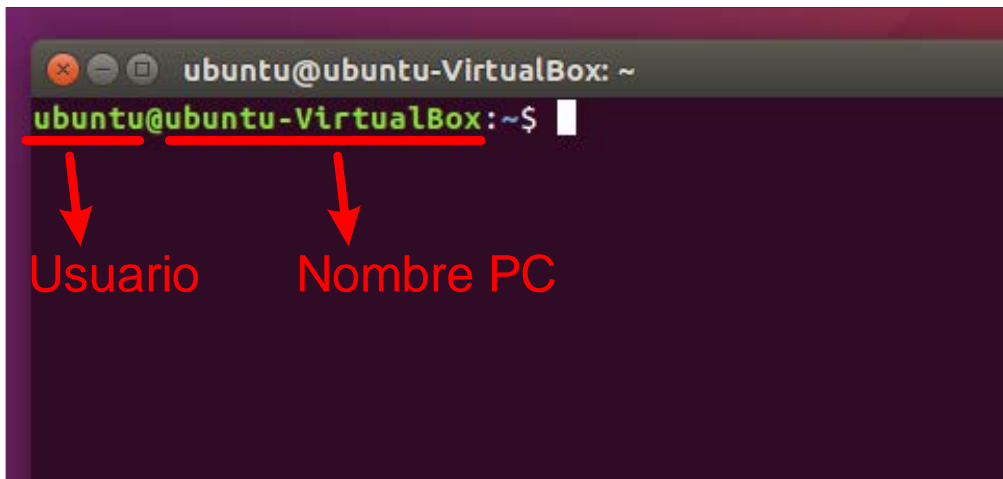


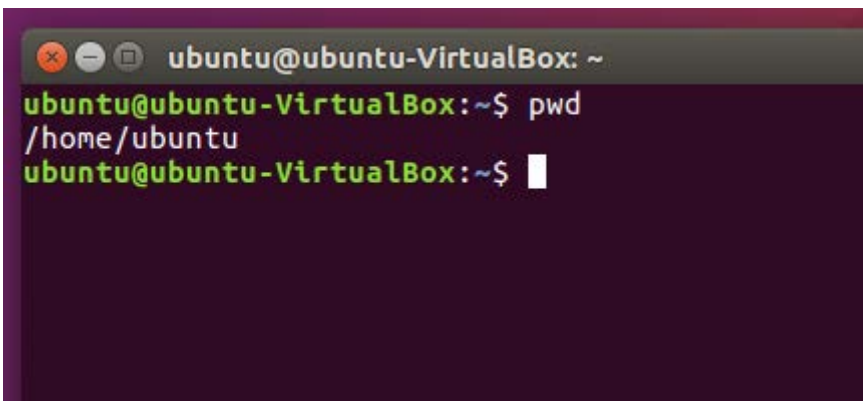
Comandos básicos, suficientes y necesarios para Programación y Algoritmos

1. Terminal recién abierta



2. ¿Dónde estoy?

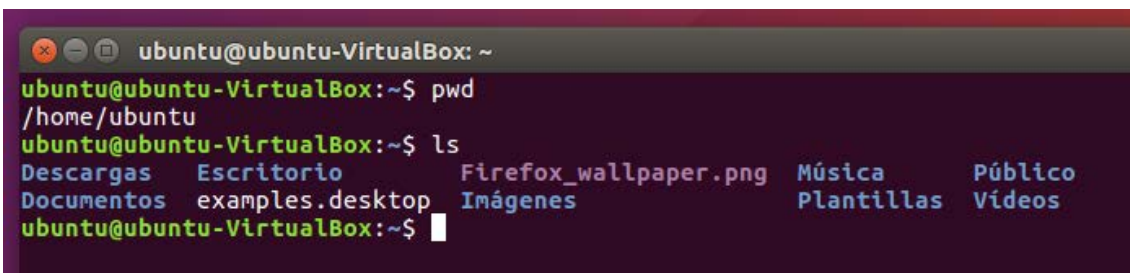
Comando `pwd`



- La carpeta "/" es como el "C:\\" en Windows, es la raíz del disco (sistema de archivos)
- La carpeta "home" es la carpeta de usuarios, entonces "/home/ubuntu" es la carpeta de usuario actual (Usuario: ubuntu).

3. ¿Qué otras carpetas y archivos hay en mi carpeta actual?

Comando `ls`

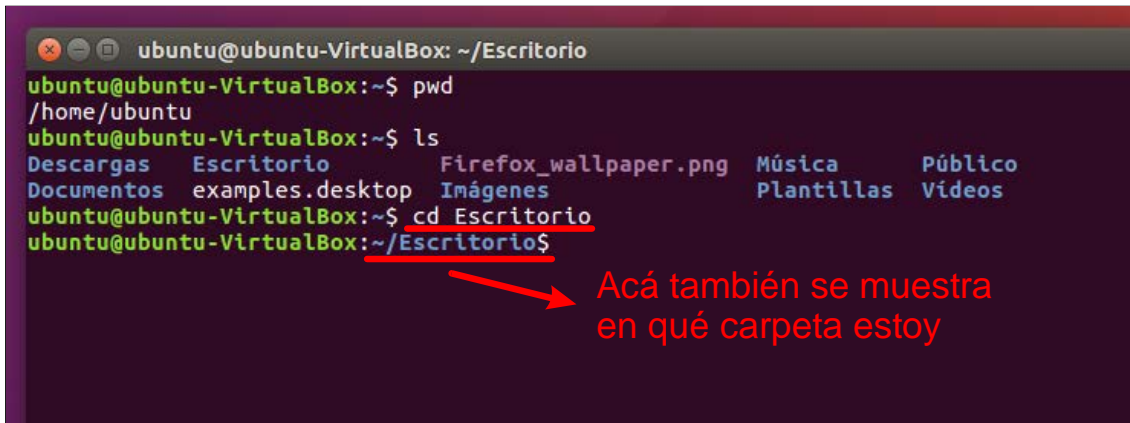


- En esta terminal Ubuntu muestra las otras carpetas en Azul

4. ¿Cómo me muevo a otra carpeta?

Comando `cd`

Formato: `cd <Carpeta destino>`

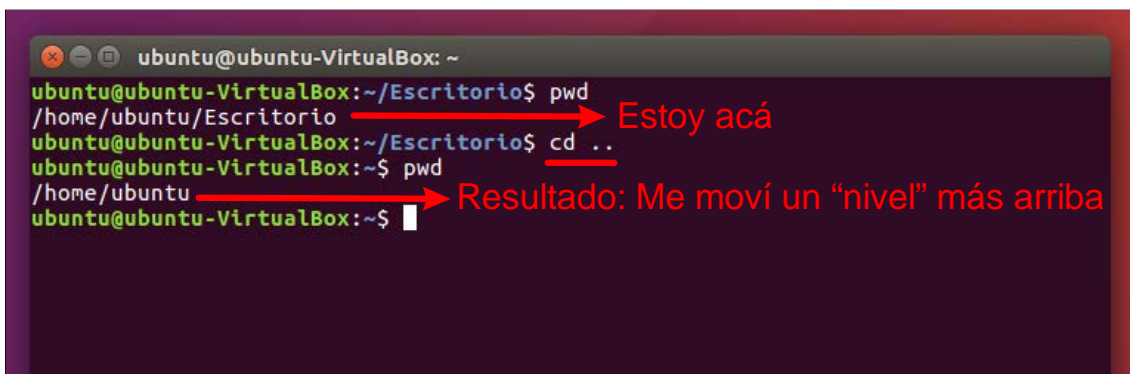


```
ubuntu@ubuntu-VirtualBox: ~/Escritorio
ubuntu@ubuntu-VirtualBox:~$ pwd
/home/ubuntu
ubuntu@ubuntu-VirtualBox:~$ ls
Descargas  Escritorio  Firefox_wallpaper.png  Música  Público
Documentos examples.desktop  Imágenes             Plantillas  Vídeos
ubuntu@ubuntu-VirtualBox:~$ cd Escritorio
ubuntu@ubuntu-VirtualBox:~/Escritorio$
```

Acá también se muestra en qué carpeta estoy

5. ¿Cómo voy a una carpeta más “arriba”?

Comando `cd ..`



```
ubuntu@ubuntu-VirtualBox: ~
ubuntu@ubuntu-VirtualBox:~/Escritorio$ pwd
/home/ubuntu/Escritorio
ubuntu@ubuntu-VirtualBox:~/Escritorio$ cd ..
ubuntu@ubuntu-VirtualBox:~$ pwd
/home/ubuntu
ubuntu@ubuntu-VirtualBox:~$
```

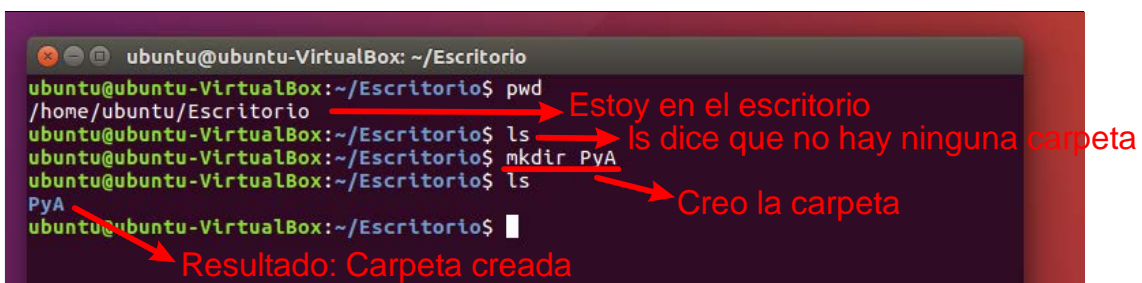
Estoy acá

Resultado: Me moví un “nivel” más arriba

- En Linux, “..” (dos puntos seguidos) significa: “El directorio que está un nivel más arriba”
- En Linux, “.” (un punto) significa: “La carpeta actual”

6. ¿Cómo creo una carpeta?

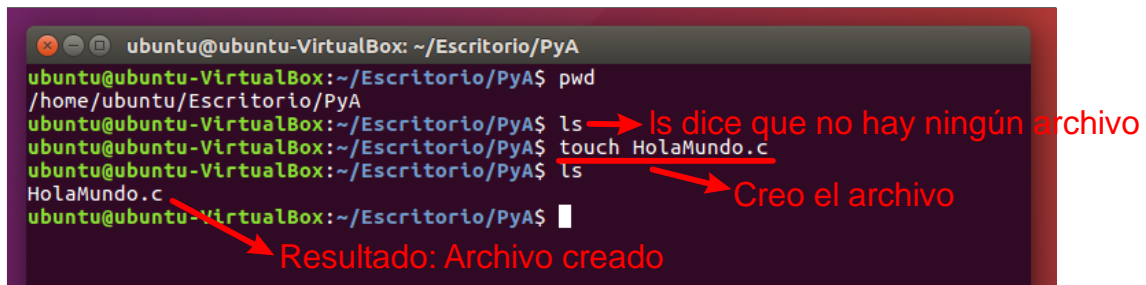
Comando `mkdir`



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio
ubuntu@ubuntu-VirtualBox:~/Escritorio$ pwd
/home/ubuntu/Escritorio
ubuntu@ubuntu-VirtualBox:~/Escritorio$ ls
ls dice que no hay ninguna carpeta
ubuntu@ubuntu-VirtualBox:~/Escritorio$ mkdir PyA
Creo la carpeta
ubuntu@ubuntu-VirtualBox:~/Escritorio$ ls
PyA
Resultado: Carpeta creada
```

7. ¿Cómo creo un archivo vacío?

Comando `touch <nombre_de_archivo>`



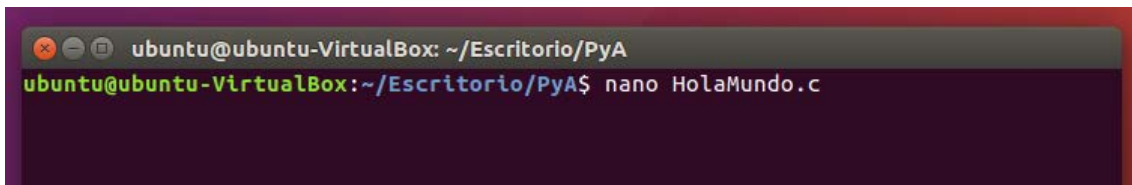
```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ pwd
/home/ubuntu/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ touch HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
HolaMundo.c
```

Annotations: Red arrows point from the text to the terminal. "ls dice que no hay ningún archivo" points to the first 'ls' command. "Creo el archivo" points to the 'touch' command. "Resultado: Archivo creado" points to the second 'ls' command output.

8. ¿Cómo edito el archivo?

Opción 1: Editor de texto en la consola.

`nano <nombre_de_archivo>`



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ nano HolaMundo.c
```

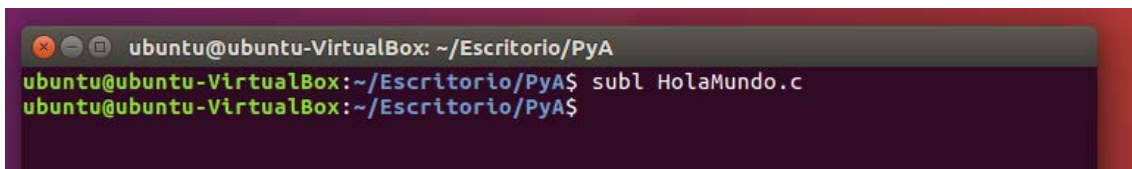
Esta opción es un poco “complicada”, pero es útil para editar cosas rápidas.



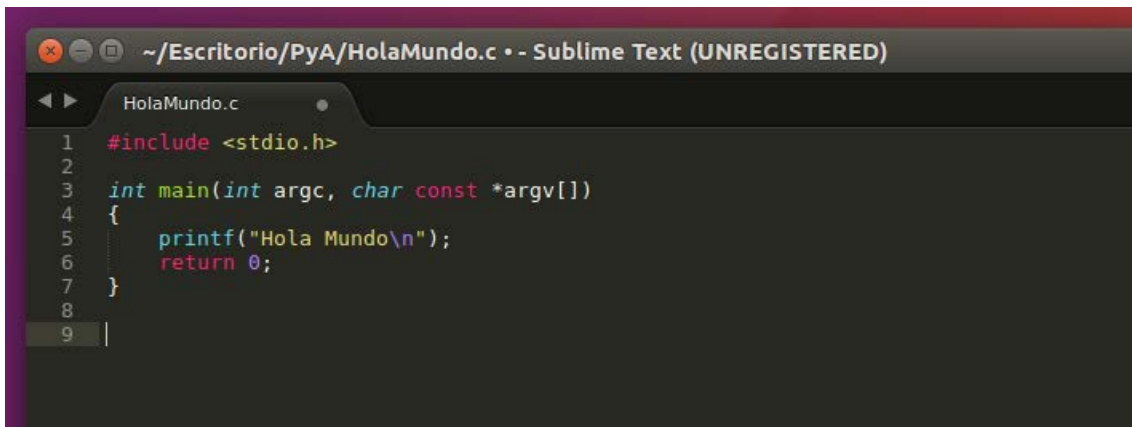
The screenshot shows the nano 2.5.3 editor interface with the file 'HolaMundo.c' open. The bottom status bar is circled in red and contains the following text: `[0 líneas leídas]` and a list of keyboard shortcuts: `^G Ver ayuda`, `^O Guardar`, `^W Buscar`, `^K Cortar Tex`, `^J Justificar`, `^C Posición`, `^X Salir`, `^R Leer fich.`, `^_ Reemplazar`, `^U Pegar txt`, `^T Ortografía`, `^_ Ir a línea`. A red arrow points from the text "Comandos: Ejemplo, Ctrl+X = Salir" to the `^X Salir` option.

Opción 2: Editor de texto recomendado: **SublimeText**

Abrir archivo desde la consola: `subl <nombre_de_archivo>`



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ subl HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

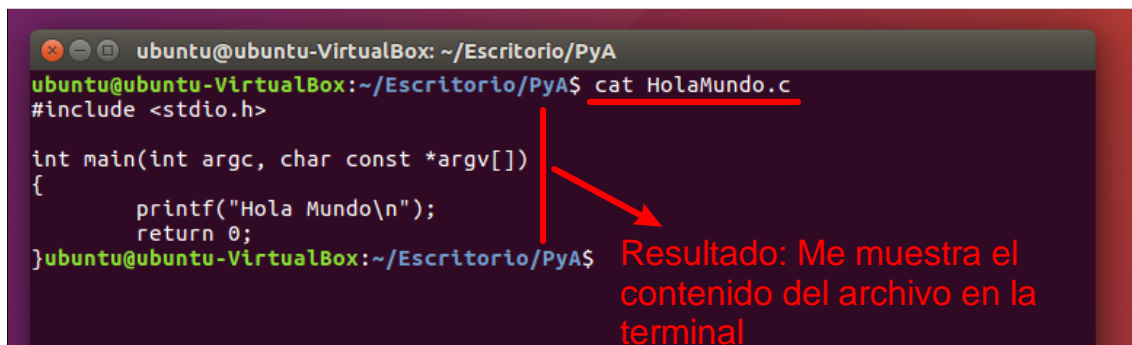


```
~/Escritorio/PyA/HolaMundo.c - Sublime Text (UNREGISTERED)
HolaMundo.c
1 #include <stdio.h>
2
3 int main(int argc, char const *argv[])
4 {
5     printf("Hola Mundo\n");
6     return 0;
7 }
8
9
```

Opción 3: Se pueden usar otros editores de texto, como GEdit.

9. ¿Cómo veo el contenido de un archivo desde la consola?

Comando: `cat <nombre_de_archivo>`



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ cat HolaMundo.c
#include <stdio.h>

int main(int argc, char const *argv[])
{
    printf("Hola Mundo\n");
    return 0;
}
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

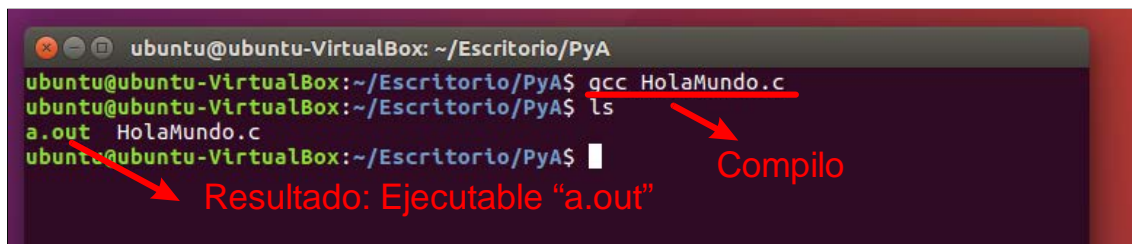
Resultado: Me muestra el contenido del archivo en la terminal

10. IMPORTANTE: COMPILAR

- Usamos el compilador `gcc`

Comando: `gcc <nombre_de_archivo.c>`

Resultado: Compila el archivo.c y genera un ejecutable con el nombre "a.out"



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ gcc HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
a.out  HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

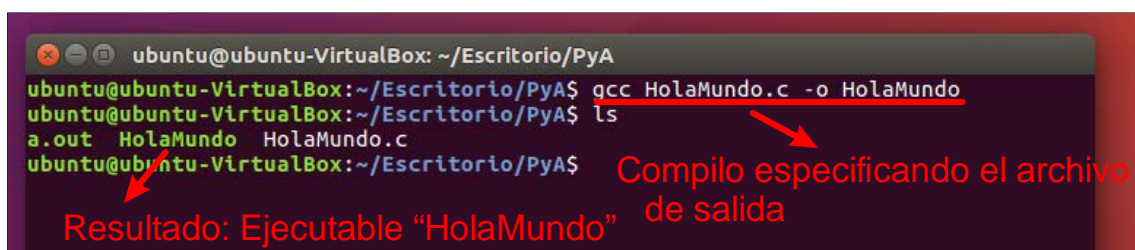
Resultado: Ejecutable "a.out"

Compilo

- Cambiar el nombre del archivo de salida

Comando: `gcc <nombre_de_archivo.c> -o <nombre_del_ejecutable>`

Las opciones para los programas que ejecutamos por líneas de comando se introducen con un guión, seguido de la opción. "-o" para el ejemplo anterior



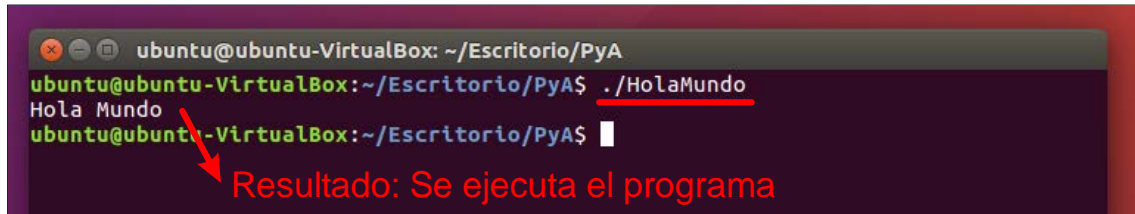
```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ gcc HolaMundo.c -o HolaMundo
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
a.out  HolaMundo  HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

Resultado: Ejecutable "HolaMundo"

Compilo especificando el archivo de salida

11. ¿Cómo ejecuto el programa compilado?

Comando: `./<nombre_del_programa>`



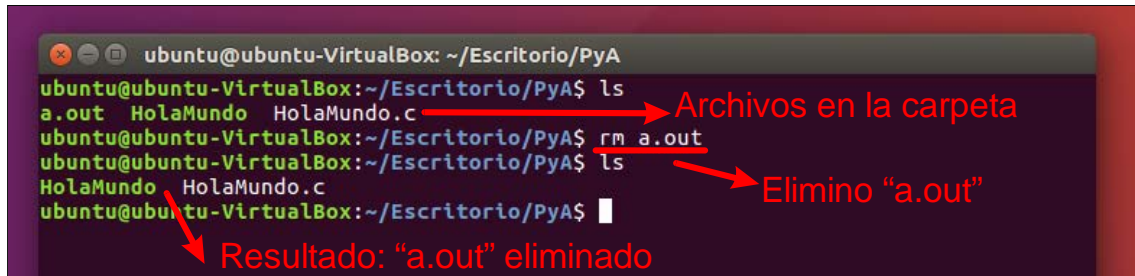
A terminal window titled 'ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA'. The prompt is 'ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA\$'. The user enters the command `./HolaMundo`. The output is 'Hola Mundo'. A red arrow points from the text 'Resultado: Se ejecuta el programa' to the command line.

```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ./HolaMundo
Hola Mundo
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

Resultado: Se ejecuta el programa

12. ¿Cómo elimino un archivo?

Comando: `rm <nombre_del_archivo>`



A terminal window titled 'ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA'. The user enters `ls`, showing 'a.out HolaMundo HolaMundo.c'. Then the user enters `rm a.out`. Then the user enters `ls`, showing 'HolaMundo HolaMundo.c'. A red arrow points from the text 'Archivos en la carpeta' to the first `ls` command. Another red arrow points from the text 'Elimino "a.out"' to the `rm a.out` command. A third red arrow points from the text 'Resultado: "a.out" eliminado' to the second `ls` command.

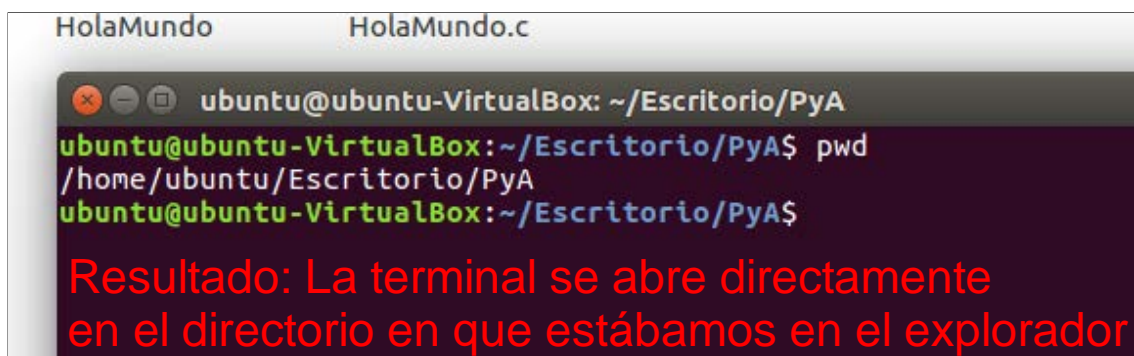
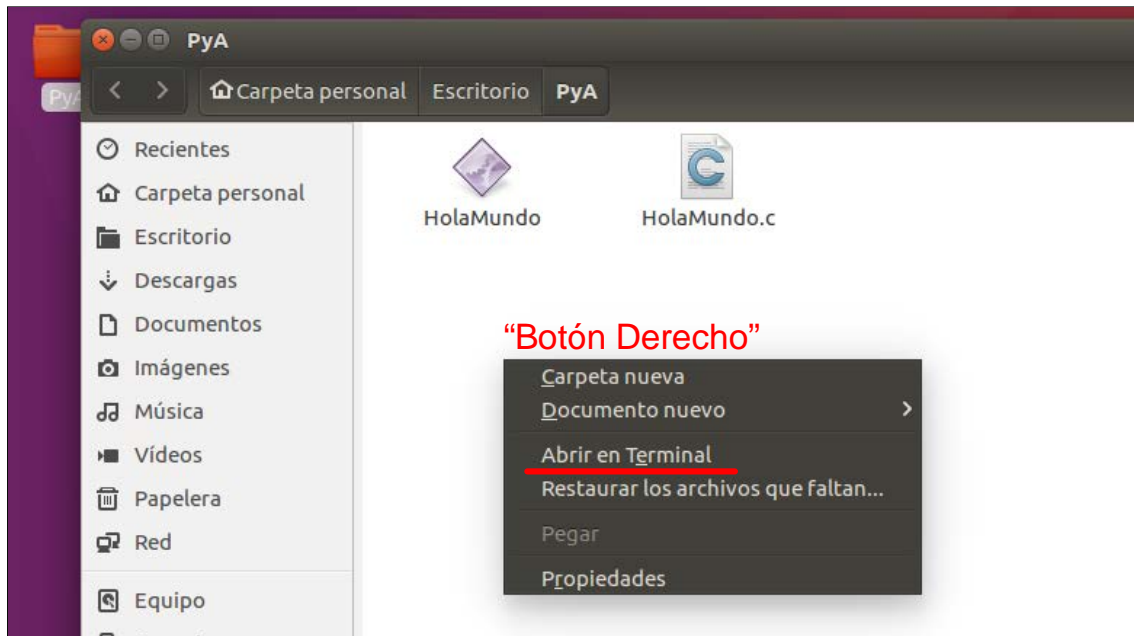
```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
a.out  HolaMundo  HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ rm a.out
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ ls
HolaMundo  HolaMundo.c
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$
```

Archivos en la carpeta

Elimino "a.out"

Resultado: "a.out" eliminado

13. Atajo desde el explorador de archivos para abrir una terminal directamente en una carpeta.



14. ÚTIL: consultar la entrada del manual de Linux para un comando específico

Comando: `man <consulta>`

```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ man ls
```

Ejemplo del manual para el comando ls

```
LS(1)                                User Commands                                LS(1)
NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
  fied.

  Mandatory arguments to long options are mandatory for short options
  too.

  -a, --all
      do not ignore entries starting with .

  -A, --almost-all
      do not list implied . and ..

  --author

Manual page ls(1) line 1 (press h for help or q to quit)
```

Ejemplo para el comando "ls"

La sinopsis explica la forma de ejecutar el comando

En la descripción se puede ver qué hace el comando y las distintas opciones

Apretar "q" para salir

El manual de Linux tiene varias secciones, que se pueden ver en la entrada del manual del comando man:

Comando: `man man`



```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
La siguiente tabla muestra los números de sección del manual y los
tipos de páginas que contienen.

1  Programas ejecutables y guiones del intérprete de órdenes
2  Llamadas del sistema (funciones servidas por el núcleo)
3  Llamadas de la biblioteca (funciones contenidas en las bibliotecas
   del sistema)
4  Ficheros especiales (se encuentran generalmente en /dev)
5  Formato de ficheros y convenios p.ej. I/etc/passwd
6  Juegos
7  Paquetes de macros y convenios p.ej. man(7), groff(7).
8  Órdenes de administración del sistema (generalmente solo son para
   root)
9  Rutinas del núcleo [No estándar]
n  nuevo [obsoleto]
l  local [obsoleto]
p  público [obsoleto]
o  viejo [obsoleto]
```

Secciones

La entrada número 3 del manual es útil para programar, por ejemplo, para buscar el manual de "printf()"

¿Cómo entro a otra sección del manual?

Comando: `man <numero_de_seccion> <consulta>`

Ejemplo: “printf()”

```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
ubuntu@ubuntu-VirtualBox:~/Escritorio/PyA$ man 3 printf
```

```
ubuntu@ubuntu-VirtualBox: ~/Escritorio/PyA
PRINTF(3)                                Linux Programmer's Manual                                PRINTF(3)
NAME                                     Manual del programador de Linux 🦾
printf, fprintf, dprintf, sprintf, snprintf, vprintf, vfprintf,
vdprintf, vsprintf, vsnprintf - formatted output conversion
SYNOPSIS
#include <stdio.h>

int printf(const char *format, ...);
int fprintf(FILE *stream, const char *format, ...);
int dprintf(int fd, const char *format, ...);
int sprintf(char *str, const char *format, ...);
int snprintf(char *str, size_t size, const char *format, ...);

#include <stdarg.h>

int vprintf(const char *format, va_list ap);
int vfprintf(FILE *stream, const char *format, va_list ap);
int vdprintf(int fd, const char *format, va_list ap);
int vsprintf(char *str, const char *format, va_list ap);
int vsnprintf(char *str, size_t size, const char *format, va_list ap);

Manual page printf(3) line 1 (press h for help or q to quit)
```