	Carátula para entrega de prácticas
Facultad de Ingeniería	Laboratorio de docencia

Laboratorio de Computación Salas A y B

Profesor(a): César Fabián Domínguez Velasco

Asignatura: Fundamentos de programación

Grupo: 8

No de practica(s): 2

Integrante(s): Mosqueda Zacatenco Eduardo Daniel

No de lista o brigada: 27

Semestre: 2025-2

Fecha de entrega: 27/02/2025

Observaciones:

Calificación:

Práctica 2 GNU/Linux

Objetivo:

El alumno comprenderá que el sistema operativo es una parte fundamental de un sistema de cómputo. Asimismo, explorará un sistema operativo GNU/Linux para familiarizarse y utilizar sus comandos básicos.

Actividades:

- Iniciar sesión en un sistema operativo **GNU/Linux** y abrir una **"terminal"**.
- Utilizar los comandos básicos para navegar por el sistema de archivos.
- Emplear comandos para manejo de archivos.

Introducción

Hoy en día, las computadoras, portátiles y móviles son herramientas funcionales y casi indispensables en la vida cotidiana, tanto en la escuela, el trabajo o el entretenimiento. Sin embargo, no sería posible utilizarlas sin los sistemas operativos (OS). ¿Qué es un sistema operativo?

"Un programa que gestiona los recursos de una computadora, especialmente la asignación de esos recursos entre otros programas". (Hemmendinger & David, 2025)

Históricamente, las primeras computadoras digitales no contaban con un sistema operativo. Fue hasta la década de 1950 que comenzaron a surgir pequeños *"programas de supervisor que proporcionaban operaciones básicas de entrada y salida (E/S)". (Hemmendinger & David, 2025)*

El gran avance llegó en la década de 1970 con el desarrollo de UNIX, que no solo era más avanzado, sino también accesible al público. Posteriormente, se desarrolló Linux, una versión de UNIX de código abierto.

Con el tiempo, surgieron otros sistemas operativos que conocemos hoy en día, como Windows, Mac OS y GNU/Linux. En este contexto, nos enfocaremos en **GNU/Linux** debido a que es de código abierto, lo que permite modificar su código fuente y acceder al núcleo del sistema operativo (kernel). A diferencia de otros sistemas, Linux es de libre distribución y no requiere pagar licencia, lo que lo convierte en el favorito de los programadores.

Para utilizar Linux se deben tener conocimientos de programación, ya que es un sistema operativo con una interfaz de texto que requiere el uso de comandos para acceder al sistema. Los comandos son órdenes que recibe el shell para ejecutar programas en el sistema operativo, lo que resulta muy útil, ya que todo en GNU/Linux se puede controlar mediante comandos.

"El sistema operativo proporciona un acceso adecuado a sus objetos, las tablas de ubicaciones de disco en un caso y las rutinas para transferir datos a la pantalla en el otro". (Hemmendinger & David, 2025)

Cabe mencionar que la versión de Linux que se utilizará no es tan avanzada, lo que significa que no es una interfaz de texto al 100%, pero sí es adecuada para empezar a practicar.

Los sistemas operativos son esenciales para la computación, ya que de cierta forma le dan vitalidad a la computadora. En la ingeniería, trabajar con sistemas operativos puede ayudar a desempeñarnos en distintas áreas. Por esta razón, el objetivo de esta práctica es aprender los conceptos básicos de GNU/Linux para poder trabajar con sistemas operativos.

Práctica en Linux

En primer lugar, se abrió la terminal para ejecutar los comandos y llevar a cabo la práctica. Para abrir la terminal, se presiona: **[Tecla F4] + Terminal.** Al abrir la terminal, aparecerá una ventana con la consola y un espacio para insertar los comandos.

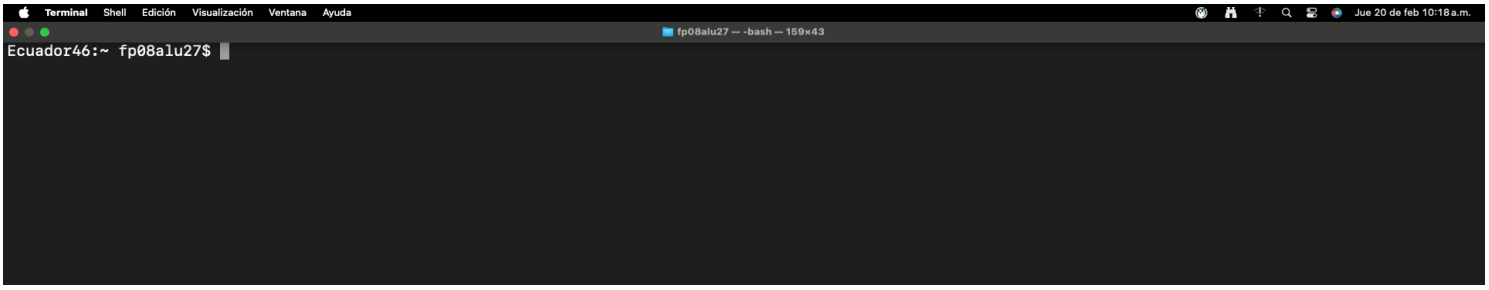

Figura 0: Terminal de GNU/Linux

El primer comando utilizado fue **'cal date'**, el cual sirve para visualizar un calendario. Como se muestra en la imagen, la terminal desplegó un calendario con el día, la fecha y la hora.

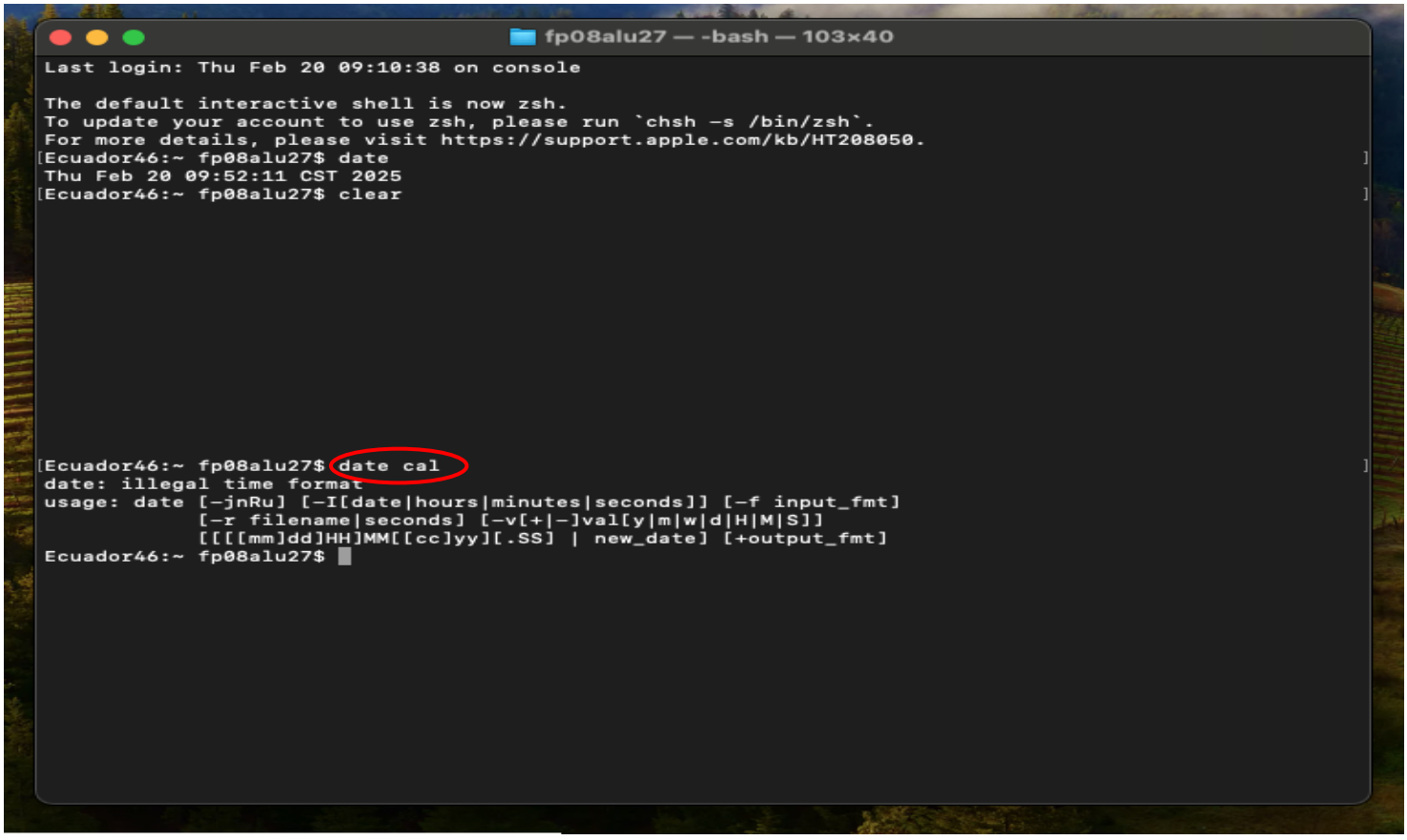

Figura 1: Comando “cal date”

Posteriormente, se utilizó el mismo comando con una variante, **'cal -y'**, que muestra un calendario con todos los meses del año e indica el día en el que nos encontramos.

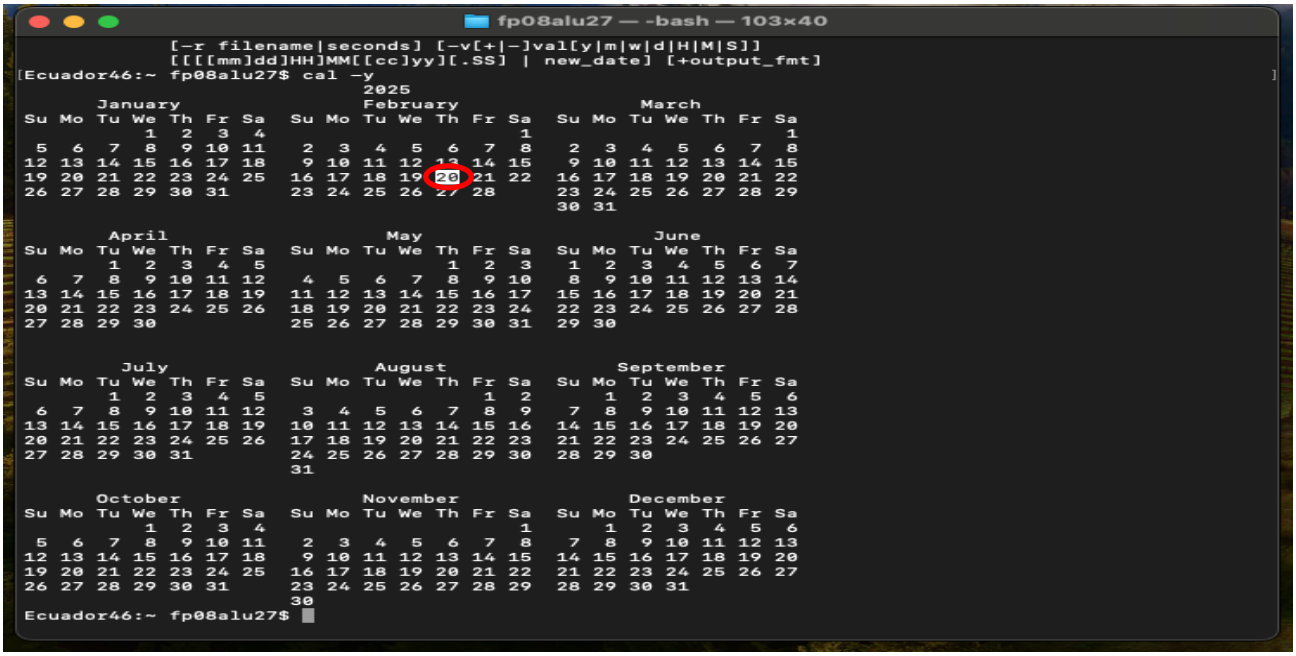

Figura 2: Comando “cal -y”

Además, podemos mostrar únicamente los primeros tres meses del año utilizando el comando **'cal -3'**.

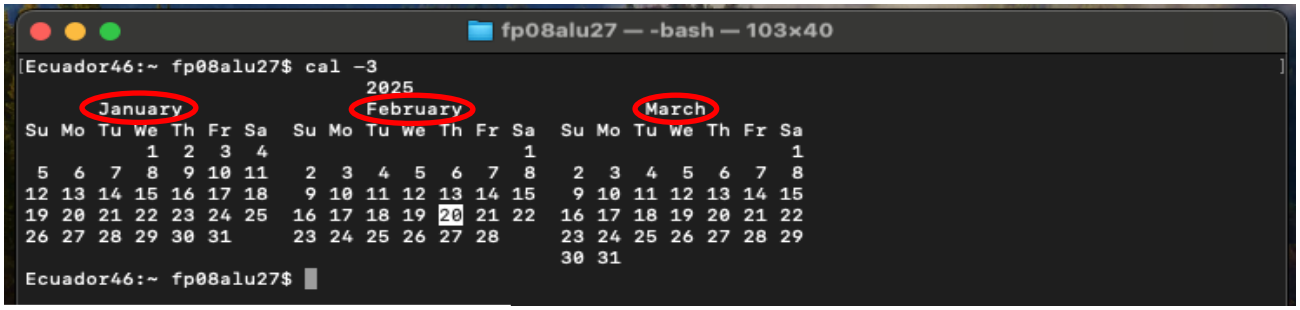

Figura 3: Comando “cal -3”

Durante la práctica, se empleó el comando **'ls'** para listar y visualizar los directorios del sistema.

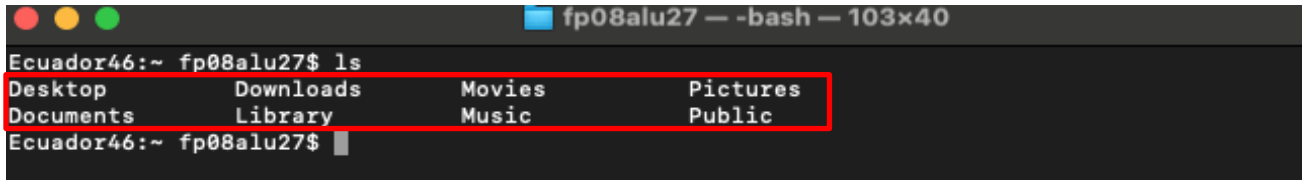

Figura 4: Comando “ls”

Posteriormente, se examinaron las siguientes variantes del comando, **'ls'**.

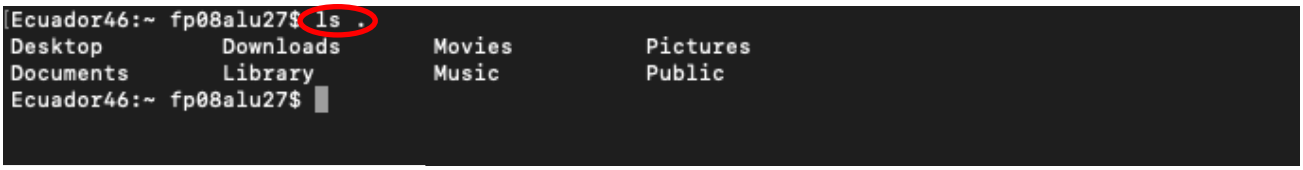

Figura 4.1: Comando “ls.”

Aunque a primera vista no se note ningún cambio, se quiso demostrar que añadir un punto al final no modifica el comportamiento del comando.

Otro ejemplo es el comando **'ls -l'**, que proporciona información detallada acerca de los directorios.

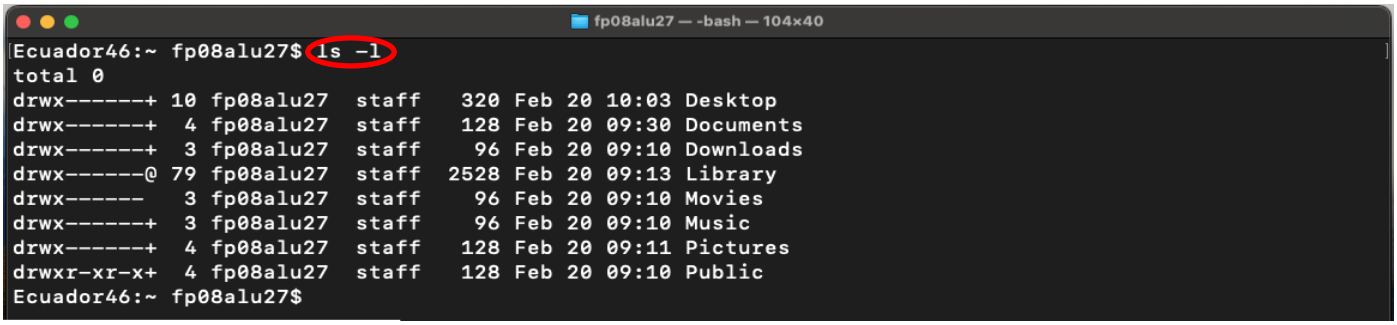

Figura 4.2: Comando "ls- "

El comando **'ls /'** se utiliza para listar el contenido del directorio raíz.

Figura 4.3: Comando "ls /"

El comando **'ls /home'** no se mostró, ya que el directorio estaba vacío; en cambio, el comando **'ls /Applications/'** se utilizó para listar las aplicaciones del sistema.

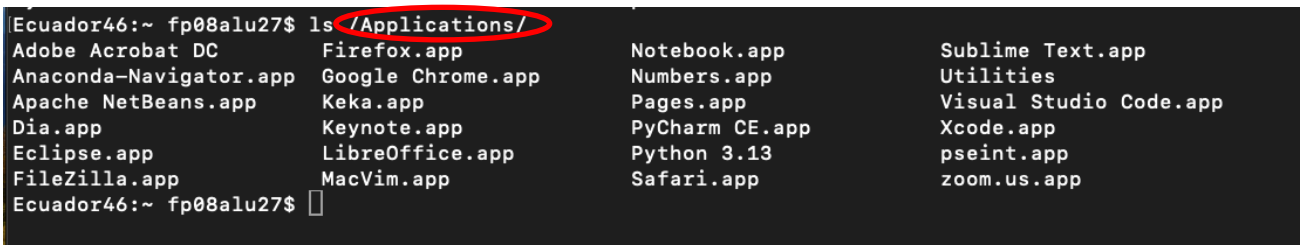

Figura 4.4: Comando "ls /Applications/"

Memorizar varios comandos y conocer la función de cada uno puede resultar un desafío, pero afortunadamente el comando **'man ls'** nos facilita esta tarea. Al ejecutarlo, se despliega un editor de texto con la documentación completa de cada comando. Cabe destacar que, para salir del editor, se debe escribir: **: + Q**, lo que nos regresará a la terminal.

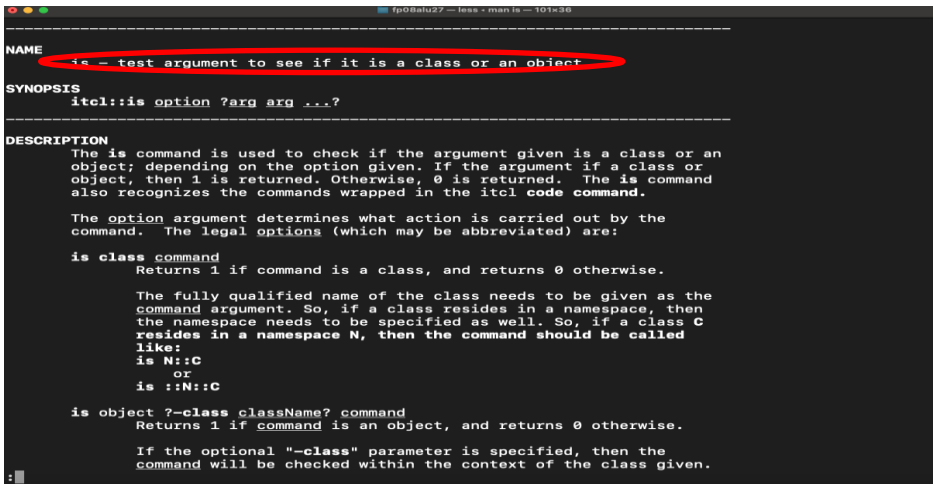

Figura 5: Comando "man ls"

Navegación

Antes de avanzar a otros comandos, primero exploramos la navegación en el sistema de archivos, aprendiendo que existen dos enfoques: la **ruta absoluta** y la **ruta relativa**. La ruta absoluta consiste en especificar la ubicación completa del archivo o directorio desde la raíz. Por ejemplo, en el comando **ls /usr**, se utiliza **ls** para listar el contenido y **/usr** indica la ruta completa, donde **/** representa la raíz y **usr** la carpeta que se desea explorar.

Para utilizar la ruta relativa escribimos **“..”**, se utiliza para referirse al directorio **“padre”**, de esta manera se listan los archivos que dependen de ese directorio.

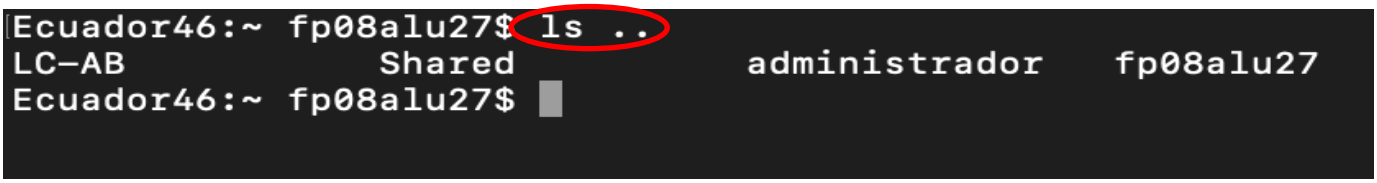

Figura 6: Comando “ls ..”

Para utilizar una ruta relativa, se emplea **“..”**, que representa al directorio padre y permite listar los archivos contenidos en él.

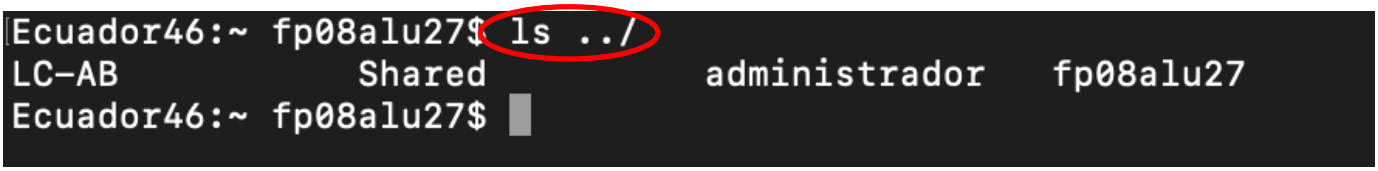

Figura 6.1: Comando “ls ..”

Por ejemplo, al escribir **'ls ../'** le indicamos al sistema que acceda al directorio superior, es decir, al directorio padre de nuestra ubicación actual.

Otro ejemplo es **'ls ../ ../usr'**, con los primeros dos puntos se hace referencia a **home**, con los siguientes al directorio **raíz** y finalmente **usr.**

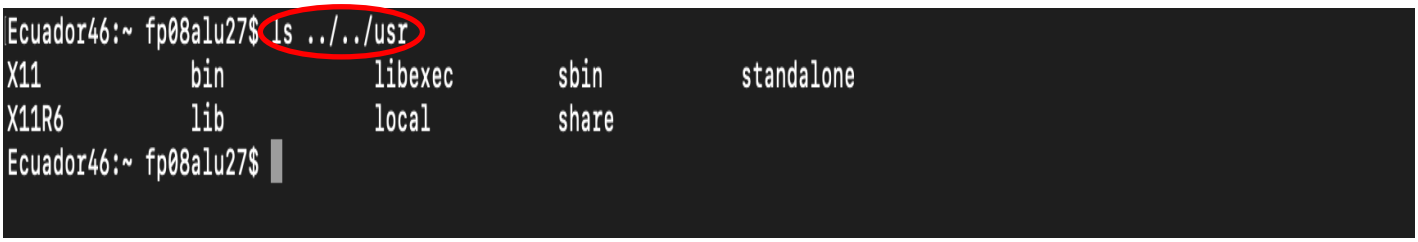

Figura 6.2: Comando “ls ../”

Creación de Archivos

En esta sección se explicó cómo crear archivos utilizando la terminal. Para ello, se emplea el comando **'touch'**, siguiendo la estructura: **touch nombre_archivo.extension**

El archivo recibió el nombre **'Perro.txt'**, donde **'Perro'** es el nombre y **'txt'** la extensión, que indica el tipo de archivo. Posteriormente, se ejecutó el comando **ls** para verificar su creación.

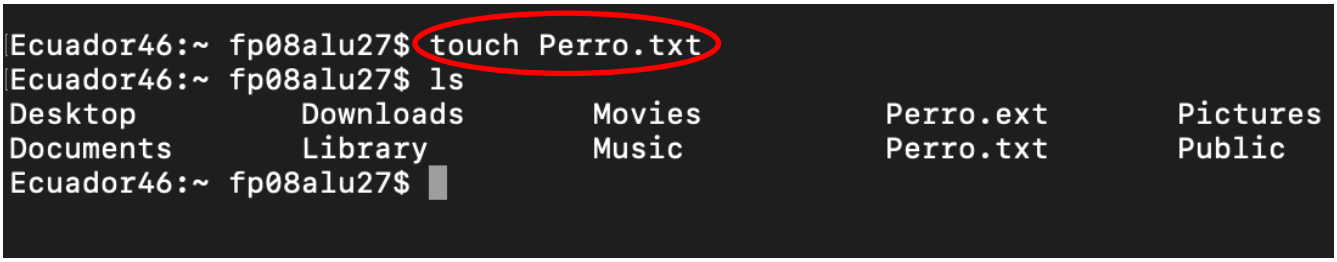

Figura 7: Comando "touch"

Para crear una carpeta o directorio, se utiliza el comando **'mkdir'**, siguiendo la estructura: **mkdir nombre_carpeta.** En este caso, se nombró la carpeta como **'Tareas'** y, posteriormente, se ejecutó el comando **ls** para confirmar que se hubiera creado correctamente.

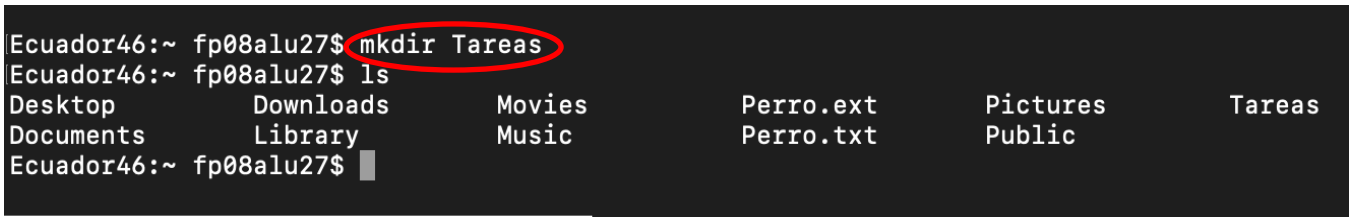

Figura 7.1: Comando "mkdir"

Para conocer la ubicación actual dentro de la terminal, utilizamos el comando **'pwd'**.

Figura 8: Comando "pwd"

Para cambiar de directorio, utilizamos el comando **'cd'** seguido de la ruta del directorio deseado. Por ejemplo, al escribir **'cd tareas/'** accedemos al directorio **'tareas'**.

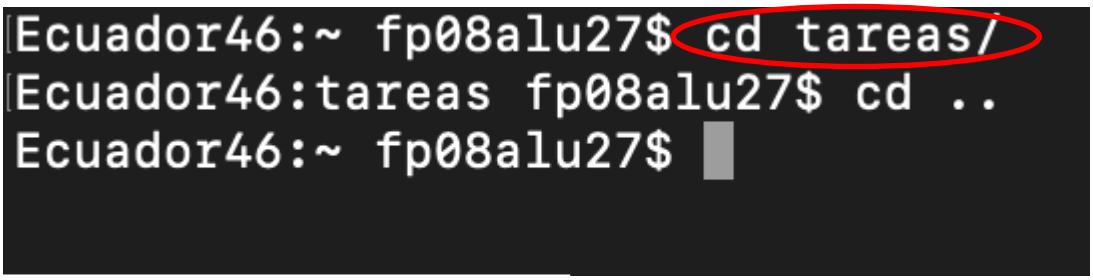

Figura 9: Comando "cd"

Para buscar un elemento dentro del sistema de archivos, utilizamos el comando **'find'** con la siguiente sintaxis: **find [directorio] -name [cadena_buscar]**

Es recomendable especificar tanto el directorio en el que se realizará la búsqueda como el nombre del elemento a buscar. Por ejemplo, podemos indicarle al sistema que busque el directorio **'Tareas'**.

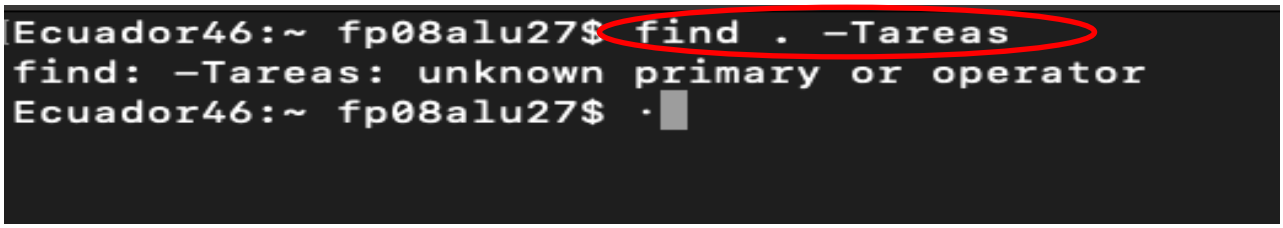

Figura 10: Comando "find"

Para copiar un archivo a otra carpeta, se utiliza el comando **'cp'** siguiendo esta sintaxis: **cp archivo_origen archivo_destino**.

Procedimos a copiar el archivo **Perro.txt** al directorio **Tareas**.

Figura 11: Comando "cp"

El comando cp también permite copiar un archivo y asignarle un nombre diferente. Por ejemplo, al ejecutar **cp Perro.txt Gato.txt** le indicamos al sistema que cree una copia del archivo **Perro.txt** con el nuevo nombre **Gato.txt**

Figura 12: Comando "cp"

Ahora podemos constatar que el archivo **Gato.txt** aparece en el directorio.

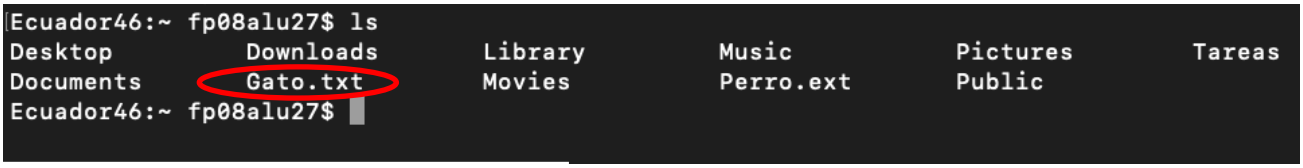

Figura 13: Comando "cp"

Para mover un archivo de una ubicación a otra en el sistema, se utiliza el comando **'mv'** con la siguiente sintaxis: **mv ubicación_origen/archivo ubicación_destino**

Después de ejecutar el comando, el archivo **Perro.txt** ya no aparece en su ubicación original, ya que ha sido trasladado al directorio **Tareas**.

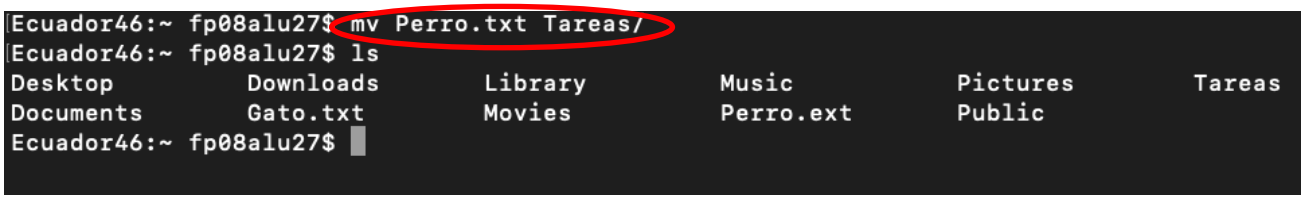

Figura 14: Comando "mv"

Finalmente, para eliminar un elemento del sistema se utiliza el comando **'rm'**, siguiendo la sintaxis: **rm nombre_archivo** o **rm nombre_carpeta**

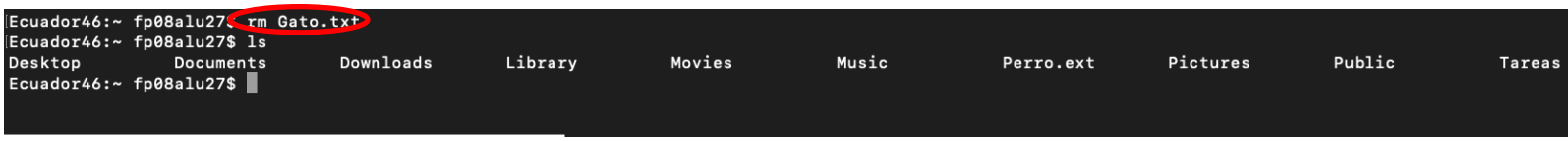

Figura 15: Comando “rm”

"El archivo **Gato.txt** ha sido eliminado, por lo que ya no se muestra en el directorio."

Resultados

- Se inició sesión en un sistema operativo GNU/Linux y se abrió una “terminal”
- Se utilizó los comandos básicos para navegar por el sistema de archivos.
- Se empleó comandos para manejo de archivos.

Conclusión

En la actualidad, los sistemas operativos son indispensables para realizar tareas en las computadoras, ya que sin ellos su uso sería considerablemente más complejo. Tal como se evidenció en la práctica, los comandos facilitan la interacción con el sistema y la emisión de instrucciones, por lo que resulta esencial contar con un conocimiento básico de los mismos.

Aunque cada sistema operativo tiene sus particularidades, GNU/Linux destaca por su versatilidad. Al ser de código abierto y de libre distribución, ha impulsado importantes avances en el campo de la programación. Hoy en día, además de ser ampliamente utilizado en la ingeniería de software, permite incursionar en distintas áreas de otras ingenierías.

Asimismo, GNU/Linux posibilita agilizar y optimizar el trabajo, al mismo tiempo que permite modificar e implementar mejoras en los sistemas desarrollados.

En definitiva, la programación, como cualquier otra disciplina, requiere de tiempo y dedicación. Aunque dominar GNU/Linux demanda práctica constante, el manejo básico de sus comandos constituye una base sólida para lograr un dominio completo del sistema.

Referencias:

- *Facultad de Ingeniería & Área/Departamento: Laboratorio de computación salas A y B. (2025). Manual de prácticas del Laboratorio de Fundamentos de Programación (MADO-17). Facultad de Ingeniería. Recuperado 27 de febrero de 2025, de http://lcp02.fi-b.unam.mx/static/docs/MADO-17_FPv5-1.pdf*
- *Hemmendinger, & David. (2025, 15 febrero). Operating system (OS) | Definition, Examples, & Concepts. Encyclopedia Britannica. https://www.britannica.com/technology/operating-system*