

Tutorial: Uso básico del terminal de Linux.

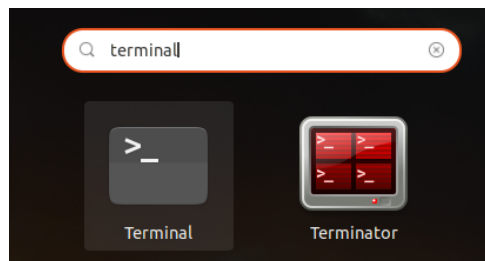
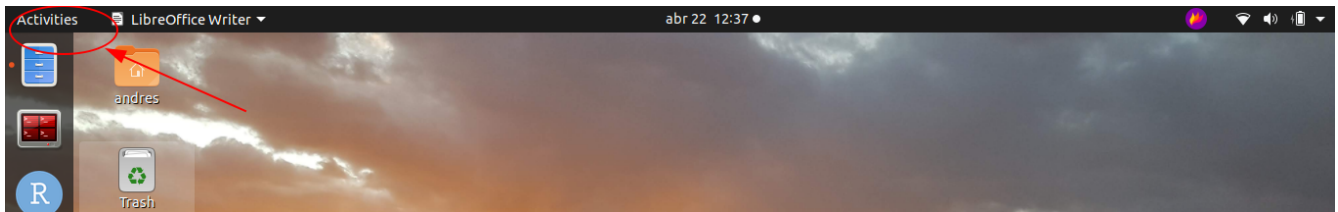
Autor: A. Cumsille

¿Por qué usar el terminal de linux?

El terminal es una interfaz de texto que permite acceder a todo el potencial de un computador de mejor forma que cualquier interfaz gráfica. Comparado con los gráficas, el texto es muy ligero y no utiliza tantos recursos, permitiendo optimizar la velocidad y eficiencia de los procesos.

1. Abrir el terminal

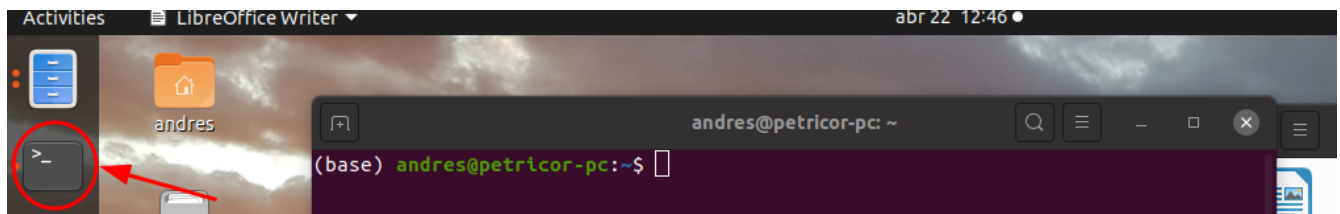
1.1 A través de Actividades tipeando terminal en el buscador



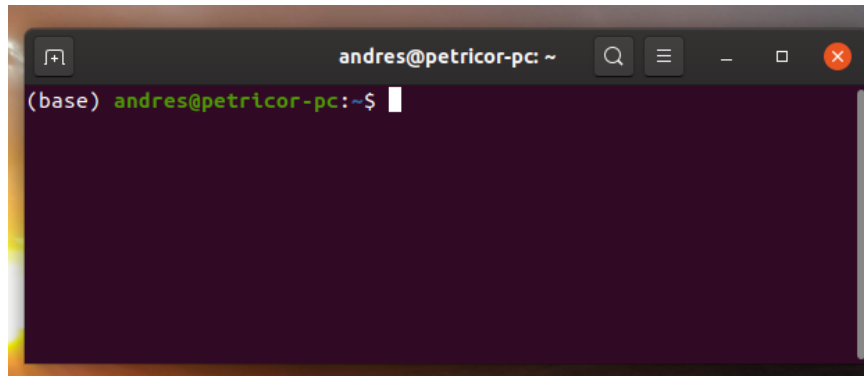
1.2 Entrando a alguna carpeta y presionando con botón derecho del mouse abrir en terminal

1.3 Presionando Alt + Ctrl + T

Nota: Se puede guardar el terminal en los favoritos, entonces estará accesible siempre en la barra de herramientas



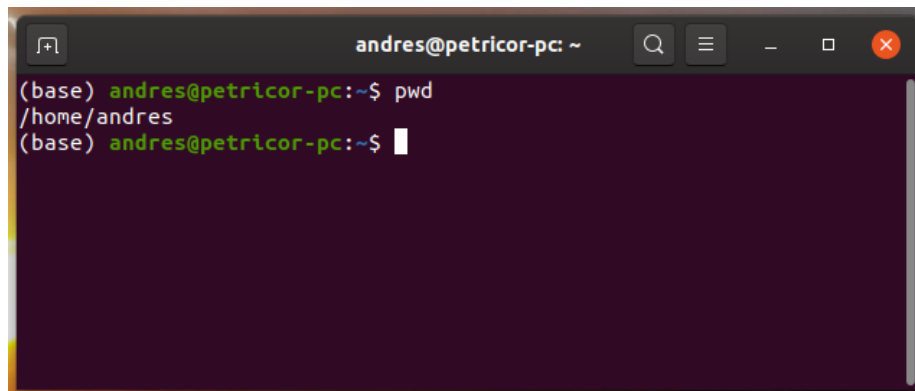
Tu terminal se debe ver así, quizás con otros colores y el texto ligeramente diferentes, pero básicamente lo mismo.



2. Comandos básicos

2.1 *Print working directory (pwd)*: Este comando entrega como resultado la ubicación del directorio donde se encuentra trabajando el terminal.

Escribe pwd en tu terminal y luego presiona enter



El resultado que entrega es /home/andres que es la carpeta donde se encuentra el terminal. Es importante saber esto, ya que el terminal realizará todos los procesos que le indiques en este directorio de trabajo, a menos que le especifiques lo contrario.

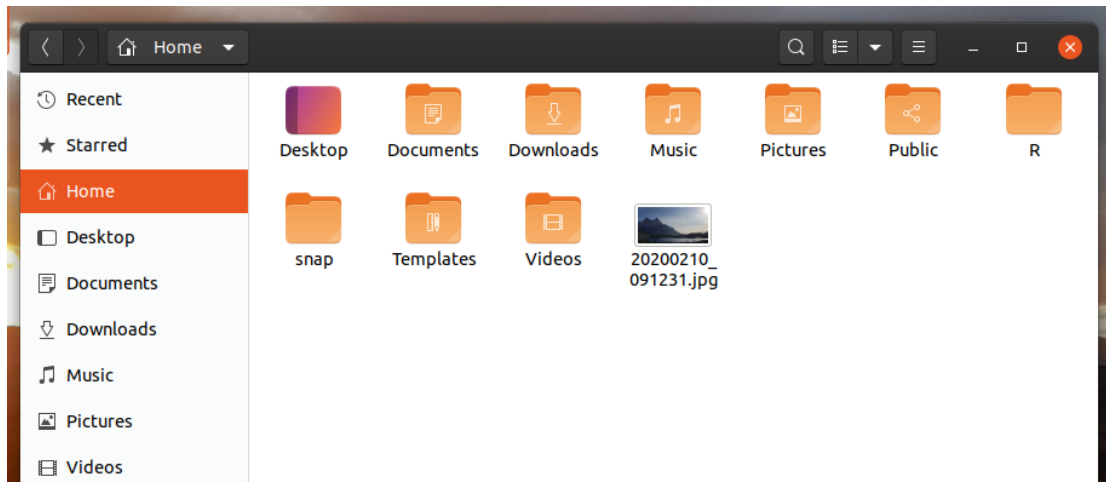
Si te fijas bien, el resultado se imprime en el mismo terminal y cuando termina de correr este comando aparece nuevamente el prompt, que en mi caso es el texto “andres@petricor-pc:~\$”. Este prompt indica que el proceso terminó y que el terminal está listo para aceptar otro comando.

2.2 *List (ls)*: Este comando lo que hace es generar una lista de todos los archivos y carpetas presentes en la carpeta de trabajo.

Escribe ls en tu terminal

```
andres@petricor-pc: ~  
(base) andres@petricor-pc:~$ pwd  
/home/andres  
(base) andres@petricor-pc:~$ ls  
20200210_091231.jpg  Documents  Music      Public  snap      Videos  
Desktop              Downloads  Pictures   R       Templates  
(base) andres@petricor-pc:~$
```

El resultado que entrega muestra todas las carpetas y los archivos en diferente color, en mi caso, el único archivo que hay es una imagen llamada 20200210_091231.jpg.
Si te fijas esto es igual a lo que muestra el explorador de archivos, solo que el terminal lo muestra en formato de texto.

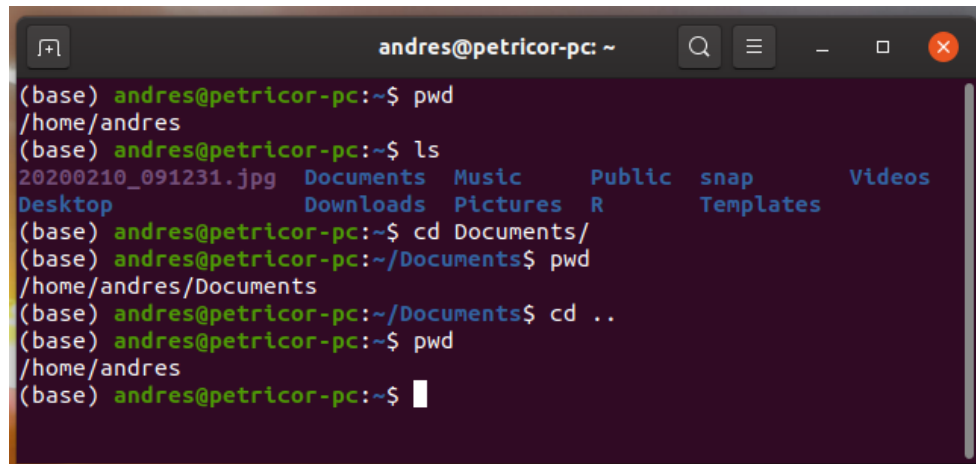


2.3 *Change directory* (cd): Este comando se utiliza para cambiar de directorio.

Escribe cd en tu terminal y luego de un espacio el nombre del directorio que te quieres mover, en este caso la carpeta Documents

```
andres@petricor-pc: ~/Documents  
(base) andres@petricor-pc:~$ pwd  
/home/andres  
(base) andres@petricor-pc:~$ ls  
20200210_091231.jpg  Documents  Music      Public  snap      Videos  
Desktop              Downloads  Pictures   R       Templates  
(base) andres@petricor-pc:~$ cd Documents/  
(base) andres@petricor-pc:~/Documents$
```

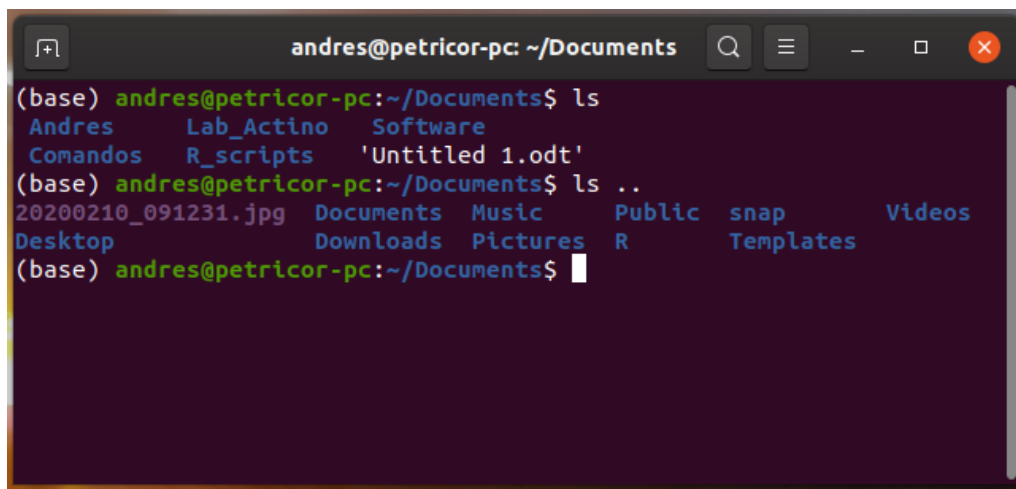
Puedes cambiarte al directorio principal, es decir a la carpeta madre que contiene a la carpeta donde estamos situados escribiendo `cd ..`



```
andres@petricor-pc: ~  
(base) andres@petricor-pc:~$ pwd  
/home/andres  
(base) andres@petricor-pc:~$ ls  
20200210_091231.jpg  Documents  Music      Public  snap      Videos  
Desktop              Downloads  Pictures   R       Templates  
(base) andres@petricor-pc:~$ cd Documents/  
(base) andres@petricor-pc:~/Documents$ pwd  
/home/andres/Documents  
(base) andres@petricor-pc:~/Documents$ cd ..  
(base) andres@petricor-pc:~$ pwd  
/home/andres  
(base) andres@petricor-pc:~$
```

Como puedes ver, luego de ingresar a la carpeta Documents, escribí `pwd` para que me diga cual es el directorio de trabajo (`/home/andres/Documents`), luego cuando escribí `cd ..` volví a tipear `pwd` y el directorio de trabajo volvió a ser `/home/andres`. Pero por ahora volvamos a la carpeta Documents.

Los dos puntos seguidos indican que uno está refiriendo la acción del comando a la carpeta principal, por ejemplo si uno escribe el comando `ls` desde el directorio de trabajo Documents, listará todo lo que hay en ese directorio, pero si uno escribe `ls ..` listará todo lo que hay en la carpeta contenedora



```
andres@petricor-pc: ~/Documents  
(base) andres@petricor-pc:~/Documents$ ls  
Andres  Lab_Actino  Software  
Comandos  R_scripts  'Untitled 1.odt'  
(base) andres@petricor-pc:~/Documents$ ls ..  
20200210_091231.jpg  Documents  Music      Public  snap      Videos  
Desktop              Downloads  Pictures   R       Templates  
(base) andres@petricor-pc:~/Documents$
```

2.4 Make directory (mkdir). Este comando sirve para crear un directorio

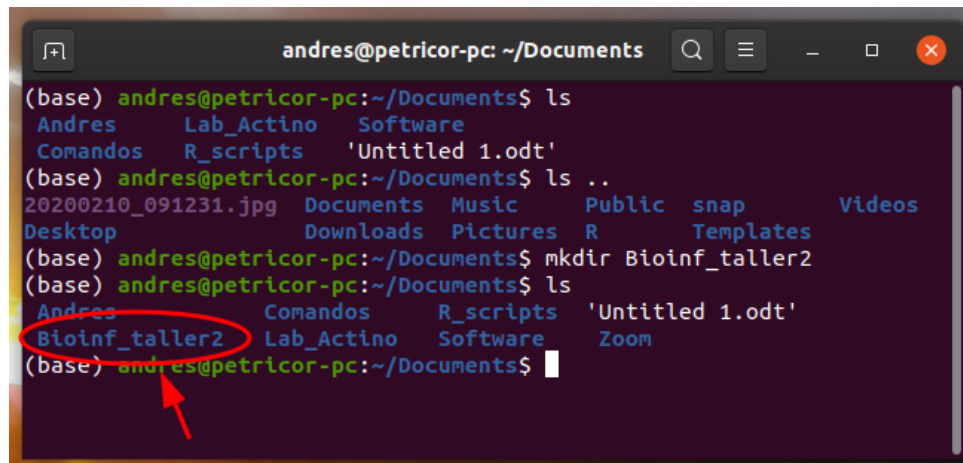
Vamos a crear una carpeta con el nombre `Bioinf_taller2`

Para eso escribe `mkdir Bioinf_taller2` en tu terminal. Podemos revisar con el comando `ls` que la carpeta se creó, y al mismo tiempo podemos verlo en el explorador de archivos.

Nota: Evita utilizar espacios al crear tus carpetas y archivos en ubuntu. El terminal reconoce los espacios como nuevos parámetros dentro del comando, por lo tanto lo mejor es comenzar a trabajar desde el comienzo en Ubuntu sin espacios.

Nota: Ten cuidado con el uso de mayúsculas y minúsculas, el terminal nota la diferencia, por ende no da igual si uno escribe `cd bioinf_taller2` o `cd Bioinf_taller2`

Nota: Al utilizar la flecha para arriba puedes volver a observar los comandos que ya has utilizado.



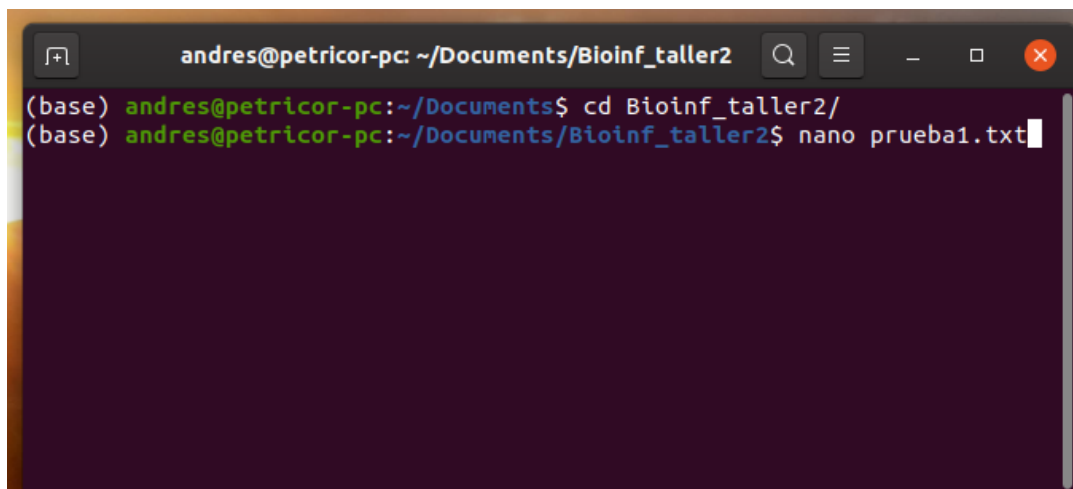
```
(base) andres@petricor-pc: ~/Documents
Andres Lab_Actino Software
Comandos R_scripts 'Untitled 1.odt'
(base) andres@petricor-pc: ~/Documents$ ls ..
20200210_091231.jpg Documents Music Public snap Videos
Desktop Downloads Pictures R Templates
(base) andres@petricor-pc: ~/Documents$ mkdir Bioinf_taller2
(base) andres@petricor-pc: ~/Documents$ ls
Andres Comandos R_scripts 'Untitled 1.odt'
Bioinf_taller2 Lab_Actino Software Zoom
(base) andres@petricor-pc: ~/Documents$
```

A terminal window titled 'andres@petricor-pc: ~/Documents'. It shows a series of commands: `ls`, `ls ..`, `mkdir Bioinf_taller2`, and another `ls`. The output of the final `ls` command lists the contents of the `~/Documents` directory, including the newly created `Bioinf_taller2` folder. A red circle highlights the `Bioinf_taller2` folder in the output, and a red arrow points to it.



2.5 (nano): Comando que sirve para abrir archivos y también para crearlos.

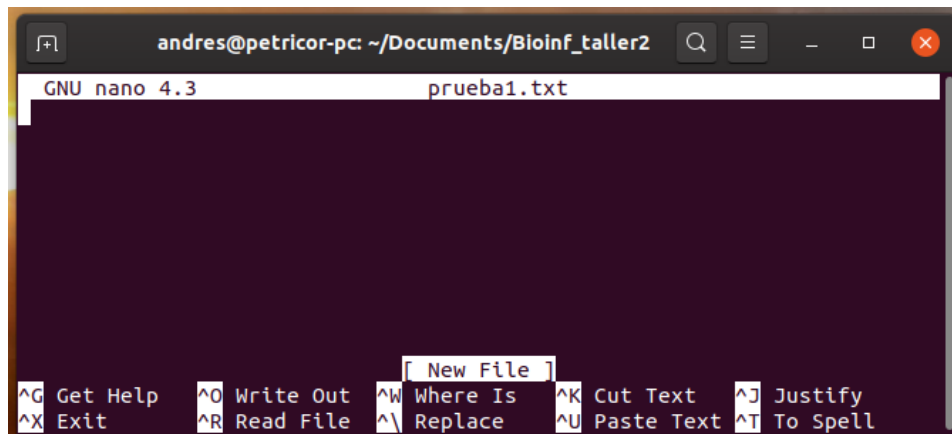
Vamos a ingresar a la carpeta `Bioinf_taller2` y dentro de esta carpeta vamos a crear un archivo llamado `prueba1.txt`. Para esto escribe en tu terminal `nano prueba1.txt`



```
(base) andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc: ~/Documents$ cd Bioinf_taller2/
(base) andres@petricor-pc: ~/Documents/Bioinf_taller2$ nano prueba1.txt
```

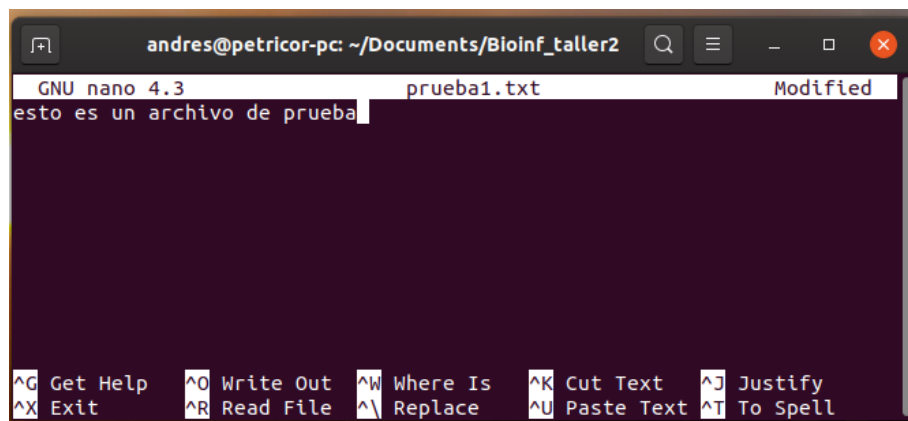
A terminal window titled 'andres@petricor-pc: ~/Documents/Bioinf_taller2'. It shows the commands `cd Bioinf_taller2/` and `nano prueba1.txt`. The prompt is now `andres@petricor-pc: ~/Documents/Bioinf_taller2$`.

Luego se abrirá nano, el editor de texto



```
andres@petricor-pc: ~/Documents/Bioinf_taller2
GNU nano 4.3 prueba1.txt
New File
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify
^X Exit      ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell
```

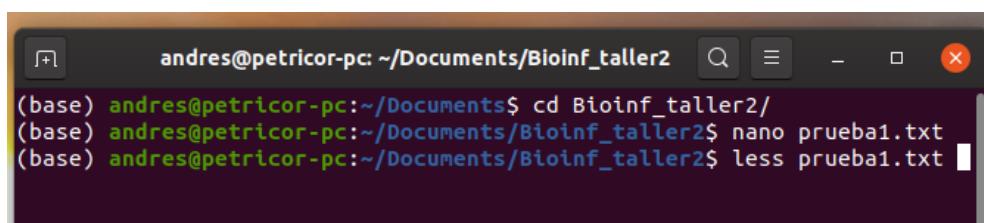
Aquí vamos a escribir lo que queramos ingresar en nuestro archivo, por ejemplo escribamos “esto es un archivo de prueba” y luego apretamos ctrl + x para salir y guardar el archivo.



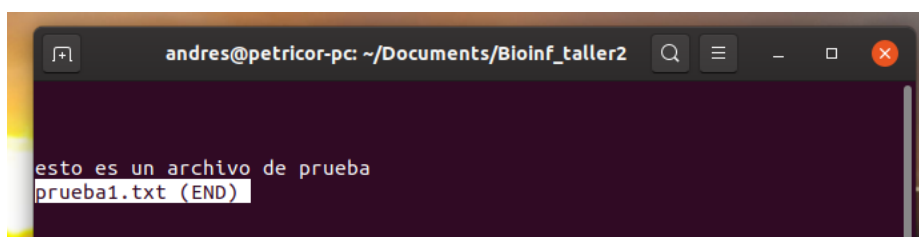
```
andres@petricor-pc: ~/Documents/Bioinf_taller2
GNU nano 4.3 prueba1.txt Modified
esto es un archivo de prueba
^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify
^X Exit      ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell
```

2.6 (less): este comando se utiliza para ver, y no editar un archivo. Es particularmente útil para cuando uno quiere leer un archivo que puede ser delicado si uno modifica por error, y también para navegar dentro de archivos de gran tamaño.

Veamos el archivo que hicimos en el paso anterior con less.



```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents$ cd Bioinf_taller2/
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ nano prueba1.txt
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ less prueba1.txt
```

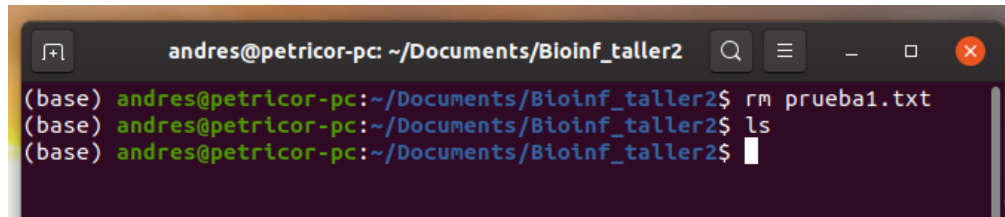


```
andres@petricor-pc: ~/Documents/Bioinf_taller2
esto es un archivo de prueba
prueba1.txt (END)
```

Podemos ver la información que habíamos puesto en el archivo de prueba que creamos. Para salir del visor de texto debemos apretar la letra q.

2.7 Remove (rm): Este archivo se utiliza para eliminar archivos y carpetas.

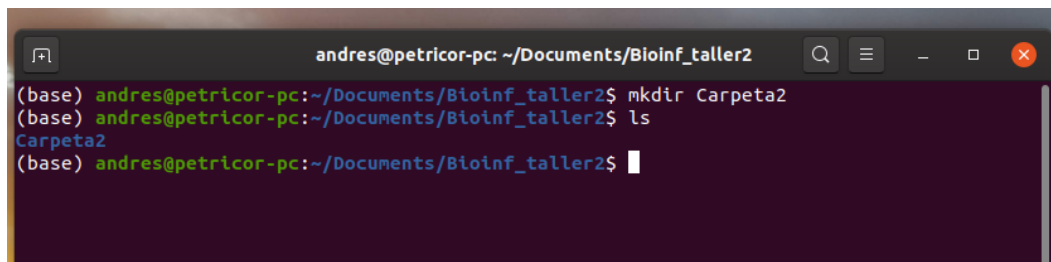
Eliminemos el archivo de prueba que recién creamos. Esto lo logramos escribiendo en el terminal `rm prueba1.txt`.

A terminal window titled 'andres@petricor-pc: ~/Documents/Bioinf_taller2'. It shows three commands: 'rm prueba1.txt', 'ls', and a prompt. The output of 'ls' is empty, indicating the file has been removed.

```
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ rm prueba1.txt
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

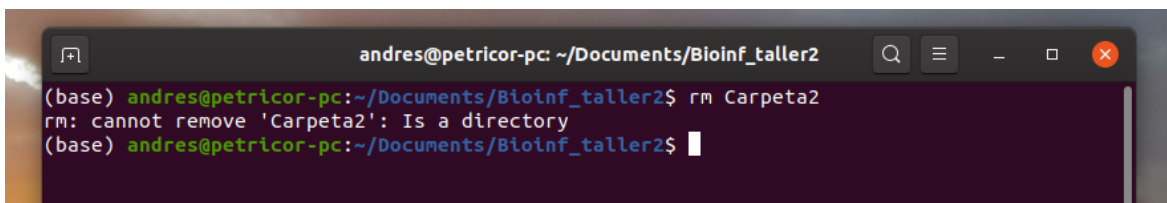
Puedes observar que luego de eliminar el archivo utilicé el comando `ls` para ver la lista de cosas que hay dentro de la carpeta, y no me entregó nada de resultado debido a que no hay ningún archivo dentro de esta carpeta.

Vamos a crear una nueva carpeta dentro de esta carpeta. Escribe `mkdir Carpeta2`. Comprobemos que se creó con el comando `ls`

A terminal window titled 'andres@petricor-pc: ~/Documents/Bioinf_taller2'. It shows two commands: 'mkdir Carpeta2' and 'ls'. The output of 'ls' shows 'Carpeta2' as a new directory.

```
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ mkdir Carpeta2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls
Carpeta2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

Ahora para eliminar esta carpeta escribe `rm Carpeta2`

A terminal window titled 'andres@petricor-pc: ~/Documents/Bioinf_taller2'. It shows the command 'rm Carpeta2' which results in an error message: 'rm: cannot remove 'Carpeta2': Is a directory'.

```
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ rm Carpeta2
rm: cannot remove 'Carpeta2': Is a directory
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

Aparece un error, esto es porque es necesario agregar la opción `-r` a este comando. Este `-r` significa que va a remover una carpeta de forma recursiva. Es necesario agregar esta opción, porque el comando necesita remover la carpeta, sus archivos dentro y las carpetas dentro de esta carpeta y por eso es que lleva la opción `-r`. Intentemos de nuevo ahora escribiendo `rm -r Carpeta2`

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ rm Carpeta2
rm: cannot remove 'Carpeta2': Is a directory
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ rm -r Carpeta2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

Ahora para seguir trabajando, descarga el archivo Taller2.tar.xz en la carpeta Bioinf_taller2.

2.8 tar: Este comando permite descomprimir archivos comprimidos en formato tar, formato muy típico utilizado sólomente por Linux y Mac.

Vamos a descomprimir el archivo escribiendo el comando tar con la opción -xf y el nombre del archivo, es decir tar -xf Taller2.tar.xz.

La opción -xf es una abreviación de extract file.

Nota: Los comandos generalmente tienen diversas opciones para modificar sus funciones, estas se colocan con un guión luego de escribir el comando. Para saber las opciones de cada comando frecuentemente uno puede conocerlas al escribir el comando y la opción --help o -h.

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ tar --help
Usage: tar [OPTION...] [FILE]...
GNU 'tar' saves many files together into a single tape or disk archive, and can
restore individual files from the archive.

Examples:
  tar -cf archive.tar foo bar    # Create archive.tar from files foo and bar.
  tar -tvf archive.tar           # List all files in archive.tar verbosely.
  tar -xf archive.tar            # Extract all files from archive.tar.

Local file name selection:

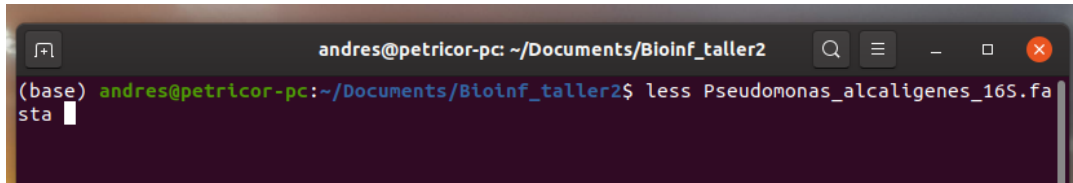
  --add-file=FILE                add given FILE to the archive (useful if its name
                                starts with a dash)
  -C, --directory=DIR           change to directory DIR
```

Entonces, ahora descomprimamos el archivo y veamos lo que hay adentro.

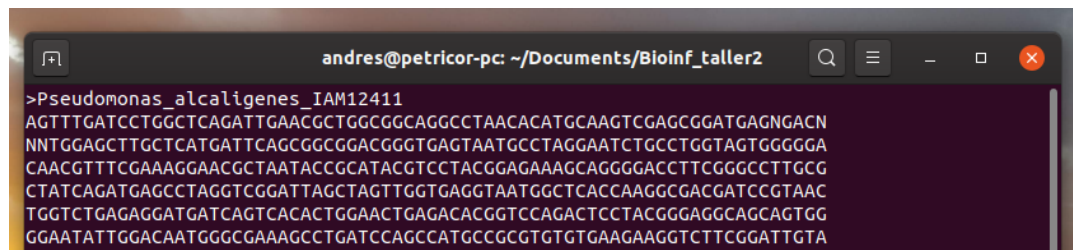
```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ tar -xf Taller2.tar.xz
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls
Acinetobacter_baumannii_16S.fasta  Pseudomonas_stutzeri_23S.fasta
Acinetobacter_baumannii_23S.fasta  Rhodococcus_erythropolis_16S.fasta
Acinetobacter_junii_16S.fasta       Rhodococcus_erythropolis_23S.fasta
archivo_x1.txt                     Rhodococcus_ruber_16S.fasta
archivo_x2.txt                     Rhodococcus_triatoeae_16S.fasta
Nocardia_farcinica_16S.fasta        Streptomyces_albus_16S.fasta
Nocardia_farcinica_23S.fasta        Streptomyces_albus_23S.fasta
Nocardia_seriolae_16S.fasta         Streptomyces_venezuelae_16S.fasta
Nocardia_terpenica_16S.fasta        Streptomyces_venezuelae_23S.fasta
Pseudomonas_alcaligenes_16S.fasta   Streptomyces_violaceusniger_16S.fasta
Pseudomonas_stutzeri_16S.fasta      Taller2.tar.xz
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```


Ahora lo que tenemos en la carpeta son 19 archivos fasta de diferentes bacterias, algunos archivos son la secuencia del gen rRNA 16S y otros son del gen rRNA 23S. Sumado a esto hay 2 archivos en formato .txt que están solo para ejemplificar algunas cosas más adelante.

Podemos ver el contenido de un archivo usando el comando less.



```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ less Pseudomonas_alcaligenes_16S.fasta
```

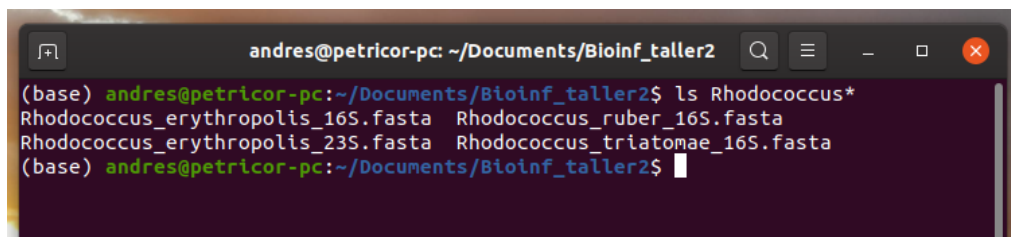


```
andres@petricor-pc: ~/Documents/Bioinf_taller2
>Pseudomonas_alcaligenes_IAM12411
AGTTTGATCCTGGCTCAGATTGAACGCTGGCGGACGGCCTAACACATGCAAGTCGAGCGGATGAGNGACN
NNTGGAGCTTGCTCATGATTACGCGGACGGGTGAGTAATGCCTAGGAATCTGCCTGGTAGTGGGGGA
CAACGTTTCGAAAGGAACGCTAATACCGCATACGTCCTACGGAGAAAGCAGGGGACCTTCGGGCCTTGCG
CTATCAGATGAGCCTAGGTCGGATTAGCTAGTTGGTGAGGTAATGGCTCACCAAGGCGACGATCCGTAAC
TGGTCTGAGAGGATGATCAGTCACACTGGAACCTGAGACACGGTCCAGACTCCTACGGGAGGCAGCAGTGG
GGAATATTGGACAATGGGCGAAAGCCTGATCCAGCCATGCCGCGTGTGTGAAGAAGGTCTTCGGATTGTA
```

Podemos movernos dentro del archivo utilizando las flechas del teclado. Luego salimos apretando q.

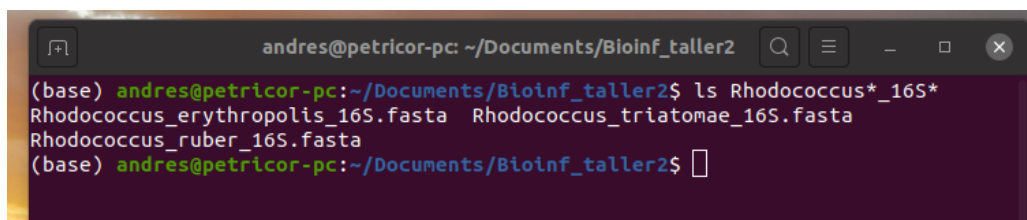
También podemos modificar los archivos utilizando nano, pero ahora no necesitamos esto.

Nota: el uso de * sirve como un carácter comodín. Usemos el * para ver listas solo de los archivos que necesitamos, por ejemplo solo de las cepas de Rhodococcus. Para eso tienes que escribir ls Rhodococcus*



```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls Rhodococcus*
Rhodococcus_erythropolis_16S.fasta  Rhodococcus_ruber_16S.fasta
Rhodococcus_erythropolis_23S.fasta  Rhodococcus_triatomae_16S.fasta
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

También se puede utilizar entre medio del texto, por ejemplo si solo quiero ver una lista de los archivos que poseen el gen rRNA 16S de Rhodococcus. Para eso tienes que escribir ls Rhodococcus*_16S*



```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls Rhodococcus*_16S*
Rhodococcus_erythropolis_16S.fasta  Rhodococcus_triatomae_16S.fasta
Rhodococcus_ruber_16S.fasta
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

Por último también puede ser utilizado para ver solo alguna extensión de archivos. Por ejemplo si uno quiere ver una lista solo de los archivos .txt. Para eso hay que escribir ls *.txt

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls *.txt
archivo_x1.txt  archivo_x2.txt
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

2.9 Concatenate (cat): Este comando sirve para concatenar archivos.

Por ejemplo podemos usar este comando para hacer un archivo concatenado de las secuencias de *Acinetobacter baumannii*. Para esto hay que escribir `cat Acinetobacter_baumannii_16S.fasta Acinetobacter_baumannii_23S.fasta > Concatenado_Abaumannii.fasta`.

La primera parte del script dice que hay que concatenar, luego está el nombre de los archivos que uno quiere concatenar, luego un `>` seguido por el nombre del archivo concatenado que queremos generar. Veamos con `ls` si se generó el archivo:

```
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ cat Acinetobacter_baumannii_16S.fasta Acinetobacter_baumannii_23S.fasta > Concatenado_Abaumannii.fasta
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls
Acinetobacter_baumannii_16S.fasta  Pseudomonas_stutzeri_23S.fasta
Acinetobacter_baumannii_23S.fasta  Rhodococcus_erythropolis_16S.fasta
Acinetobacter_junii_16S.fasta       Rhodococcus_erythropolis_23S.fasta
archivo_x1.txt                     Rhodococcus_ruber_16S.fasta
archivo_x2.txt                     Rhodococcus_triatoae_16S.fasta
Concatenado_Abaumannii.fasta       Streptomyces_albus_16S.fasta
Nocardia_farcinica_16S.fasta       Streptomyces_albus_23S.fasta
Nocardia_farcinica_23S.fasta       Streptomyces_venezuelae_16S.fasta
Nocardia_seriolae_16S.fasta        Streptomyces_venezuelae_23S.fasta
Nocardia_terpenica_16S.fasta       Streptomyces_violaceusniger_16S.fasta
Pseudomonas_alcaligenes_16S.fasta  Taller2.tar.xz
Pseudomonas_stutzeri_16S.fasta
```

Podemos leer este archivo con el comando `less` y comprobar como funciona la concatenación.

Ahora podemos concatenar todos los archivos fasta que tenemos. Para eso hay que escribir `cat *.fasta > Concatenado_total.fasta`. Aquí utilizamos el `*` para indicar que el comando `cat` debe seleccionar todos los archivos que tengan la extensión `.fasta` para generar un archivo.

Podemos explorar este archivo con el comando `less`.

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ less Concatenado_total.fasta
```

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
>Acinetobacter_baumannii_DSM30007
AACGCTGGCGGCAGGCTTAACACATGCAAGTCGAGCGGGGAAGGTAGCTTGCTACCGGACCTAGCGGCG
GACGGGTGAGTAATGCTTAGGAATCTGCCTATTAGTGGGGGACAACATCTCGAAAGGGATGCTAATACCG
CATACGCTCTACGGGAGAAAGCAGGGGATCTTCGGACCTTGCGCTAATAGATGAGCCTAAGTCGGATTAG
CTAGTTGGTGGGTAAAGGCCTACCAAGGCGACGATCTGTAGCGGGTCTGAGAGGATGATCCGCCACACT
GGGACTGAGACACGGCCAGACTCTACGGGAGGCAGCAGTGGGGAATATTGGACAATGGGGGAACCTT
GATCCAGCCATGCCGCGTGTGTGAAGAAGGCCTTATGGTTGTAAAGCACTTTAAGCGAGGAGGAGGCTAC
TTTAGTTAATACCTAGAGATAGTGGACGTTACTCGCAGAATAAGCACCGGCTAACTCTGTGCCAGCAGCC
GCGGTAATACAGAGGGTGCGAGCGTTAATCGGATTTACTGGGCGTAAAGCGTGCGTAGGCGGCTTATTAA
:
```

Podemos navegar con las flechas, o también podemos escribir un número para que el comando nos dirija a una línea que uno quiera, por ejemplo al escribir 150, bajamos 150 líneas.

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
AGCCGTTTCGAGACTAGGACGTTGATAGGTTGGATGTGGAAGCATAGTGATATGTGAAGCTGACCAATACT
AATTGCTCGTGAGGCTTGACTATACAACACCA
>Nocardia_farcinica_N898
GACGAACGCTGGCGGCGTGCTTAACACATGCAAGTCGAGCGGTAAGGCCCTTCGGGGTACACGAGCGGCG
AACGGGTGAGTAACACGTGGGTGATCTGCCCTGTACTTCGGGATAAGCCTGGGAACTGGGTCTAATACC
GGATATGACCTTACATCGCATGGTGTGGTGGAAAGATTATCGGTACAGGATGGGCGCGCGGCTATC
AGCTTGTGGTGGGTAAATGGCCTACCAAGGCGACGACGGTAGCCGGCCTGAGAGGGCGACCGGCCACA
CTGGGACTGAGACACGGCCAGACTCTACGGGAGGCAGCAGTGGGGAATATTGCACAATGGGCGAAAGC
CTGATGCAGCGACGCGCGTGAGGGATGACGGCCTTCGGGTGTAAACCTCTTCGACAGGGACGAAGCG
:
```

El comando cat es bastante útil cuando uno trabaja con muchos archivos. De esta forma uno podría crear un archivo concatenado solo de los 16S de las cepas de Actinobacterias si uno quisiera.

Esto se podría realizar escribiendo

```
cat Nocardia_*_16S* Rhodococcus_*_16S* Streptomyces_*_16S* > Concatenado_16S_actinos.fasta
```

2.10 head: Este comando sirve para mostrar solo las primeras 10 líneas de un archivo.

```
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ head Concatenado_total.fasta
>Acinetobacter_baumannii_DSM30007
AACGCTGGCGGCAGGCTTAACACATGCAAGTCGAGCGGGGAAGGTAGCTTGCTACCGGACCTAGCGGCG
GACGGGTGAGTAATGCTTAGGAATCTGCCTATTAGTGGGGGACAACATCTCGAAAGGGATGCTAATACCG
CATACGCTCTACGGGAGAAAGCAGGGGATCTTCGGACCTTGCGCTAATAGATGAGCCTAAGTCGGATTAG
CTAGTTGGTGGGTAAAGGCCTACCAAGGCGACGATCTGTAGCGGGTCTGAGAGGATGATCCGCCACACT
GGGACTGAGACACGGCCAGACTCTACGGGAGGCAGCAGTGGGGAATATTGGACAATGGGGGAACCTT
GATCCAGCCATGCCGCGTGTGTGAAGAAGGCCTTATGGTTGTAAAGCACTTTAAGCGAGGAGGAGGCTAC
TTTAGTTAATACCTAGAGATAGTGGACGTTACTCGCAGAATAAGCACCGGCTAACTCTGTGCCAGCAGCC
GCGGTAATACAGAGGGTGCGAGCGTTAATCGGATTTACTGGGCGTAAAGCGTGCGTAGGCGGCTTATTAA
GTCCGATGTGAAATCCCCGAGCTTAACCTGGGAATTGCATTGCATGACTGGTGAGCTAGAGTATGGGAGAG
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

2.11 tail: Este comando sirve para ver las últimas 10 líneas de un archivo

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ tail Concatenado_total.fasta
CGGCCGCAAGGCTAAACTCAAAGGAATTGACGGGGCCCGCACAAGCGCGGAGCATGTGGCTTAATTC
GACGCAACGCGAAGAACCTTACCAAGGCTTGACATACACGGAAACCTGGAGACAGGGTCCCCCTTGT
GGTCGGGTGTACAGGTGGTGCATGGCTGTCGTCAGCTCGTGTGAGATGTTGGGTTAAGTCCCGCAACG
AGCGCAACCCTTGTCTGTGTTGCCAGCATGCCTTTCGGGGTGATGGGGACTCACAGGAGACTGCCGGGG
TCAACTCGGAGGAAGGTGGGGACGACGTCAAGTCATCATGCCCTTATGTCTTGGGCTGCACACGTGCTA
CAATGGCCGGTACAATGAGCTGCGAAGCCGTGAGGTGGAGCGAATCTCAAAAAGCCGGTCTCAGTTCGGA
TTGGGGTCTGCAACTCGACCCCATGAAGTCGGAGTCGCTAGTAATCGCAGATCAGCATTGCTGCGGTGAA
TACGTTCCCGGGCCTTGACACACCGCCGTCACGTCACGAAAGTCGGTAACACCCGAAGCCGGTGGCCC
AACCCTTGTTGGAGGGAGCCGTCGAAGGTGGGACTGGCGATTGGGACGAAGTCGTAACAAGGTAGCCGTAC
CGGAAGG
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

2.12 *Move (mv)*: Este comando se utiliza para mover archivos

Para ejemplificar este comando vamos a crear una nueva carpeta, entonces escribe `mkdir Carpeta3`.

Me interesa mover todos los archivos con formato `.txt` a esta carpeta. Entonces hay que escribir `mv *.txt Carpeta3`. Este comando funciona escribiendo en primer lugar el comando `mv`, luego el o los archivos que queremos mover y luego la carpeta de destino. Podemos ver con `ls` que en la `Carpeta3` están los archivos que movimos.

```
andres@petricor-pc: ~/Documents/Bioinf_taller2
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ mkdir Carpeta3
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ mv *.txt Carpeta3/
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$ ls Carpeta3/
archivo_x1.txt  archivo_x2.txt
(base) andres@petricor-pc:~/Documents/Bioinf_taller2$
```

Al igual que con el comando `rm`, si queremos mover carpetas es necesario agregar la opción `-r`. Ahora eliminemos la `Carpeta3` con todos sus archivos dentro escribiendo en el terminal `rm -r Carpeta3`

2.13 *Copy (cp)*: Este comando sirve para copiar archivos. Funciona muy similar al comando `mv`.

Listo. Ya has aprendido lo básico del terminal de Linux!