Índice de contenido

Índice de figuras

Índice de tablas

Glosario de términos

Contenido

Escritorios GNOME y KDE

Todos nosotros hemos usado al menos una vez una computadora, pero pocas veces nos hemos preguntado porque esta es tan sencilla de usar. Esta facilidad se debe a los entornos de escritorio que en si son un conjunto de Software (Programas y/o Aplicaciones) que ayuda a que el usuario de una computadora interactúe cómodamente con esta de una forma amigable.

Entre los populares esta KDE, un proyecto internacional de software libre, orientado específicamente a desarrollar un entorno de escritorio e infraestructura de desarrollo para los sistemas operativos GNU/Linux, Mac OSX e incluso, Windows.

KDE es un entorno de escritorio moderno, y que según su propia página [1] ofrece un extraordinario soporte para hasta 75 idiomas diferentes, lo que ofrece a los usuarios un nivel de comprensión del entorno mucho más adaptado a su idioma.

Aun así, este software también tiene competencia siendo comparado normalmente con GNOME (GNU Network Object Model Environment /Entorno de Modelo de Objeto de Red GNU), otro entorno de escritorio e infraestructura de desarrollo nacido en 1997 como alternativa del propio KDE ,para sistemas operativos GNU/Linux, Unix y derivados Unix como BSD o Solaris; compuesto enteramente de software libre.

GNOME está disponible en las principales distribuciones GNU/Linux, incluyendo Fedora, Debian, Ubuntu, Manjaro Linux, Red Hat Enterprise Linux, SUSE Linux Enterprise, CentOS, Oracle Linux, Arch Linux, Gentoo​, SteamOS, entre otras. También, se encuentra disponible en Solaris, un importante sistema operativo UNIX y en Sistemas operativos Unix-like como FreeBSD.

Esta rivalidad entre ambos softwares ha variado con el tiempo al grado de que muchas veces se ha rumorado dentro de sus comunidades de una colaboración de ambos para crear un fusión de entornos de escritorio pero ninguno se ha concretado.

Entornos Command Line Interface y Graphical User Interface

Una computadora no sirve de mucho si no se puede comunicar uno con ella, por eso es que desde el inicio de estas se ha buscado diferentes formas de poder lograr esto, siendo una de las más exitosos las interfaces de línea de comando o interfaces de línea de órdenes (en inglés, command-line interfaces, CLI). Estos son métodos que permite a los usuarios dar instrucciones a algún programa informático por medio de una línea de texto simple. No debe confundirse los conceptos de CLI, shell y emulador de terminal, porque no son lo mismo ya que un CLI es un método mientras que un shell y emulador de terminal son programas informáticos.

Las CLI pueden emplearse interactivamente, escribiendo instrucciones en alguna especie de entrada de texto, o pueden utilizarse de una forma mucho más automatizada (archivo batch), leyendo órdenes desde un archivo de scripts.

Esta interfaz existe casi desde los comienzos de la computación, superada en antigüedad solo por las tarjetas perforadas y mecanismos similares. Existen para diversos programas y sistemas operativos, para diverso hardware, y con distinta funcionalidad.

Por ejemplo, las CLI son parte fundamental de los shells o emuladores de terminal. Aparecen en todas las interfaces de escritorio (GNOME, KDE, Microsoft Windows) como un método para ejecutar aplicaciones rápidamente. También se utilizan en aplicaciones cliente-servidor, en gestores de bases de datos, en clientes FTP, etc. Las CLI son además un elemento fundamental de aplicaciones de ingeniería tan importantes como MATLAB y AutoCAD.

Otra forma que también ha tenido mucho éxito en la forma en la cual se comunica los usuarios con la maquina son a aquellas que apelan más a la parte gráfica. Esta orientación tiene sentido ya que al ser el humano una criatura muy visual, este puede entender mejor como proceder en diferentes situaciones si hay un apoyo grafico que le ayude, además de que una buena selección de diseños y colores puede hacer que el usuario pueda comunicarse mejor con la máquina.

Estos programas son conocidos como interfaces gráficas de usuario, conocidas también como GUI (del inglés graphical user interfaces), que cumplen con lo mencionado anteriormente, siendo un programa informático, que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Habitualmente las acciones se realizan mediante manipulación directa, para facilitar la interacción del usuario con la computadora. Surge como evolución de las interfaces de línea de comandos que se usaban para operar los primeros sistemas operativos y es pieza fundamental en un entorno gráfico. Como ejemplos de interfaz gráfica de usuario, cabe citar los entornos de escritorio Windows, el X-Window de GNU/Linux o el de Mac OS X, Aqua.

En el contexto del proceso de interacción persona-computadora, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

Terminal de Linux

Como todos los sistemas operativos basados en Unix , Linux disponen de un intérprete de comandos u órdenes (conocido como terminal, consola,shell o bash ) que hace de interfaz entre el usuario y el propio sistema operativo. La terminal o consola es una herramienta poderosa capaz de realizar tareas de forma más rápida que mediante la interfaz gráfica del sistema operativo.

Si bien es cierto, que para un usuario normal no tiene mucha relevancia el aprenderlo, para un usuario que quiere convertirse en un desarrollador o administrador de sistemas operativos cobra gran importancia debido a que va a tener que utilizar esta herramienta constantemente, ya sea para navegar por el sistema/directorios de forma rápida y sencilla, iniciar o detener procesos, instalar librerías, etc.

Un terminal es una forma de acceder al sistema sin utilizar la interfaz gráfica, es decir, realizar todo tipo de tareas en formato texto. La forma de utilizar el sistema de este modo es mediante órdenes.

El terminal muestra en pantalla un indicador de línea de órdenes (en inglés se utiliza la palabra prompt que literalmente traduciría "prontuario" pero se puede definir como ayuda visual o palabra que ayuda) esperando que el usuario introduzca una orden. Este indicador finaliza generalmente por un caracter $, cuando eres un usuario normal, o # cuando se es un súper usuario (administrador).

Para acceder a una terminal se puede hacer de dos formas, una es con una aplicación como el terminal de GNOME, xterm o konsole de KDE, que son emuladores de la terminal dentro de una interfaz visual. Otra forma es salirse del entorno gráfico y acceder a un entorno completamente en modo texto, algo así como entrar en sólo símbolo de sistema en Windows 98.

Tomando como base a Ellen Seiver [2] cada vez que se ejecuta una orden, el sistema operativo le abre automáticamente tres interfaces (en los sistemas operativos tipo UNIX se utiliza el término archivo): la entrada estándar, la salida estándar y el error estándar.

* La entrada estándar (stdin) se refiere al archivo por el que una orden recibe su entrada (por defecto, es el teclado).
* La salida estándar (stdout) se refiere al archivo por el que una orden presenta sus resultados (por defecto, es la pantalla o más concretamente la ventana en la que se está ejecutando el intérprete de órdenes).
* El error estándar (stderr) se refiere al archivo por el que una orden presenta los mensajes que va generando cuando ocurre un error (por defecto, también es la pantalla).

Estos archivos se crean en el directorio /dev.

Tipos de usuario en Linux

Todo sistema que este pensado para tener más de un usuario (multiusuario) tiene que darles diferentes características y capacidades a estos ya que si no se hiciera estos podrían traer unenorme problema de seguridad. Dentro de Linux se puede identificar tres tipos de usuarios en Linux:

* Usuario ROOT: También llamado superusuario o administrador. Es el único usuario con privilegios sobre el control total del sistema, y por tanto, el único que tiene derecho a administrar las cuentas de usuario del sistema.
* Usuarios normales o finales: Son los usuarios habituales del sistema, que utilizarán los recursos de éste. Cada usuario sólo podrá personalizar su entorno de trabajo.
* Usuarios especiales o de sistema: Son incorporados por el propio sistema, y se encargan de administrar los demonios del mismo. Estos usuarios no pueden iniciar sesión en el sistema ni tener un shell donde trabajar, por tanto no tienen contraseña asignada. Ejemplos son bin, daemon, adm, lp, sync, shutdown, mail, operator, squid, apache, etc.

Linux permite además la creación de grupos de usuarios, para poder establecer permisos y restricciones por grupos en lugar de por usuarios individuales.

Por defecto, cada usuario pertenecerá a un grupo que se crea de forma automática con su mismo nombre de usuario. Será el usuario root quienque pueda asignar los usuarios a los distintos grupos del sistema.

En el sistema existen una serie de grupos predefinidos por defecto. Cada usuario posee un directorio personal o directorio HOME en el que se almacena la configuración de su entorno o perfil así como los datos personales de dicho usuario. Por defecto, a ese directorio sólo tendrá acceso el mismo usuario, a no ser que él específicamente conceda permisos a otros usuarios para poder acceder.

Linux identifica a los usuarios y grupos creados en el sistema por un número, único para cada uno de ellos, y no por el nombre que le pongamos al usuario o al grupo. Lo asigna el sistema automáticamente, aunque el usuario root también lo puede establecer manualmente.

Direccionamiento relativo y absoluto

Cuando se trabaja con comandos es habitual tener que pasar como parámetros archivos o directorios. Para indicar un archivo o directorio se utiliza la ruta o path, que puede ser absoluta o relativa.

Las rutas relativas indican el camino para encontrar un elemento, pero basándose en el directorio desde el que se ejecuta la orden, osea, desde el directorio en donde se encuentran posicionados. En la imagen 1 se puede observar mejor esto

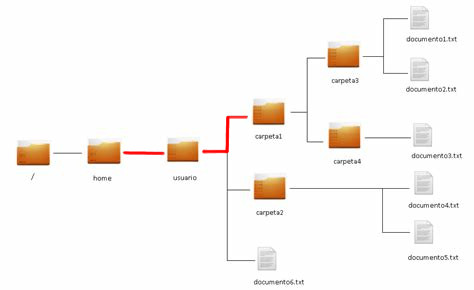


Imagen 1

Si se encuentra en el directorio /home y se quisiera mover al directorio /home/usuario/carpeta1 se debería indicar el camino hasta donde se quiere ir partiendo desde donde se encuentra.

Por otro lado las rutas absolutas son rutas que no dependen del lugar desde donde se este llamando una acción. El sistema de ficheros es una estructura jerárquica que en el caso de Linux tiene una raíz que se indica cuando se pone solamente el carácter barra / . La raíz contiene los directorios principales del sistema que a su vez tendrán subdirectorios en su interior, esta estructura de directorios se corresponde con el estándar de jerarquías de directorios FHS.

Un ejemplo de esta forma de direccionamiento se puede ejemplificar en la imagen 2. Suponiendo que se está en el directorio /home y que se quiere mover hacia el directorio /home/cristian/carpeta1 la forma de hacer esto usando una ruta absoluta seria indicando el camino desde el directorio raiz (/) hasta donde se encuentra el directorio destino

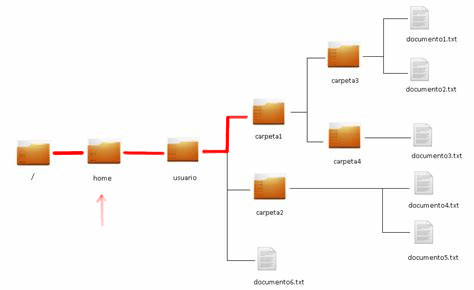
.

Imagen 2

Es muy recomendable usar estas rutas sobre las relativas ya que a pesar de que suelen ser bastante largas, tienen como ventaja que funcionan siempre, independientemente del lugar desde el que se ejecute la orden.

Redireccionamiento

Clasificación de los comandos en Linux

Variables de entorno

Contenido

Códigos y ventas de ejecución

Conclusiones

Bibliografía

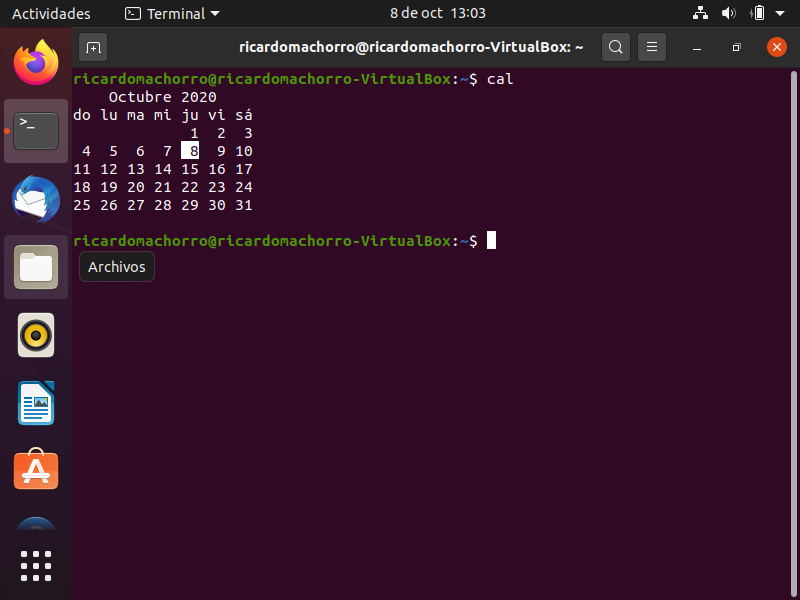
[1] KDE (2019, Julio,15). Que es KDE [Online]. Available: https://userbase.kde.org/What\_is\_KDE/es

[2] Ellen Seiver,Linux in a nutshell.sexta edicipon.Ciudad deMéixco: O’Reilly, 2009

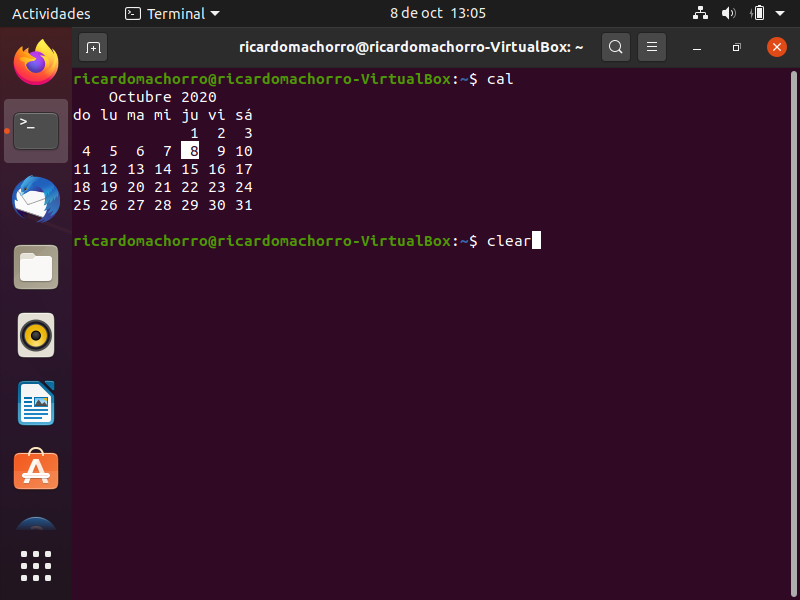
Ricardo Alberto Machorro Vences

|  |  |
| --- | --- |
| comando | funcionamiento |
| cal | Muestra un calendario en la salida standard. |
| clear | Es un comando de linux que borra toda la información del terminal y lo deja como si fuera nuevo. |
| apt | es un programa de gestión de paquetes creado por el proyecto Debian. |
| rm | es un comando de la familia de sistemas operativos Unix usada para eliminar archivos y directorios del sistema de archivos. |
| date | es un comando existente en sistemas Unix y tipo unix que muestra la hora y la fecha del sistema y el administrador también puede cambiarla. |
| ifconfig | es un programa disponible en varias versiones del sistema operativo UNIX, que permite configurar o desplegar numerosos parámetros de las interfaces de red residentes en el núcleo, como la dirección IP (dinámica o estática), o la máscara de red. |
| exit | es un comando que sale del shell actual. salida ejecutivo puede pagar el valor de estado de salida especificado |
| mv | es un comando de Unix usado para mover o renombrar archivos o directorios del sistema de archivos. |
| echo | es un comando para la impresión de un texto en pantalla. |
| df | es un comando que nos informa acerca del espacio total, ocupado y libre en nuestro sistema. |
| ps | es un comando asociado en el sistema operativo UNIX (estandarizado en POSIX y otros) que permite visualizar el estado de un Proceso (informática). |
| more | es un comando para ver (pero no modificar) el contenido de un archivo o comando y visualizarlo por páginas. |
| time | es un comando utilizado para determinar el tiempo de ejecución de una operación específica. |
| du | es un comando estándar de los sistemas operativos de la familia Unix. Se usa para estimar el uso de espacio en disco duro de un archivo, un directorio en particular o de archivos en un sistema de archivos. |
| ps -fea |  |
| less | El comando Less de linux se usa para mostrar el texto en la pantalla del terminal. |
| uname | uname sirve para imprimir información del sistema linux. |
| pstree | es un comando poderoso y útil para mostrar procesos en ejecución en Linux. |
| man | es una herramienta de sistemas Unix que se utiliza para documentar y aprender sobre comandos, archivos, llamadas de sistema, etc., en un sistema operativo tal como GNU/Linux. |
| mkdir | es una orden de los sistemas operativos UNIX, DOS, OS/2 y Microsoft Windows usada para crear un nuevo subdirectorio o carpeta del sistema de archivos |
| w | es un programa de Unix que muestra información sobre los usuarios que han iniciado sesión en el sistema y qué están haciendo. |
| kill -1-9 |  |
| cat | Concatena archivos y los muestra en el salida estándar, es utilizado en el sistema operativo Unix con licencia GNU. |
| pico | es un sencillo editor de texto basado en el editor del sistema de mensajes Pine. Al igual que en Pine, los comandos son desplegados en la parte inferior de la pantalla, y se proporciona ayuda sensible al contexto. Conforme |
| who | Este comando muestra rápidamente los usuarios que actualmente están autentificados en el sistema. |
| trap -l |  |
| fg | Reanuda trabajos suspendidos poniéndolos en foreground (primer plano) o trabajos en background los pasa a primer plano. |
| nano | es un editor de texto para sistemas Unix basado en curses. |
| bash | Es un programa informático cuya función consiste en interpretar órdenes. |
| pwd | se utiliza para imprimir el nombre del directorio actual en una sesión de comandos bajo un sistema operativo Unix o derivado. |
| cd | sirve para cambiar la ruta actual de la terminal y ubicarnos en una carpeta o directorio específico. |
| vi | vi es un editor de texto que maneja en memoria el texto entero de un archivo. Es el editor clásico de UNIX (se encuentra en todas las versiones). |
| wc | es un comando utilizado en el sistema operativo Unix que permite realizar diferentes conteos desde la entrada estándar, ya sea de palabras, caracteres o saltos de líneas. |
| su |  |
| ls | es muy útil para ver los archivos y directorios que tenemos dentro del directorio en el que estamos. |
| apt-get | Instala/informa sobre los paquetes resolviendo las dependencias, los paquetes que instala los consigne de Internet (de /etc/apt/sources.list). |
| sudo | que permite a los usuarios ejecutar programas con los privilegios de seguridad de otro usuario (normalmente el usuario root) de manera segura, convirtiéndose así temporalmente en súper usuario. |
| ls -la |  |

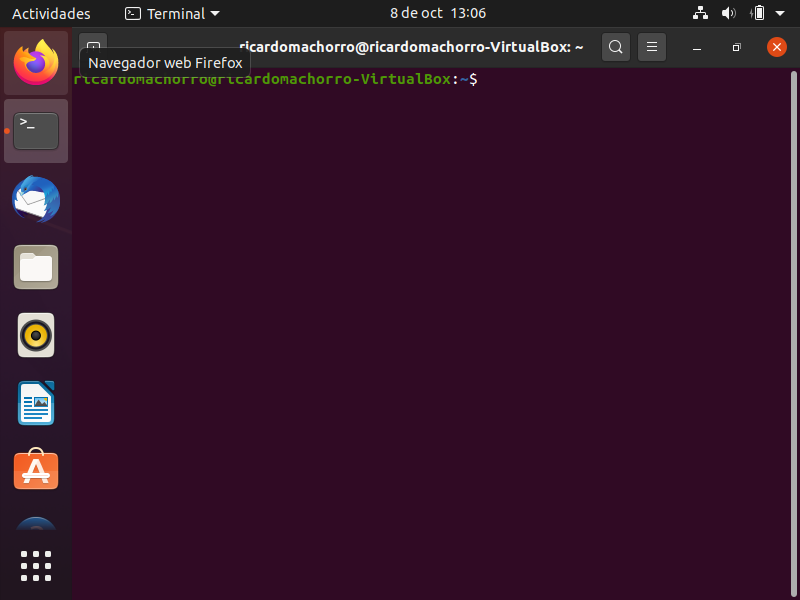
Comando cal



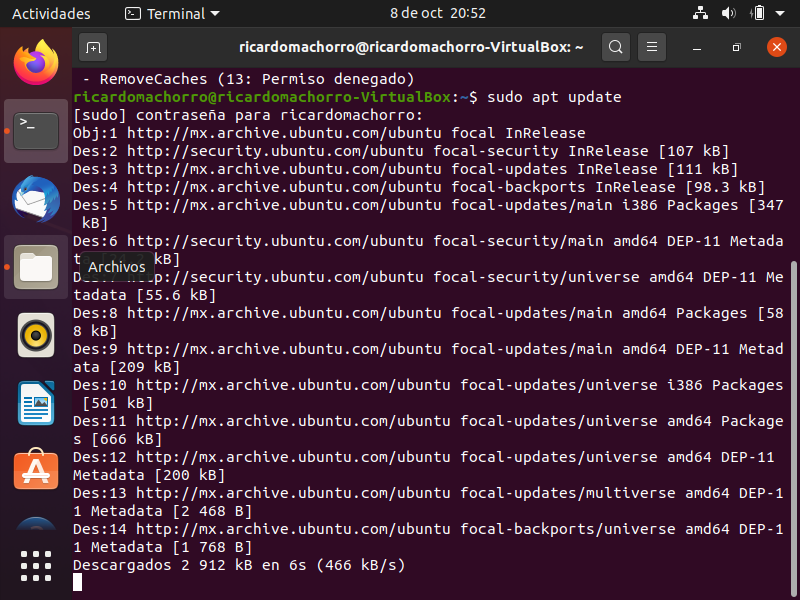
Comando clear antes de ejecutarse



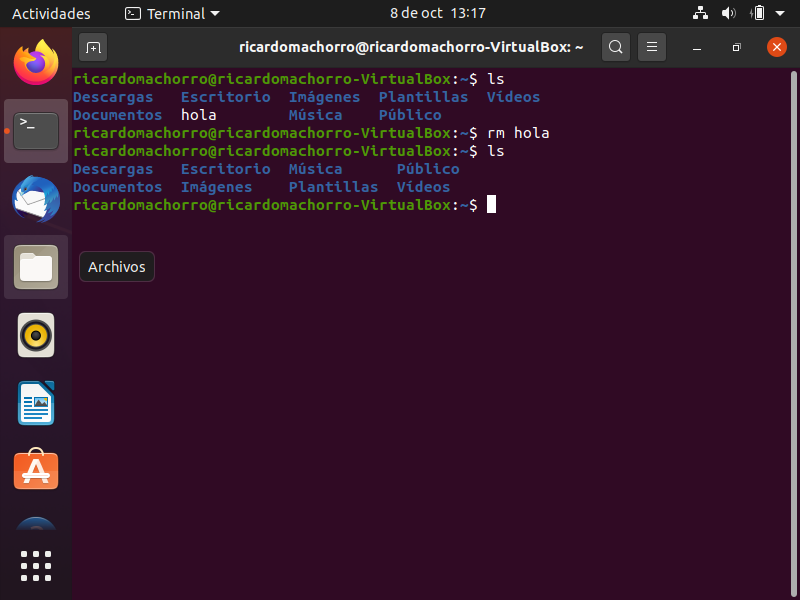
Comando clear después de ejecutarse



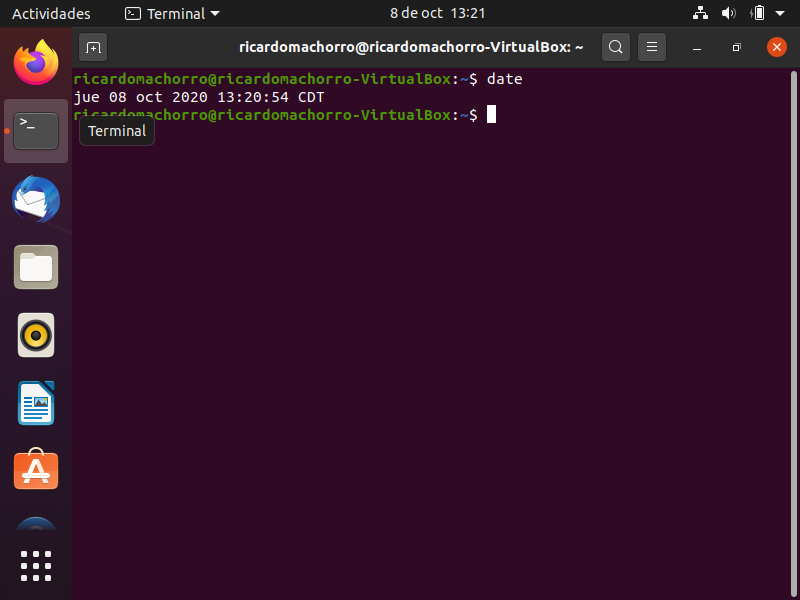
Comando apt



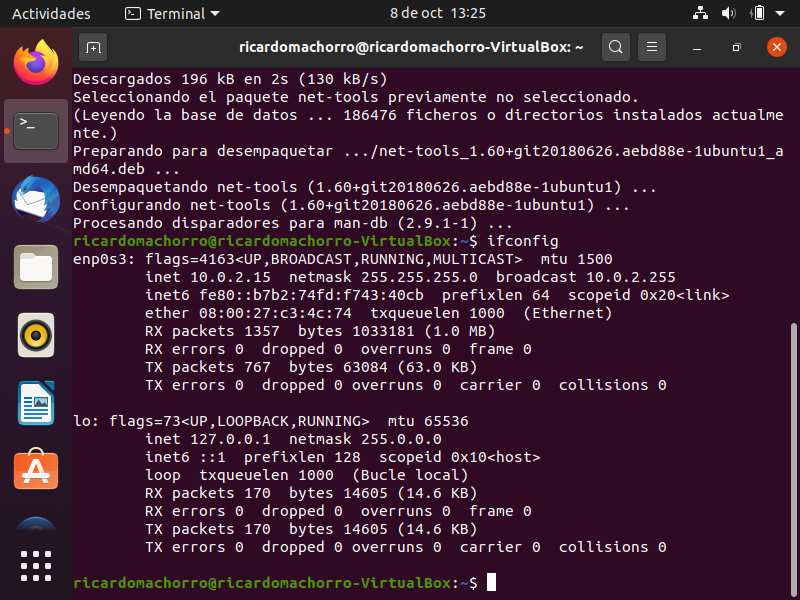
Comando rm



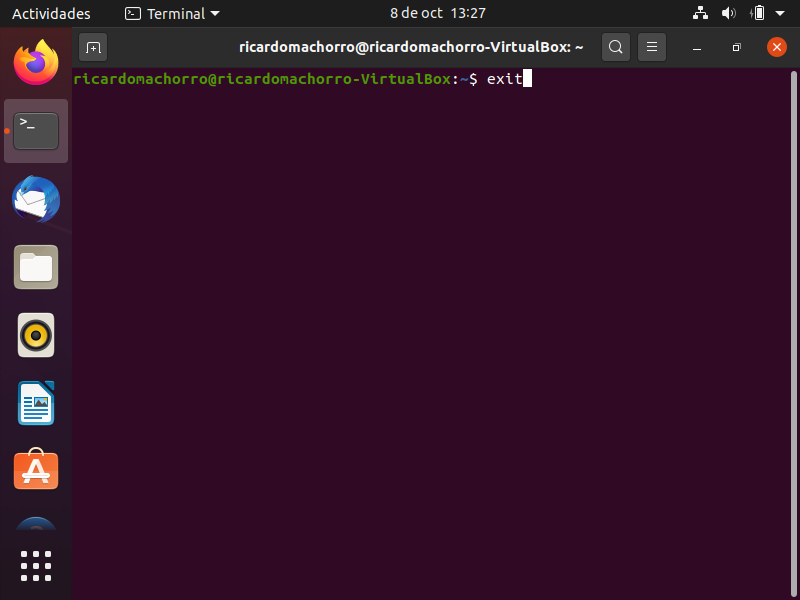
Comando Date



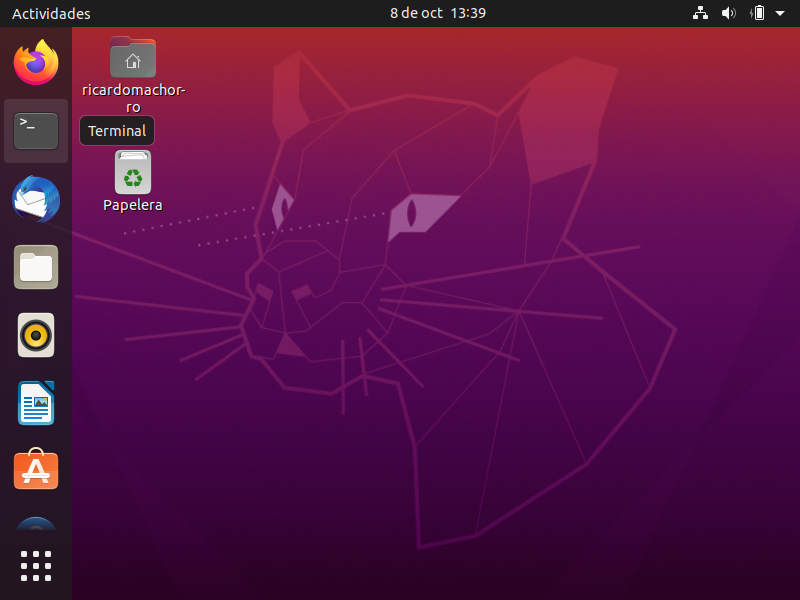
Comando ifconfig



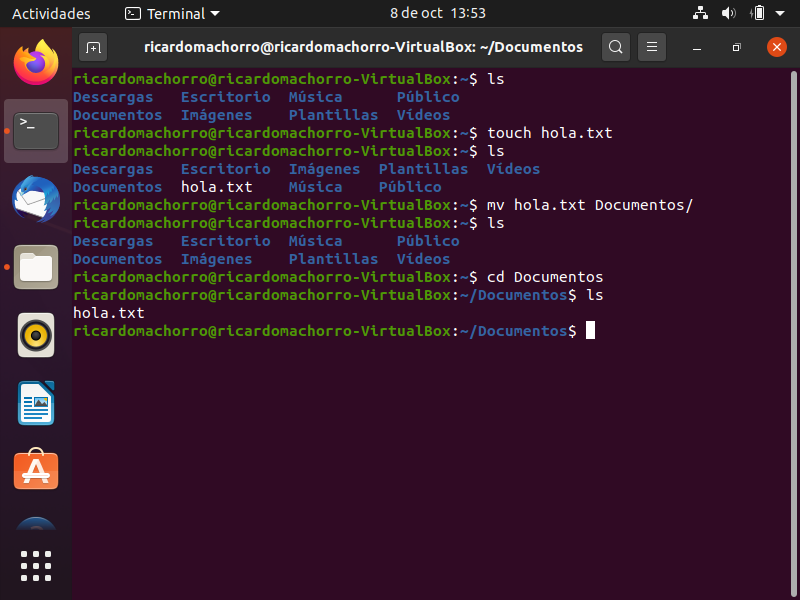
Comando exit antes de ejecutarse



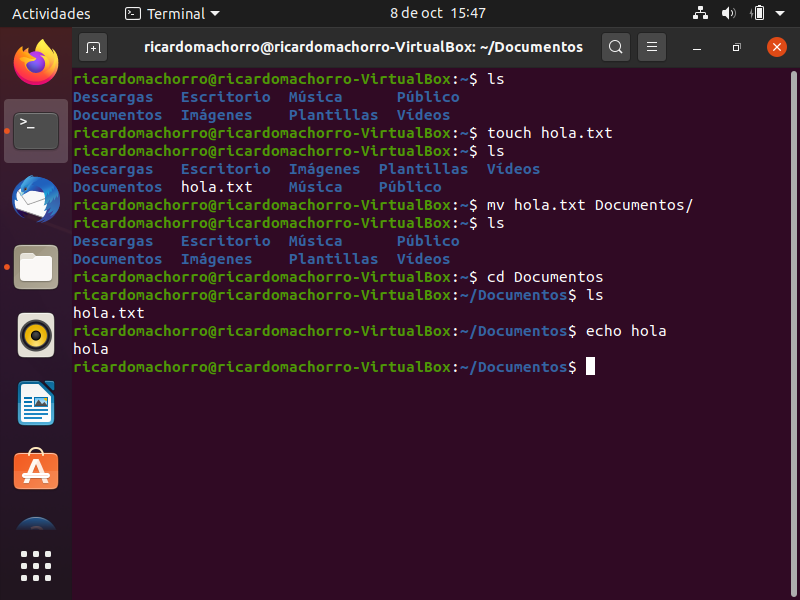
Comando exit después de ejecutarse



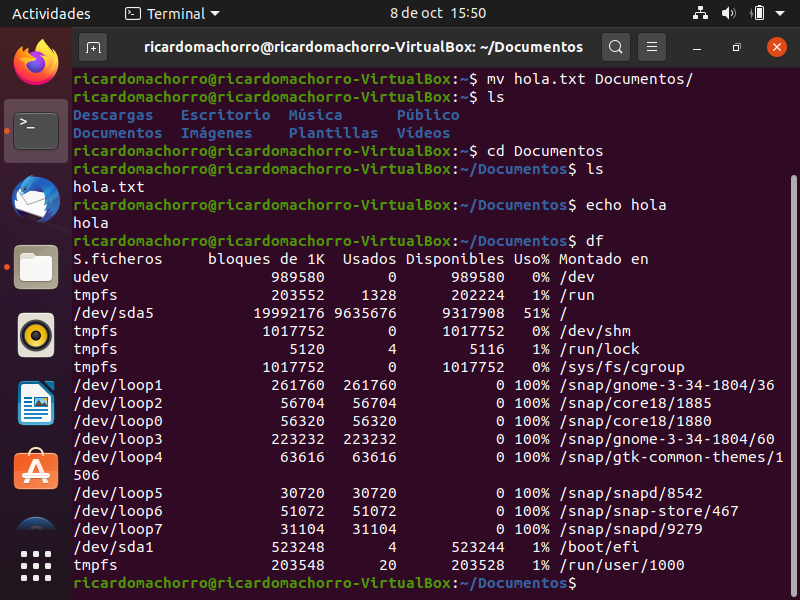
Comando mv



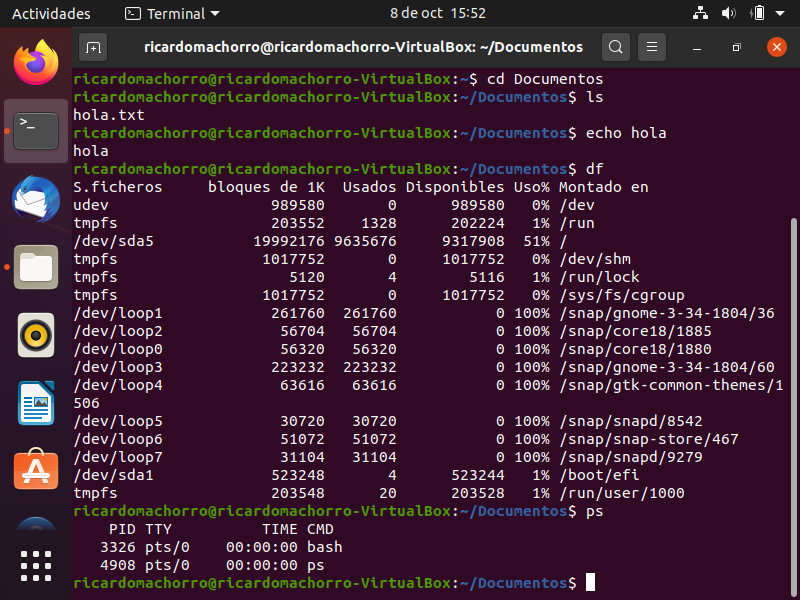
Comando echo



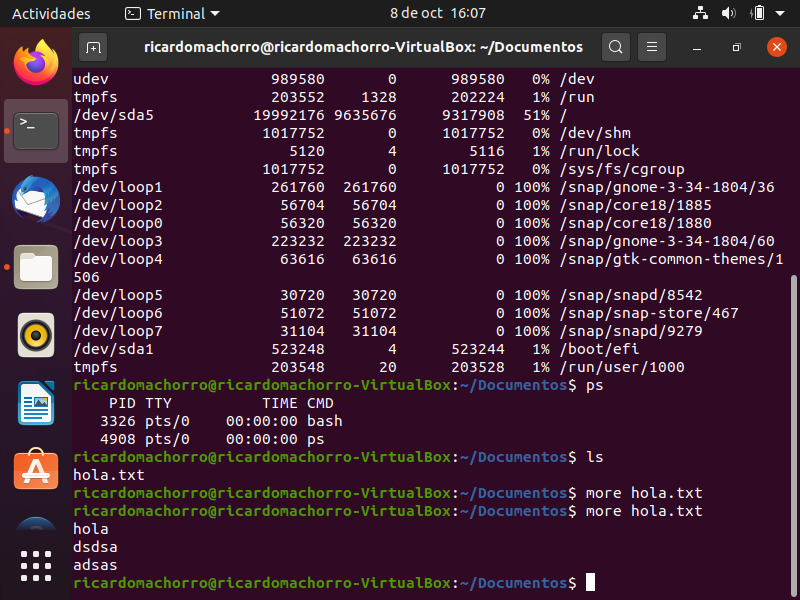
Comando df



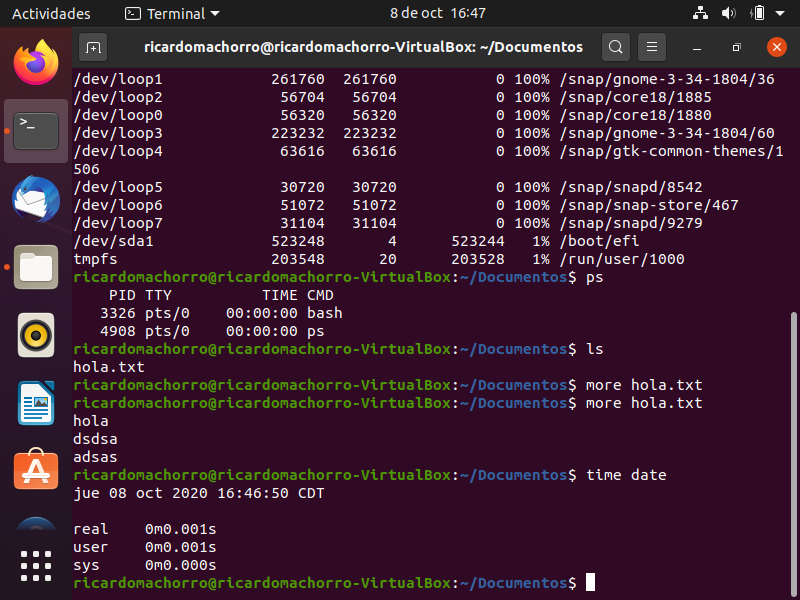
Comando ps



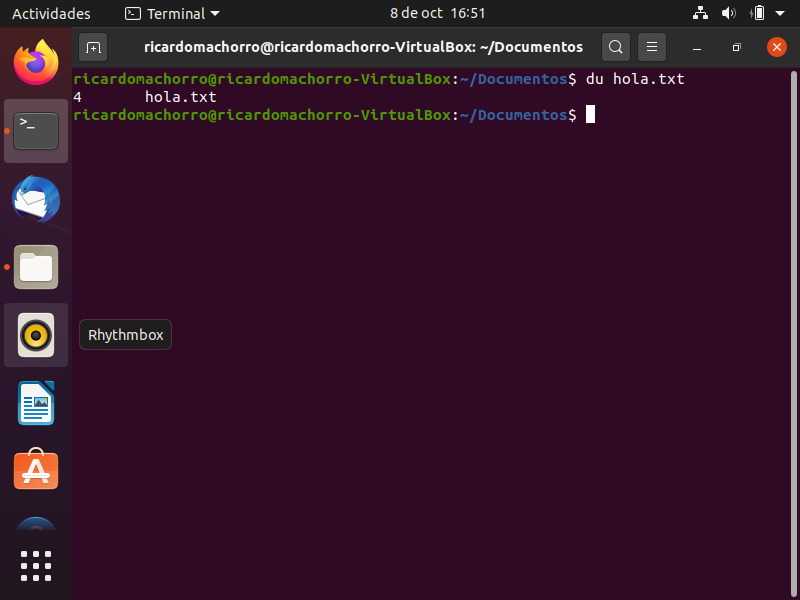
Comando more



Comando time

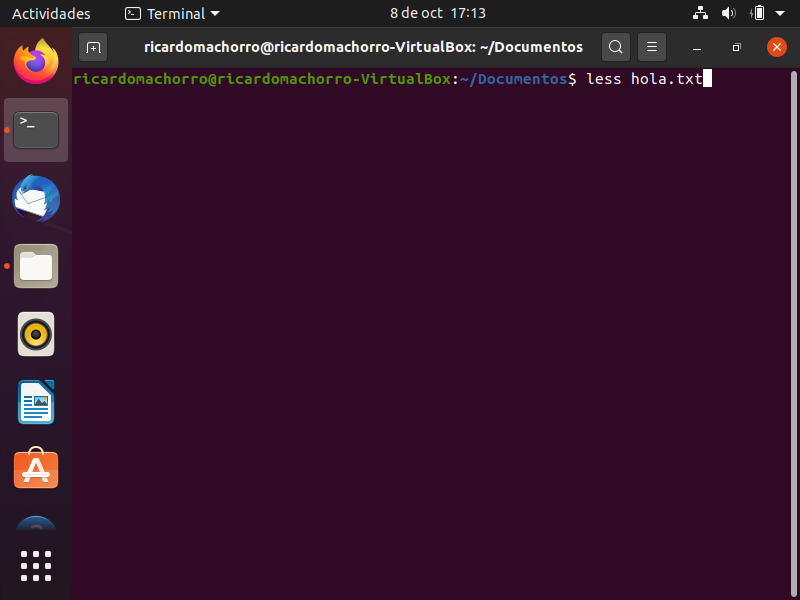


Comando du

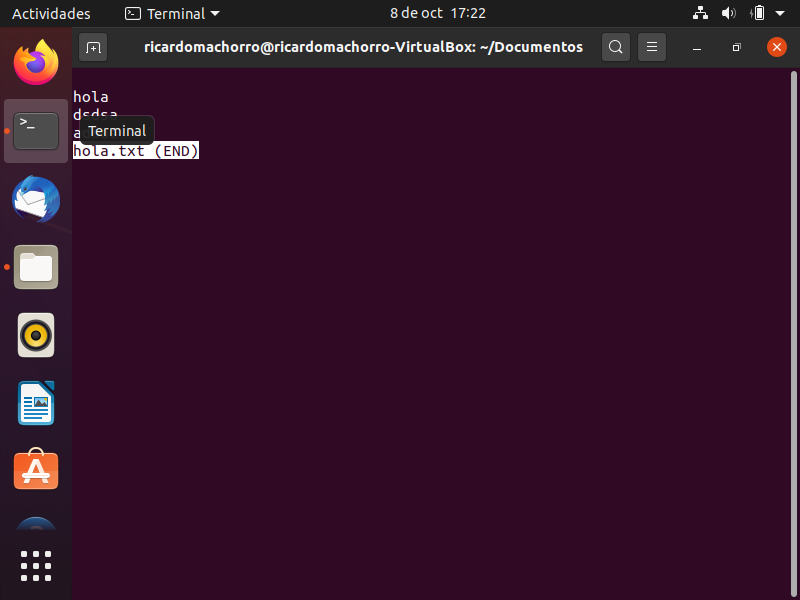


Comando ps –fea

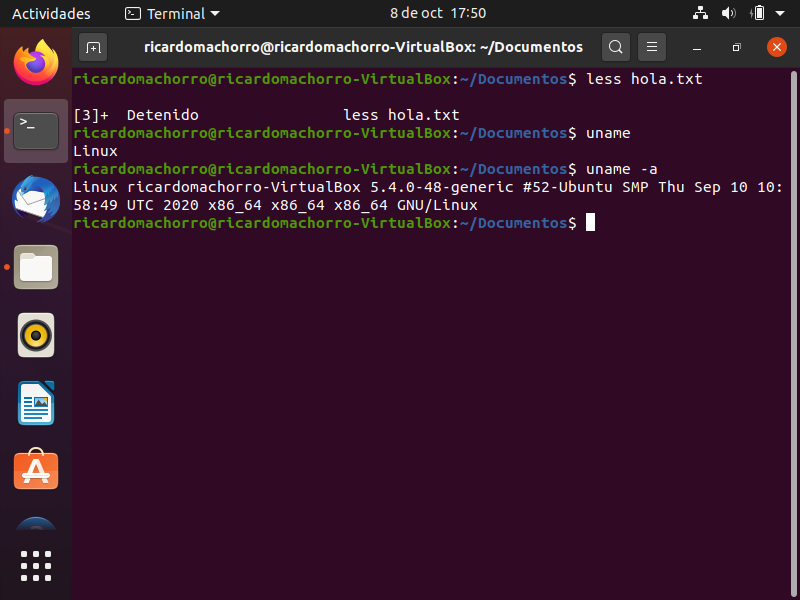
Comando less antes de ejecutarse



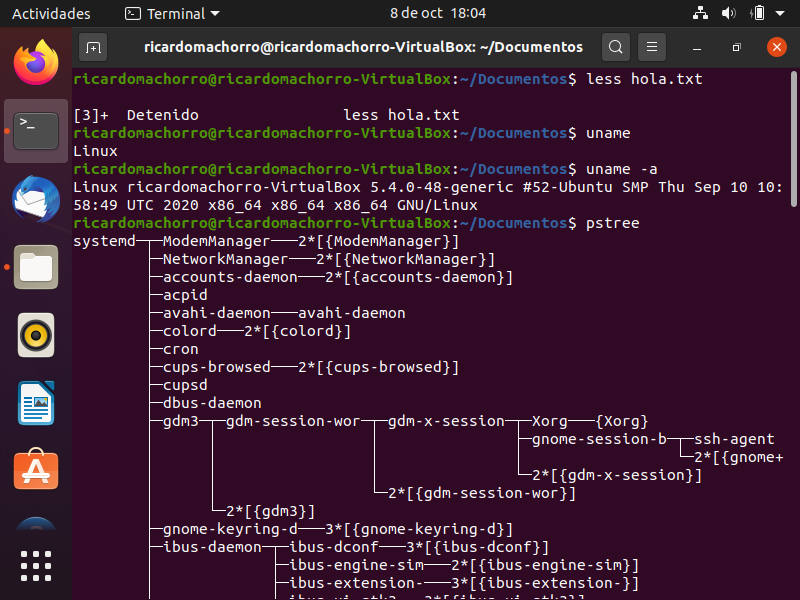
Comando Less después de ejecutarse



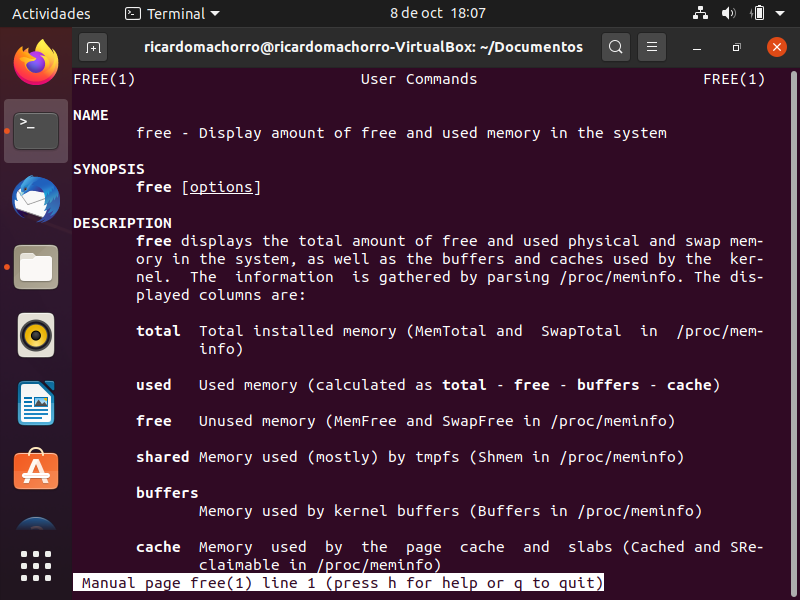
Función uname



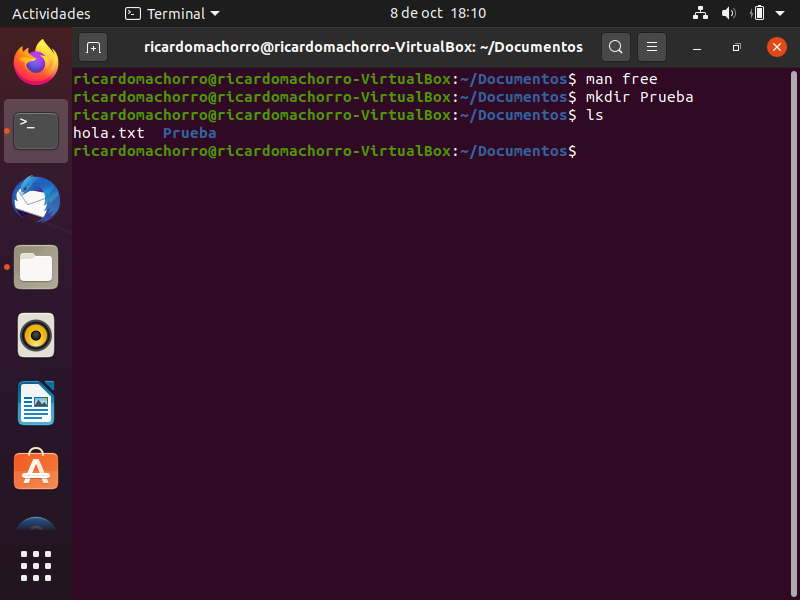
Comando pstree



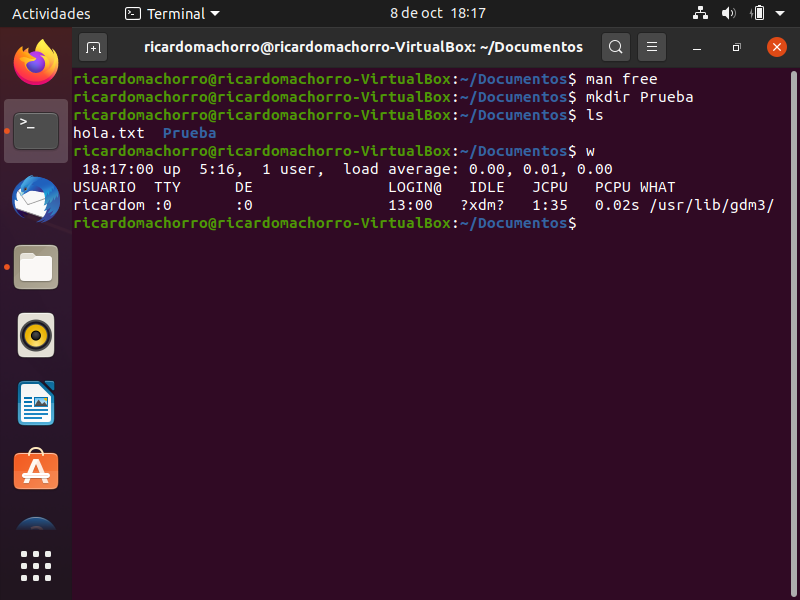
Comando man



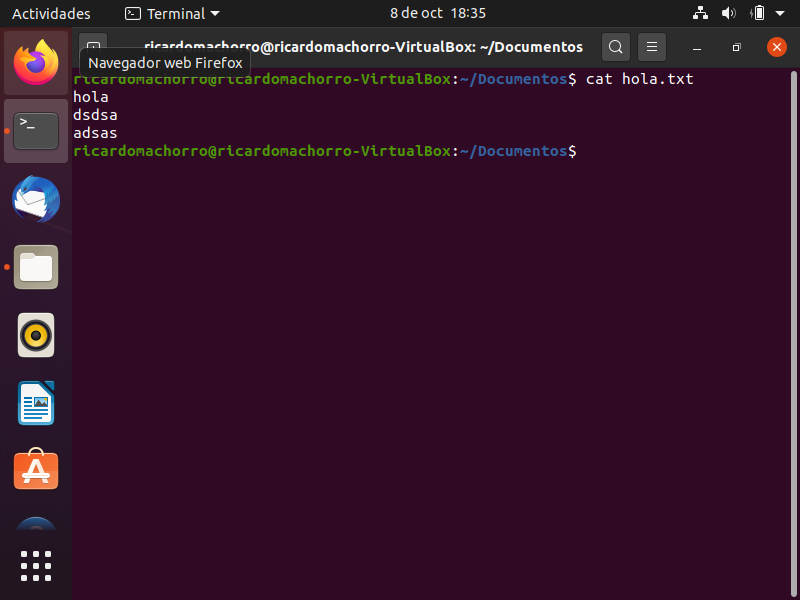
Comando mkdir



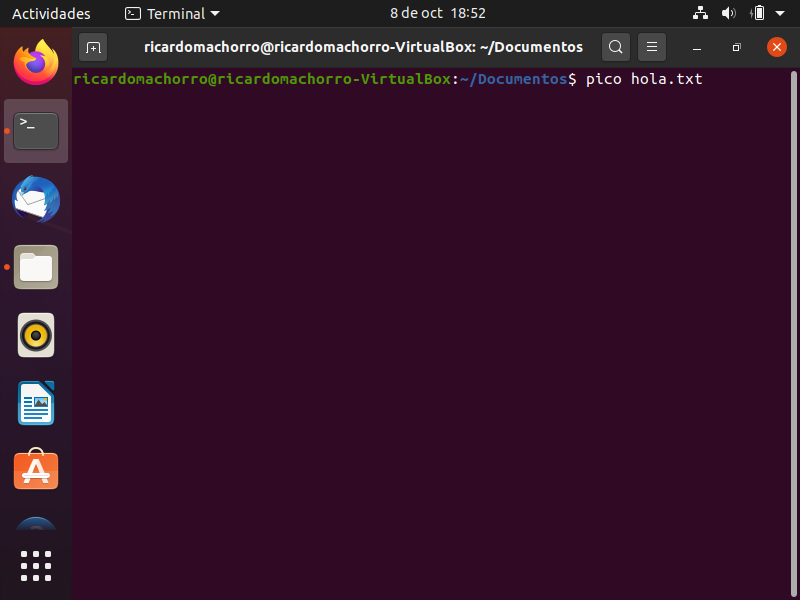
Comando w



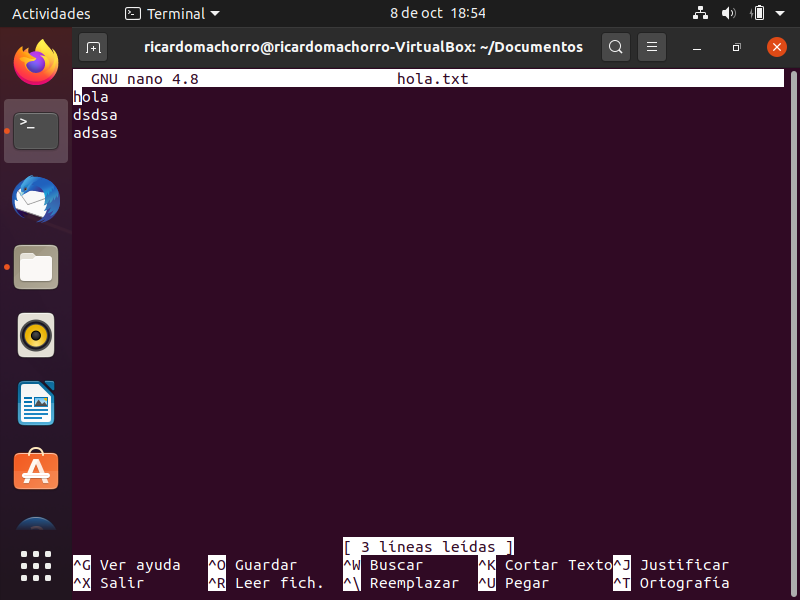
Función cat



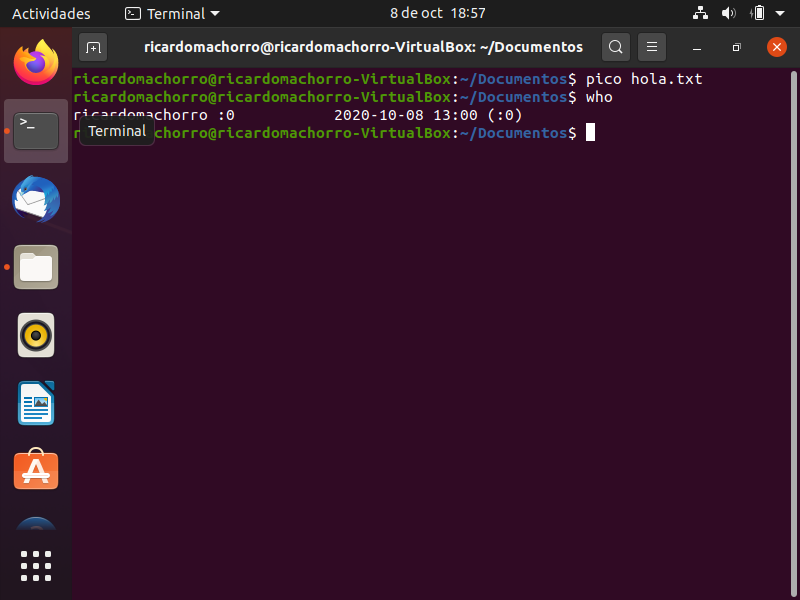
Comando pico antes de copilarse



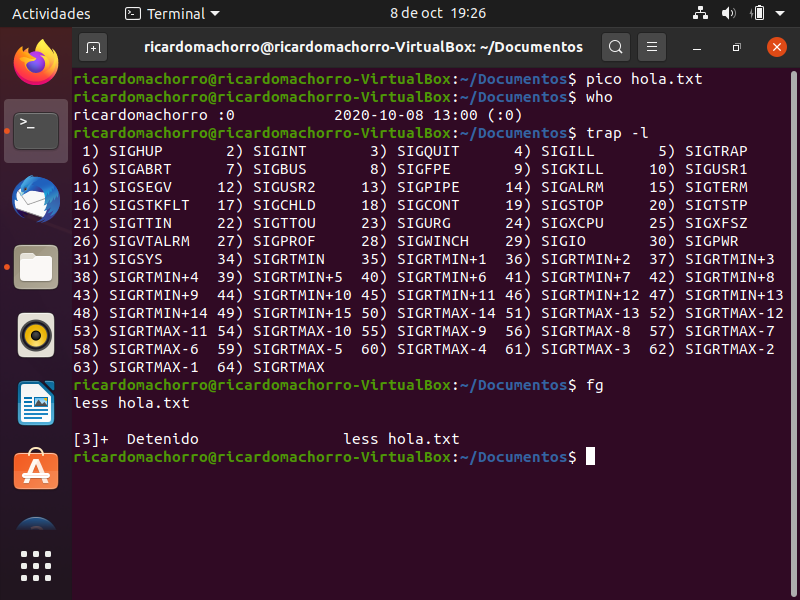
Función pico después de copilarse



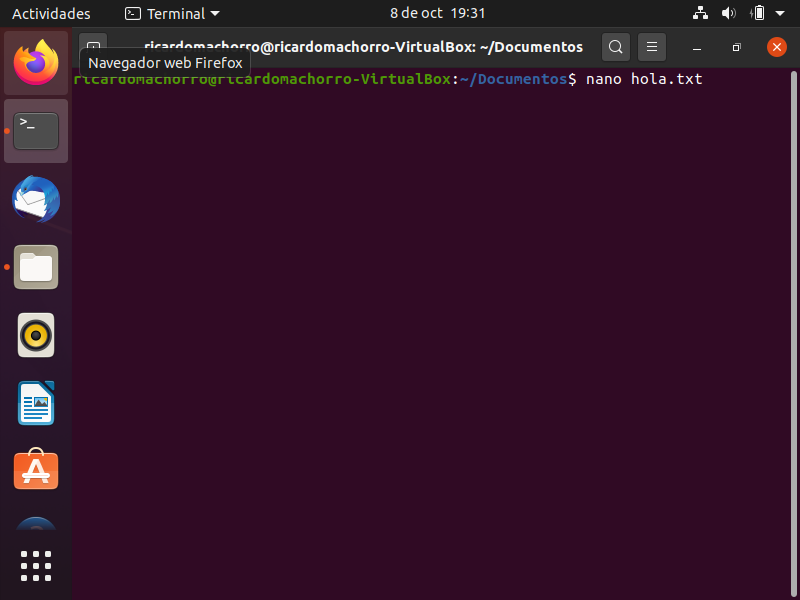
Comando who



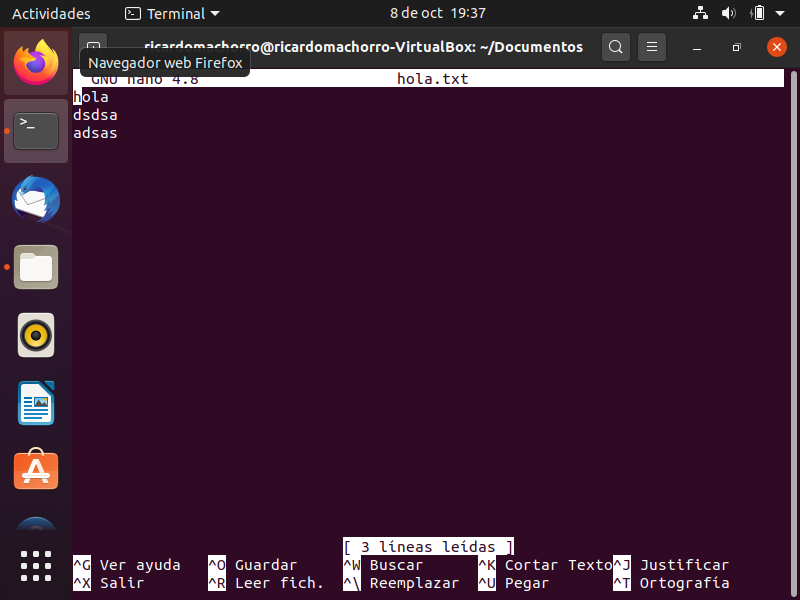
Comando fg



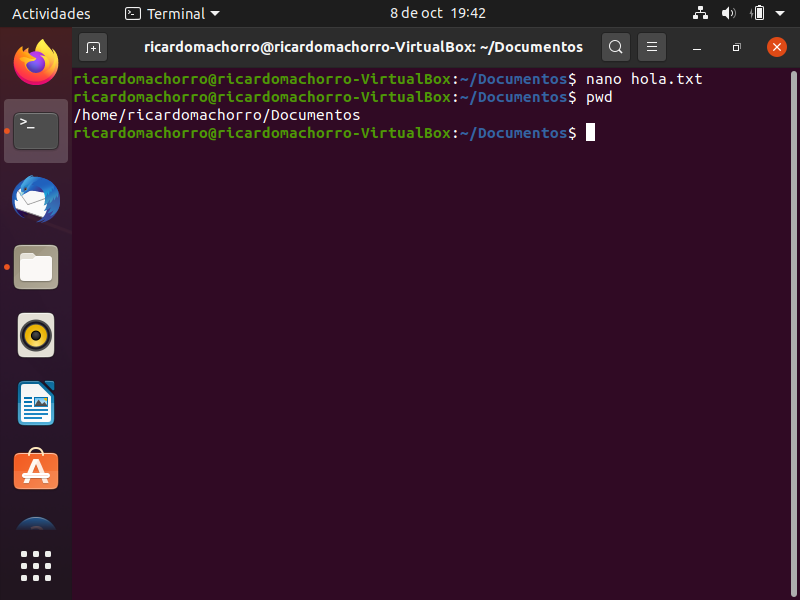
Comando nano antes de ejecutarse



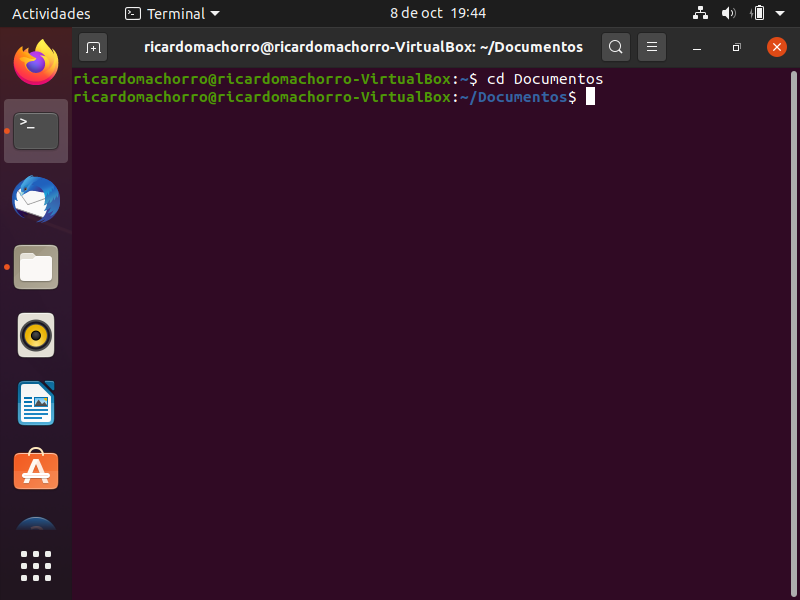
Comando nano después de ejecutarse



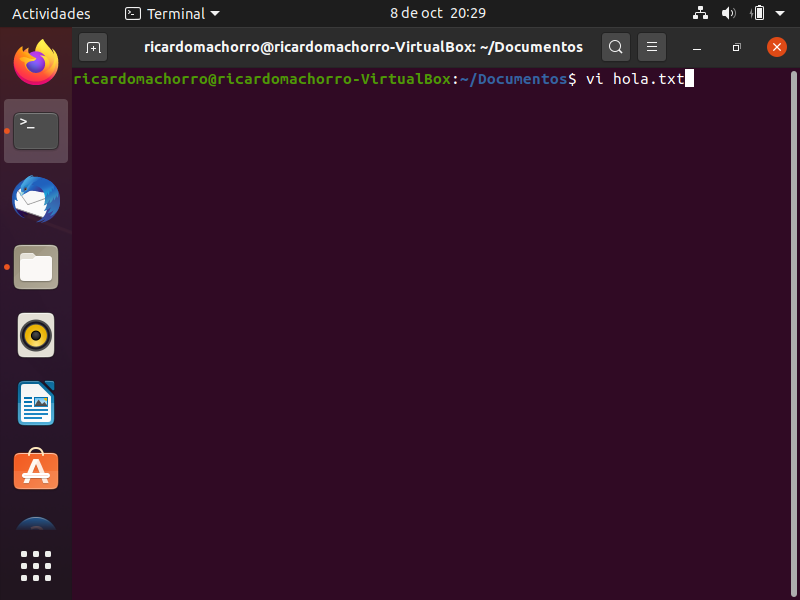
Comando pwd



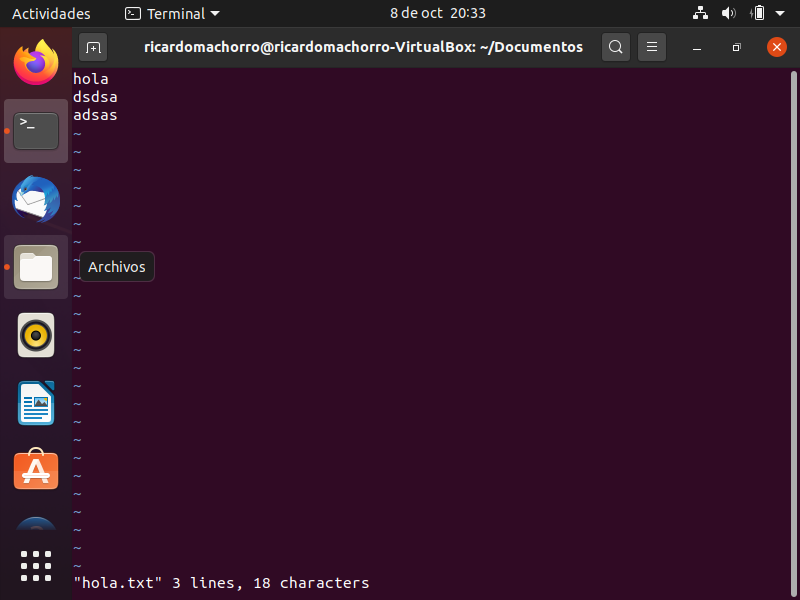
Comando cd



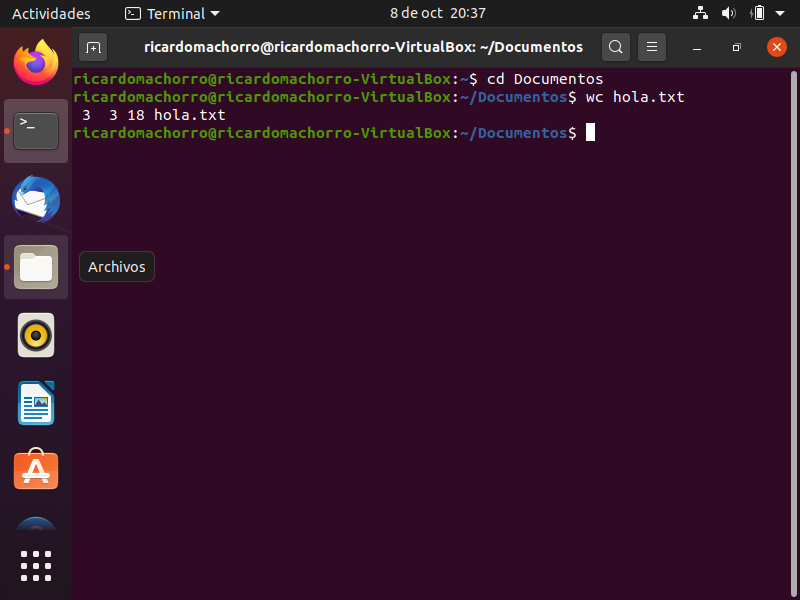
Comando vi antes de ejecutarse



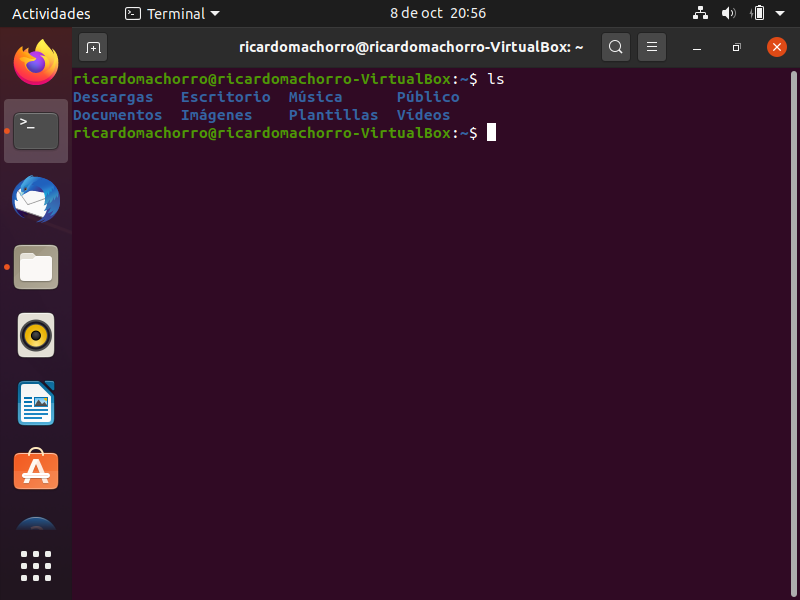
Comando vi luego de ejecutarse



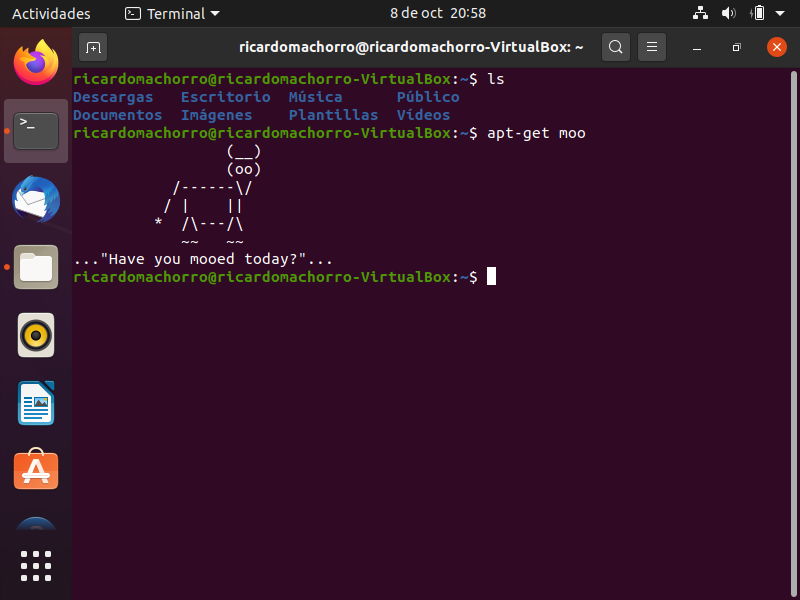
Comando wc



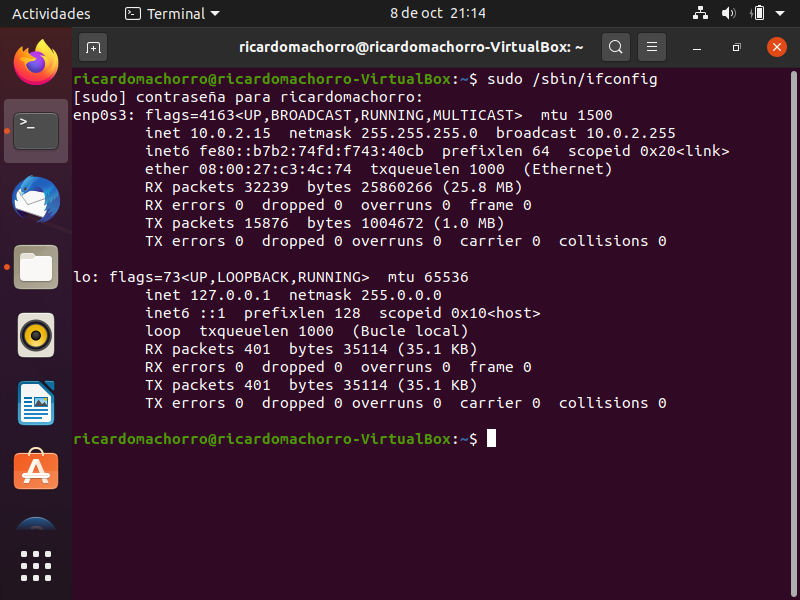
Comando ls



Comando apt-get

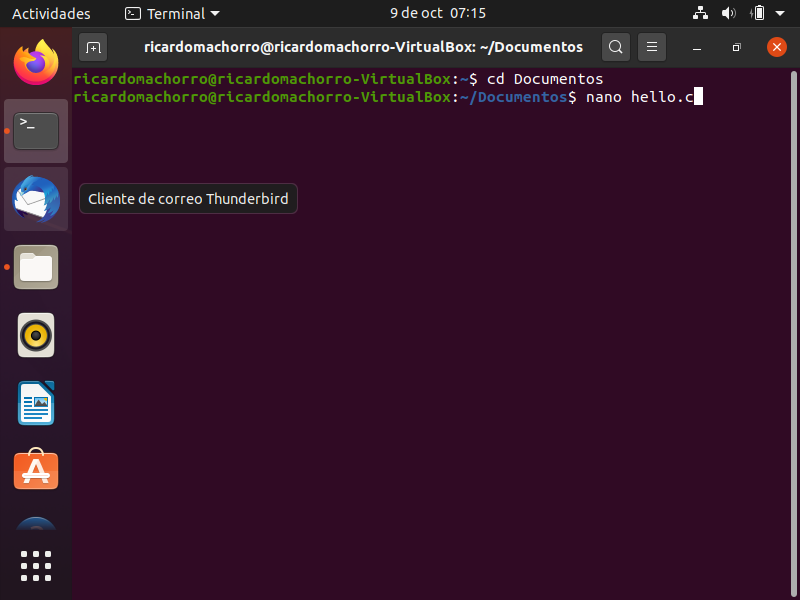


Comando sudo

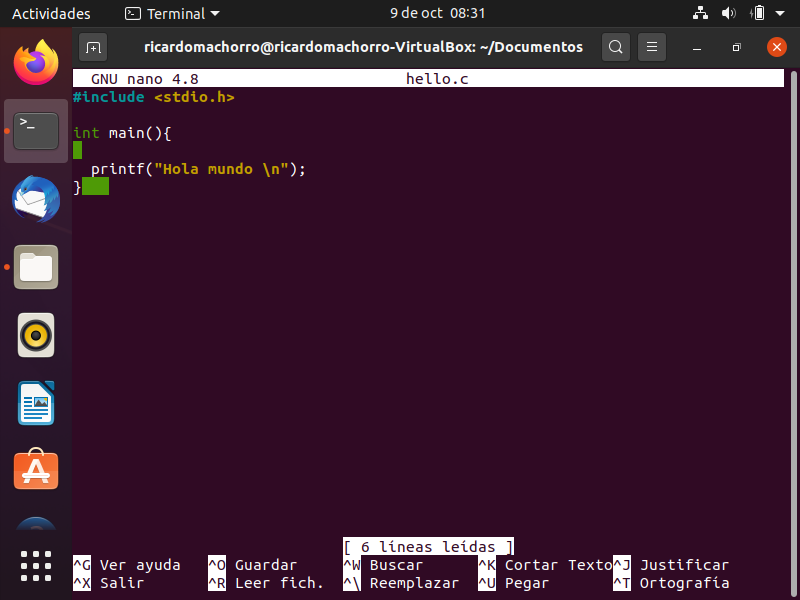


Compilar Hola Mundo en código para nano

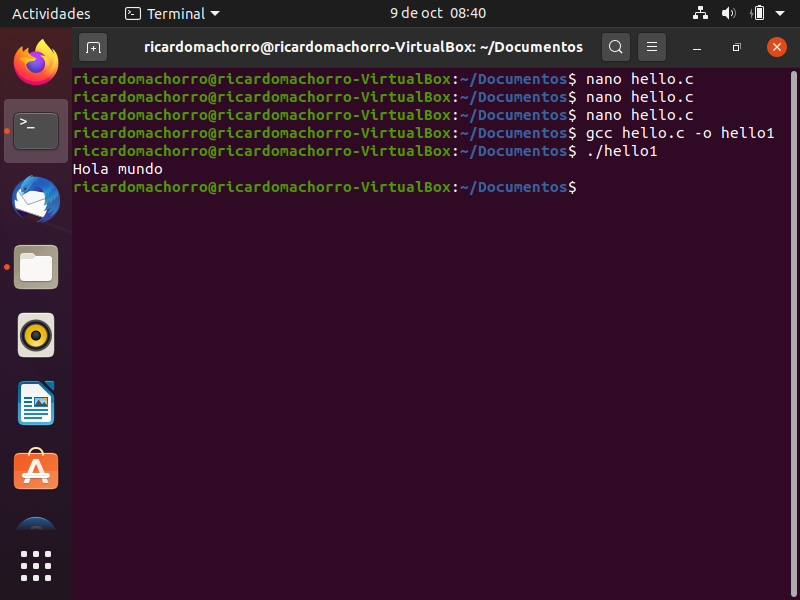
Creación Hola Mundo



Código Hola Mundo editado en nano

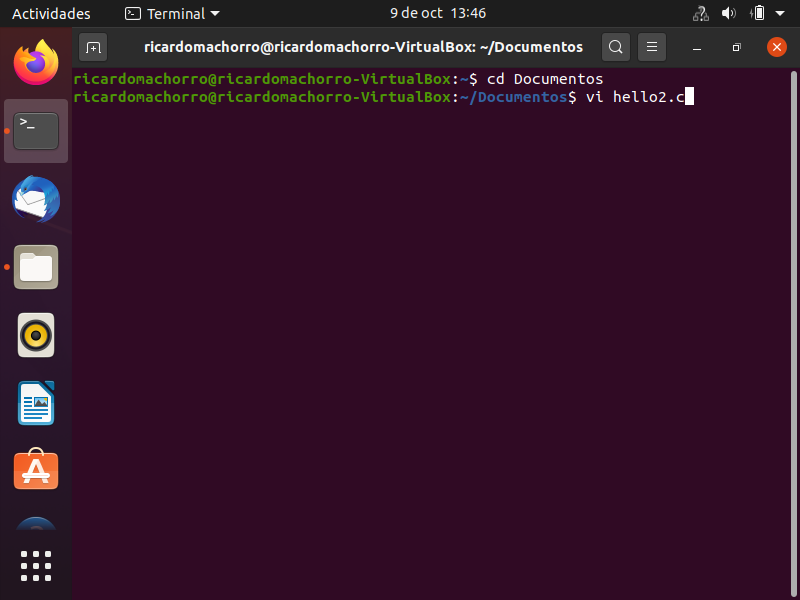


Código Hola Mundo compilación nano

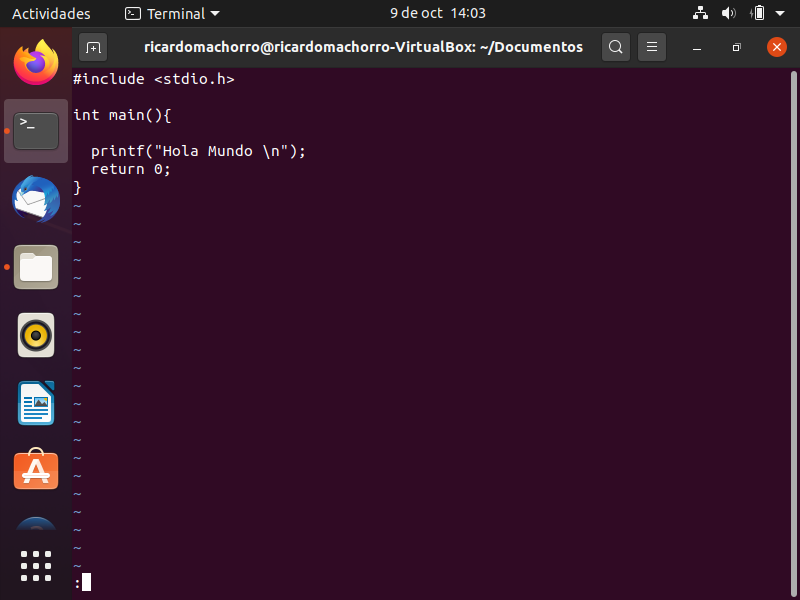


Compilar Hola Mundo en código para vi

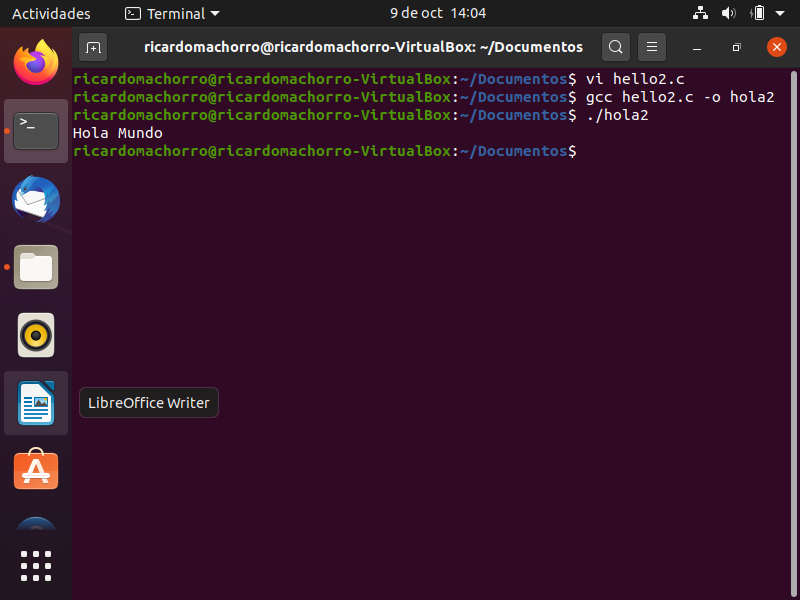
Creación Hola Mundo



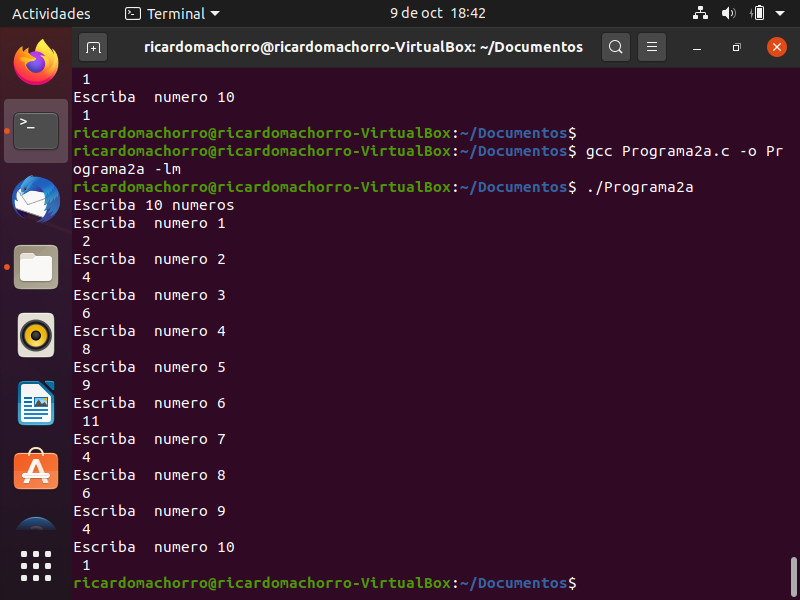
Edición Hola Mundo en vi



Ejecución hola mundo en vi



Compilación y pantalla de programa2a





Compilación y pantalla de programa3a

