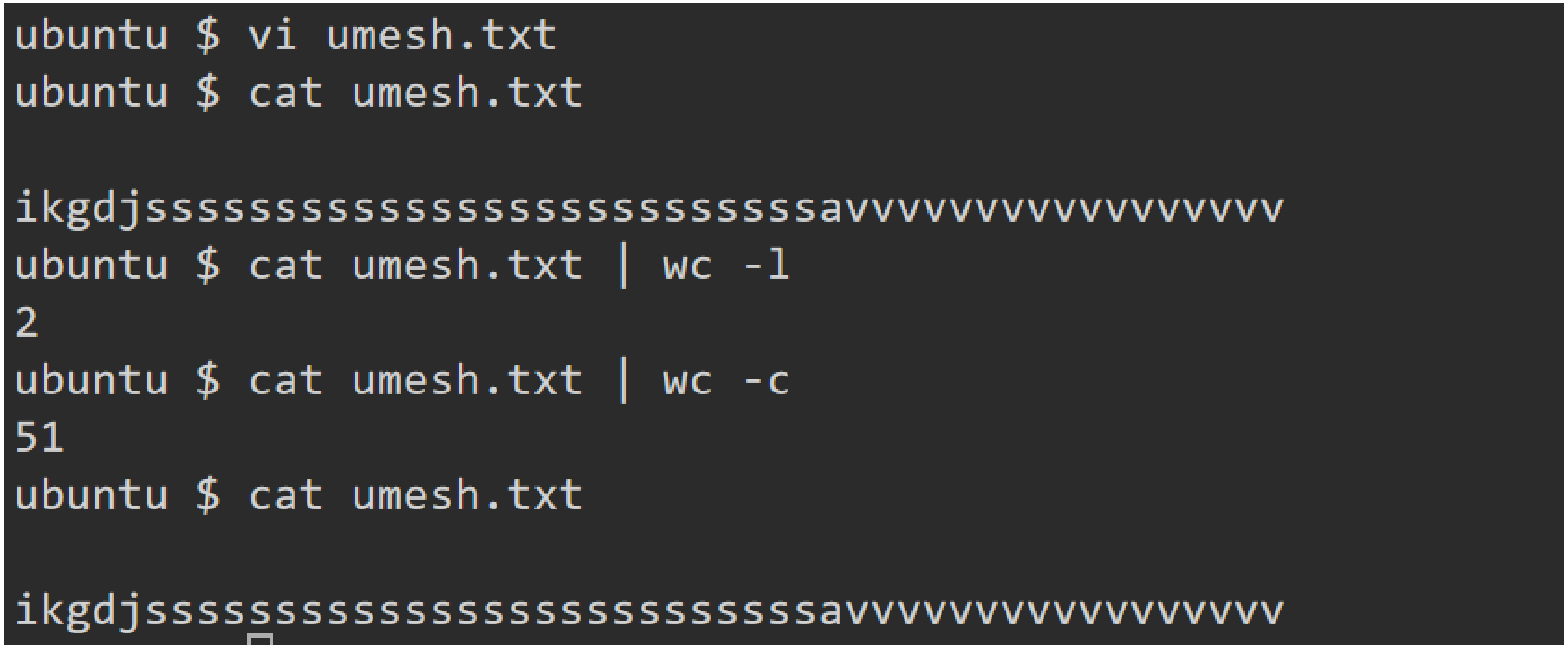
```
$ ls -l /etc | wc
   215   1940  11872
```

First ls got executed and the output of this command will become input to wc command.

<u>Eg 2:</u> `$ ls -l /etc | more`

<u>Eg 3:</u>
```
$ ls -l /etc | wc |wc -l
```
The output is: 1

Piping example



```
ubuntu $ vi umesh.txt
ubuntu $ cat umesh.txt

ikgdjsssssssssssssssssssssssssavvvvvvvvvvvvvvvvv
ubuntu $ cat umesh.txt | wc -l
2
ubuntu $ cat umesh.txt | wc -c
51
ubuntu $ cat umesh.txt

ikgdjsssssssssssssssssssssssssavvvvvvvvvvvvvvvvv
```

Running multiple command togather

We can execute multiple independent commands in a single line by using the following two ways

1st Way: By using semicolon (;)

```
cmd1;cmd2;cmd3;.....;cmdn
```

First cmd1 will be executed and then cmd2 followed by rest of the commands.
If any command fails in the middle, still rest of the commands will be executed.

2nd Way: By using &&

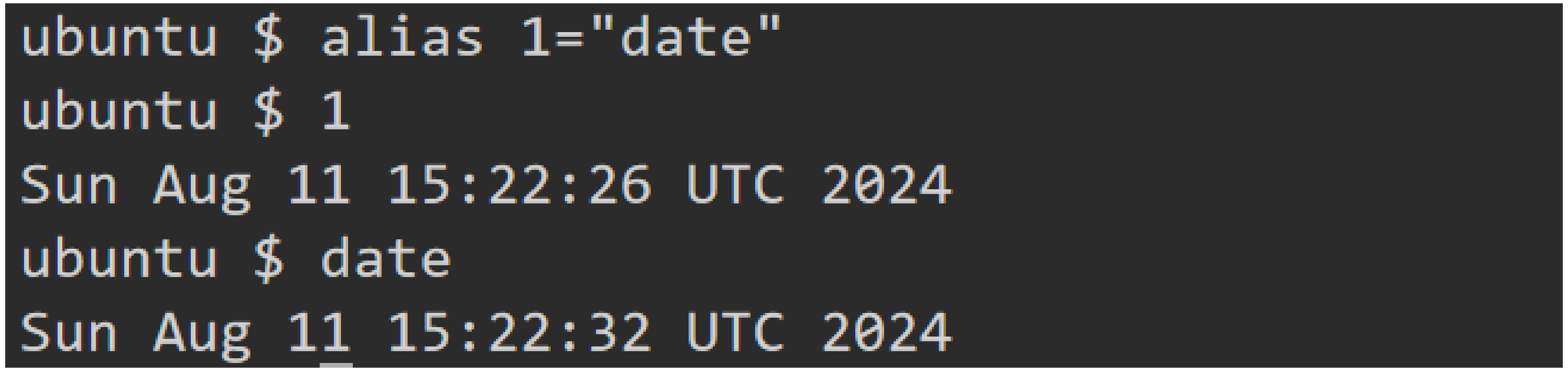
```
cmd1 && cmd2 && cmd3 &&..... && cmdn
```

Regular expression

- If we want to represent a group of strings according to a particular pattern, then we should go for regular expressions.
- By using wildcard characters, we can build regular expressions.
- A wildcard character can be used as a substitute for required sequence of characters in the regular expression

1) → Represents zero or more characters
2) ? → Represents only one character
3) []→ Range of characters
4) [abc] → Either a or b or c
5) [!abc] →Any character except a,b and c
6) [a-z] → Any lower case alphabet symbol
7) [A-Z] → Any upper case alphabet symbol
8) [a-zA-Z] → Any alphabet symbol
9) [0-9] → Any digit from 0 to 9
10) [a-zA-Z0-9] → Any alphanumeric character
11) [!a-zA-Z0-9] → Except alpha numeric character (i.e special symbol)
12) [[:lower:]] → Any lower case alphabet symbol
13) [[:upper:]] → Any upper case alphabet symbol
14) [[:alpha:]] → Any alphabet symbol
15) [[:digit:]] → Any digit from 0 to 9
16) [[:alnum:]] → Any alpha numeric character
17) [![:digit:]] → Any character except digit
18) {} → List of files with comma separator

Command aliasing

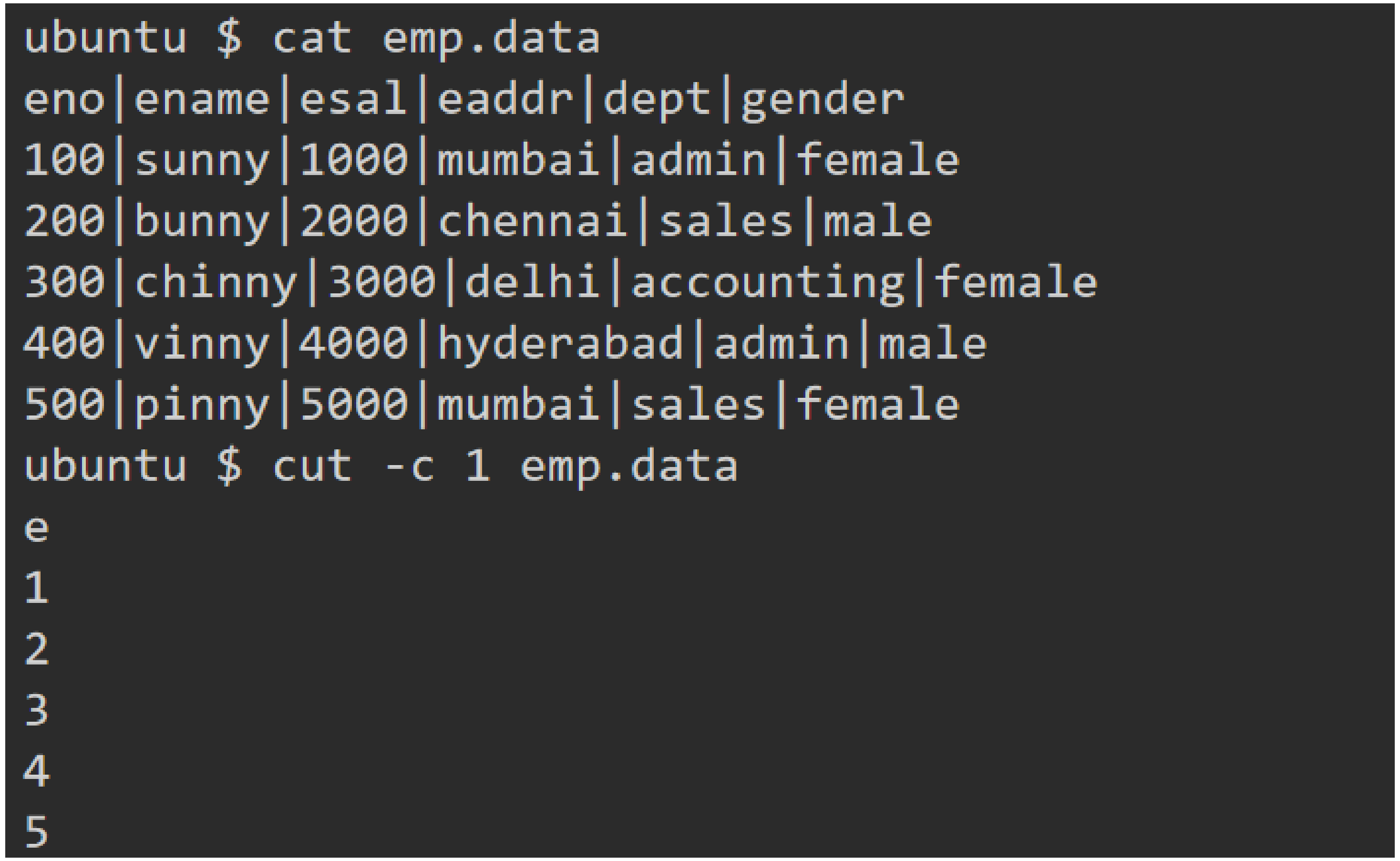


```
ubuntu $ alias 1="date"
ubuntu $ 1
Sun Aug 11 15:22:26 UTC 2024
ubuntu $ date
Sun Aug 11 15:22:32 UTC 2024
```

How to create table data in linux

```
eno|ename|esal|eaddr|dept|gender
100|sunny|1000|mumbai|admin|female
200|bunny|2000|chennai|sales|male
300|chinny|3000|delhi|accounting|female
400|vinny|4000|hyderabad|admin|male
500|pinny|5000|mumbai|sales|female
```

cut



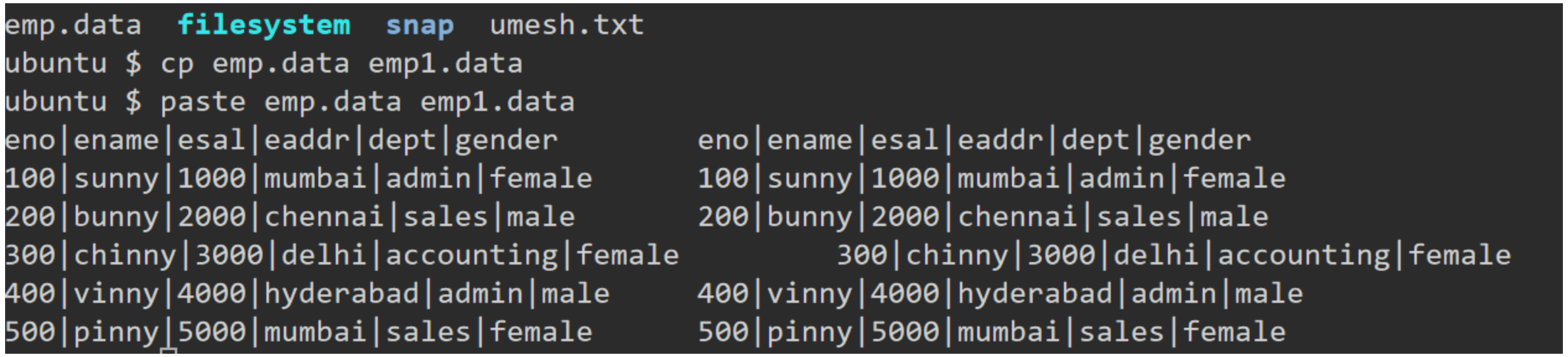
```
ubuntu $ cat emp.data
eno|ename|esal|eaddr|dept|gender
100|sunny|1000|mumbai|admin|female
200|bunny|2000|chennai|sales|male
300|chinny|3000|delhi|accounting|female
400|vinny|4000|hyderabad|admin|male
500|pinny|5000|mumbai|sales|female
ubuntu $ cut -c 1 emp.data
e
1
2
3
4
5
```

paste

paste Command:

**We can use paste command to join two or more files horizontally by using some delimiter.
Default delimiter is tab.**

Syntax: $ paste file1 file2 ...



```
emp.data  filesystem  snap  umesh.txt
ubuntu $ cp emp.data emp1.data
ubuntu $ paste emp.data emp1.data
eno|ename|esal|eaddr|dept|gender        eno|ename|esal|eaddr|dept|gender
100|sunny|1000|mumbai|admin|female      100|sunny|1000|mumbai|admin|female
200|bunny|2000|chennai|sales|male       200|bunny|2000|chennai|sales|male
300|chinny|3000|delhi|accounting|female         300|chinny|3000|delhi|accounting|female
400|vinny|4000|hyderabad|admin|male     400|vinny|4000|hyderabad|admin|male
500|pinny|5000|mumbai|sales|female      500|pinny|5000|mumbai|sales|female
```

mount command

The mount command in Linux is used to attach a filesystem to a directory, making it accessible through that directory.

vi editor

Creating / updating /retrieving /deleting data from file

Sed command

Manipulating the text

compression

Reducing the size

packing

Directory to file conversion.