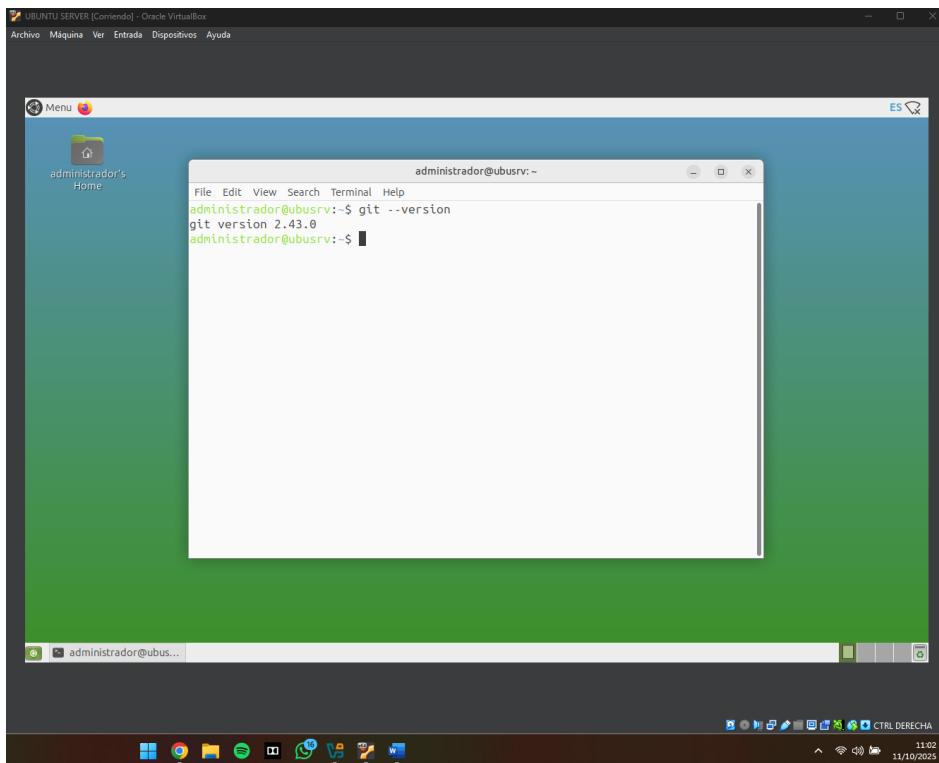


ACTIVIDAD GIT

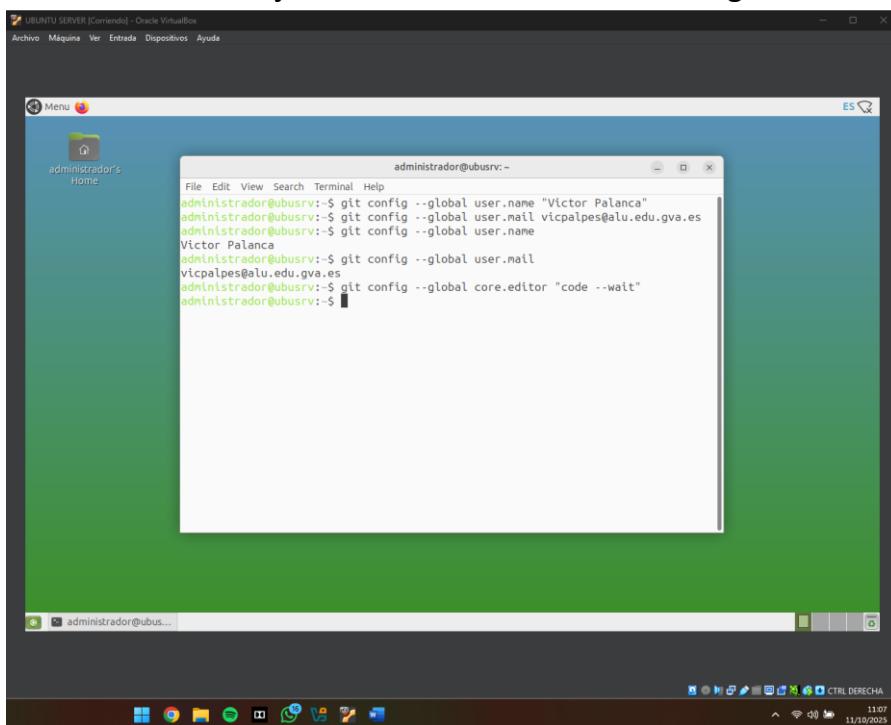
1. Instala Git en tu sistema operativo. Adjunta una captura de pantalla en la que aparezca el resultado de la ejecución del comando git --version.



```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Menu Firefox
administrador's Home
administrador@ubusrv:~$ git --version
git version 2.43.0
administrador@ubusrv:~$
```

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "administrador@ubusrv:~". Inside the terminal, the command "git --version" is run, and the output "git version 2.43.0" is displayed. The desktop background is green, and there are icons on the desktop. The taskbar at the bottom shows various application icons and the date/time as 11/10/2025.

2. Realiza la configuración de Git según lo indicado en el tema (nombre, correo electrónico y editor de preferencia). Adjunta una captura de pantalla con el resultado de la ejecución de los comandos de configuración.



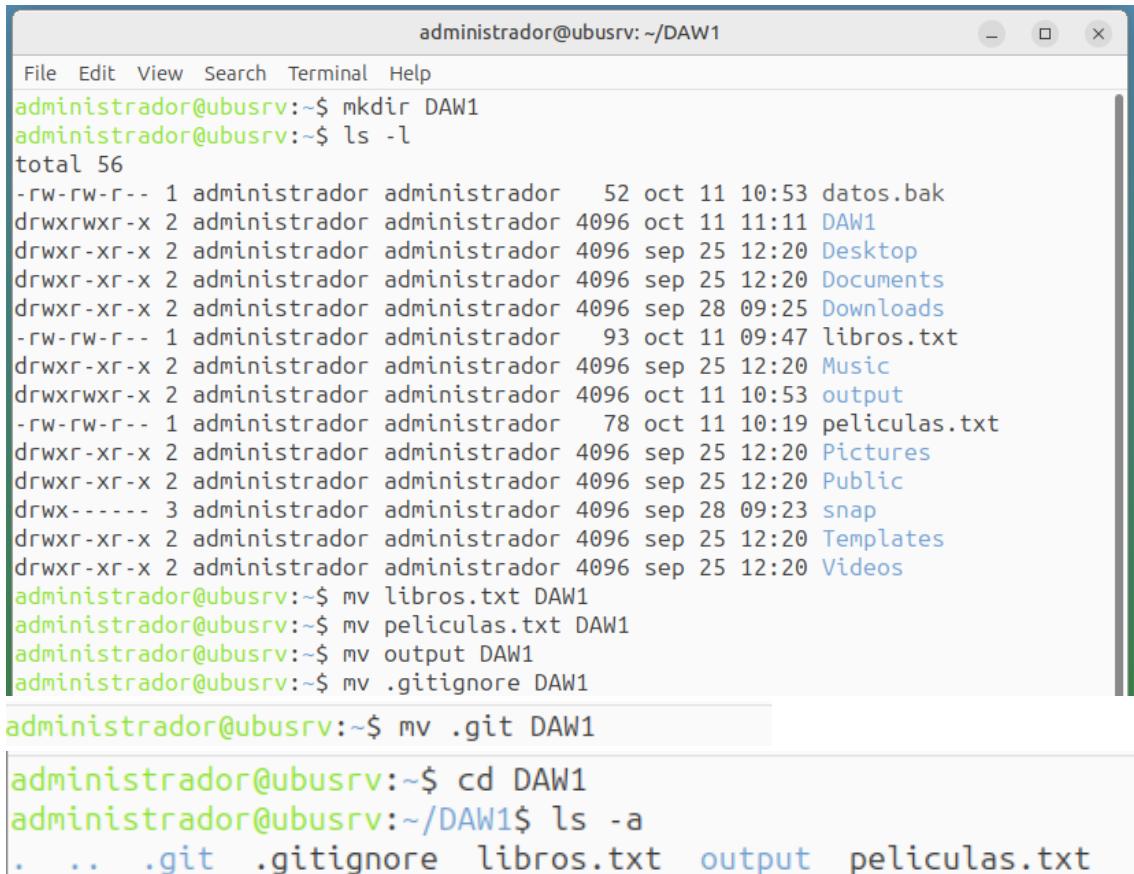
```
UBUNTU SERVER [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Menu Firefox
administrador's Home
administrador@ubusrv:~$ git config --global user.name "Victor Palanca"
administrador@ubusrv:~$ git config --global user.mail vcpalpes@alu.edu.gva.es
administrador@ubusrv:~$ git config --global user.name
Victor Palanca
administrador@ubusrv:~$ git config --global user.mail
vcpalpes@alu.edu.gva.es
administrador@ubusrv:~$ git config --global core.editor "code --wait"
administrador@ubusrv:~$
```

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "administrador@ubusrv:~". Inside the terminal, three commands are run sequentially to set global Git configuration: "git config --global user.name", "git config --global user.mail", and "git config --global core.editor". The desktop background is green, and there are icons on the desktop. The taskbar at the bottom shows various application icons and the date/time as 11/10/2025.

3. Crea una carpeta denominada DAW1. Realiza las siguientes acciones en ella:

*** Al terminar la actividad completa me di cuenta de que no había realizado las instrucciones dentro de la carpeta DAW1, por lo que, al terminar, he creado la carpeta y he pasado todos los archivos. Sin embargo, habría que crear la carpeta DAW1 nada más empezar, iniciar git y trabajar desde ahí.

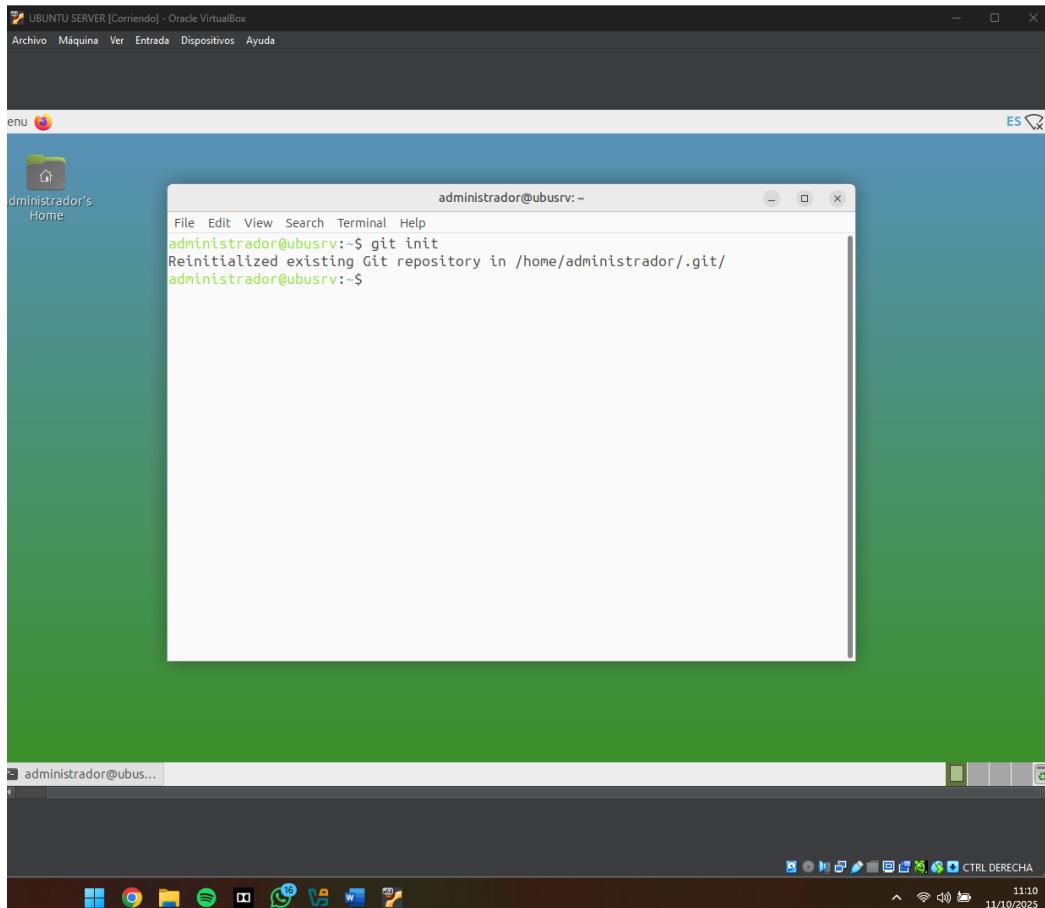
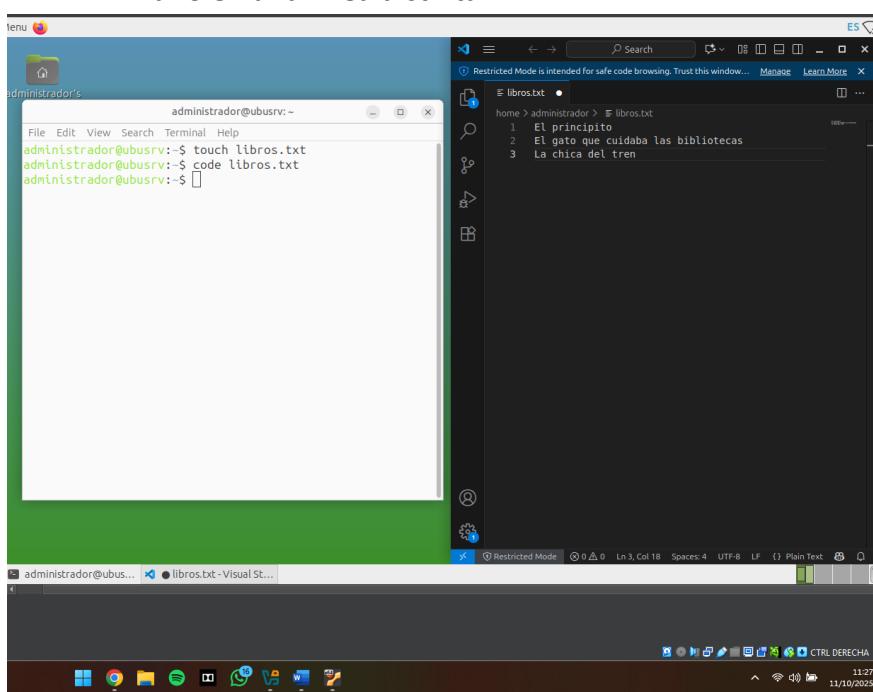
Cómo he realizado el traspaso al final:

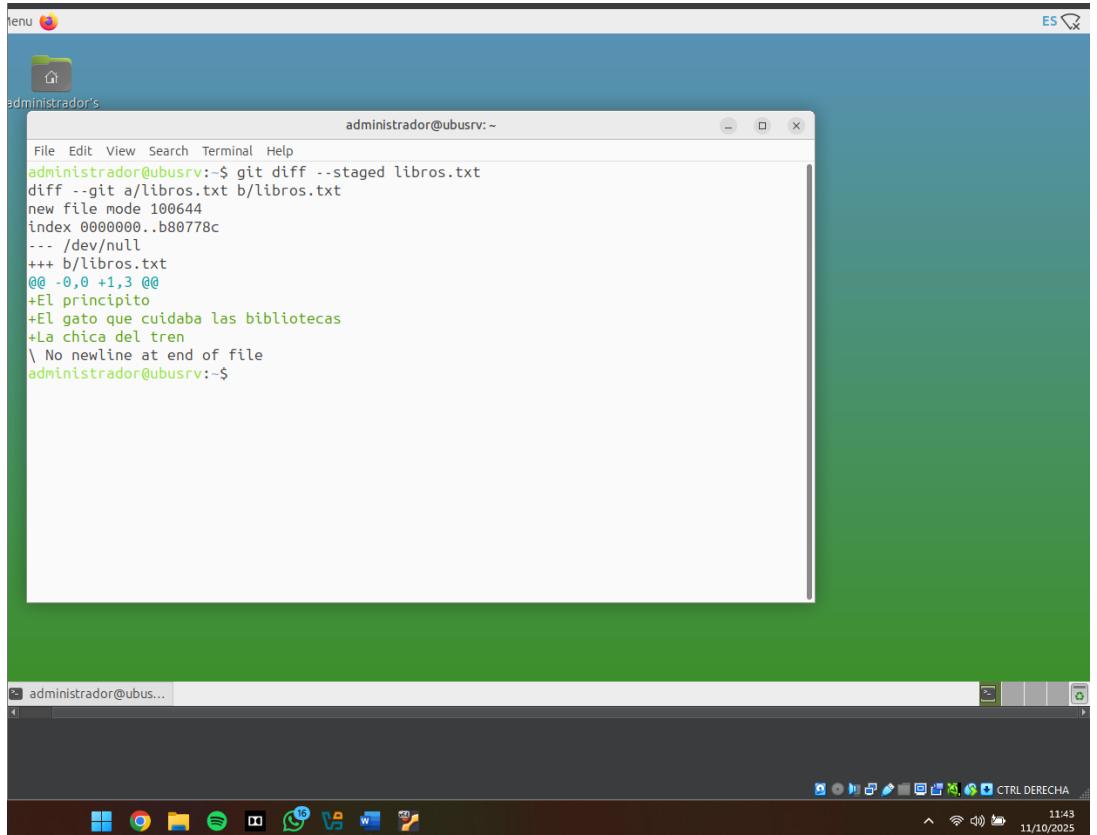


```
administrador@ubusrv: ~/DAW1
File Edit View Search Terminal Help
administrador@ubusrv:~$ mkdir DAW1
administrador@ubusrv:~$ ls -l
total 56
-rw-rw-r-- 1 administrador administrador 52 oct 11 10:53 datos.bak
drwxrwxr-x 2 administrador administrador 4096 oct 11 11:11 DAW1
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Desktop
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Documents
drwxr-xr-x 2 administrador administrador 4096 sep 28 09:25 Downloads
-rw-rw-r-- 1 administrador administrador 93 oct 11 09:47 libros.txt
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Music
drwxrwxr-x 2 administrador administrador 4096 oct 11 10:53 output
-rw-rw-r-- 1 administrador administrador 78 oct 11 10:19 peliculas.txt
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Pictures
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Public
drwx----- 3 administrador administrador 4096 sep 28 09:23 snap
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Templates
drwxr-xr-x 2 administrador administrador 4096 sep 25 12:20 Videos
administrador@ubusrv:~$ mv libros.txt DAW1
administrador@ubusrv:~$ mv peliculas.txt DAW1
administrador@ubusrv:~$ mv output DAW1
administrador@ubusrv:~$ mv .gitignore DAW1
administrador@ubusrv:~$ mv .git DAW1
administrador@ubusrv:~$ cd DAW1
administrador@ubusrv:~/DAW1$ ls -a
. ... .git .gitignore libros.txt output peliculas.txt
```

a) Crea un repositorio Git.

Aquí ya había creado el repositorio.

**b) Crea un fichero denominado libros.txt. Añade tres títulos de libros cada uno en una línea distinta.**



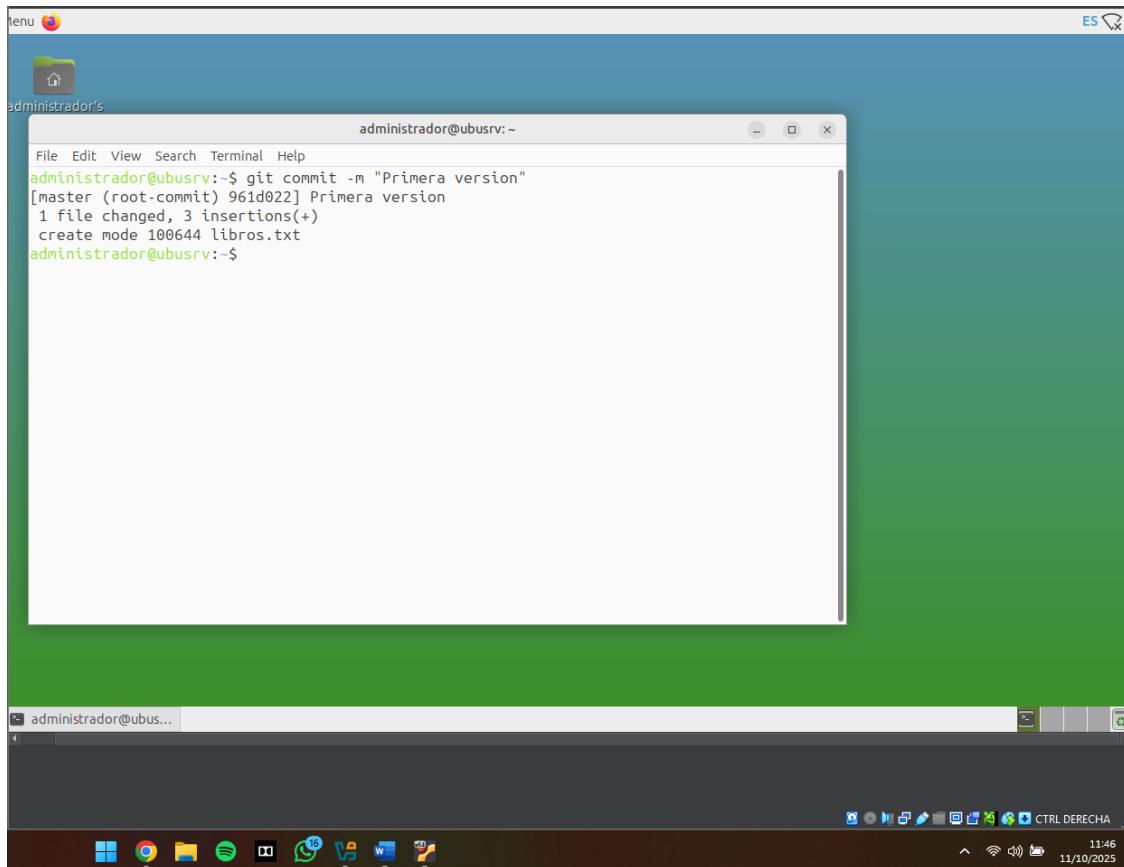
terminal

administrador@ubusrv:~

```
File Edit View Search Terminal Help
administrador@ubusrv:~$ git diff --staged libros.txt
diff --git a/libros.txt b/libros.txt
new file mode 100644
index 0000000..b80778c
--- /dev/null
+++ b/libros.txt
@@ -0,0 +1,3 @@
+El principio
+El gato que cuidaba las bibliotecas
+La chica del tren
\ No newline at end of file
administrador@ubusrv:~$
```

c) Haz un primer commit.

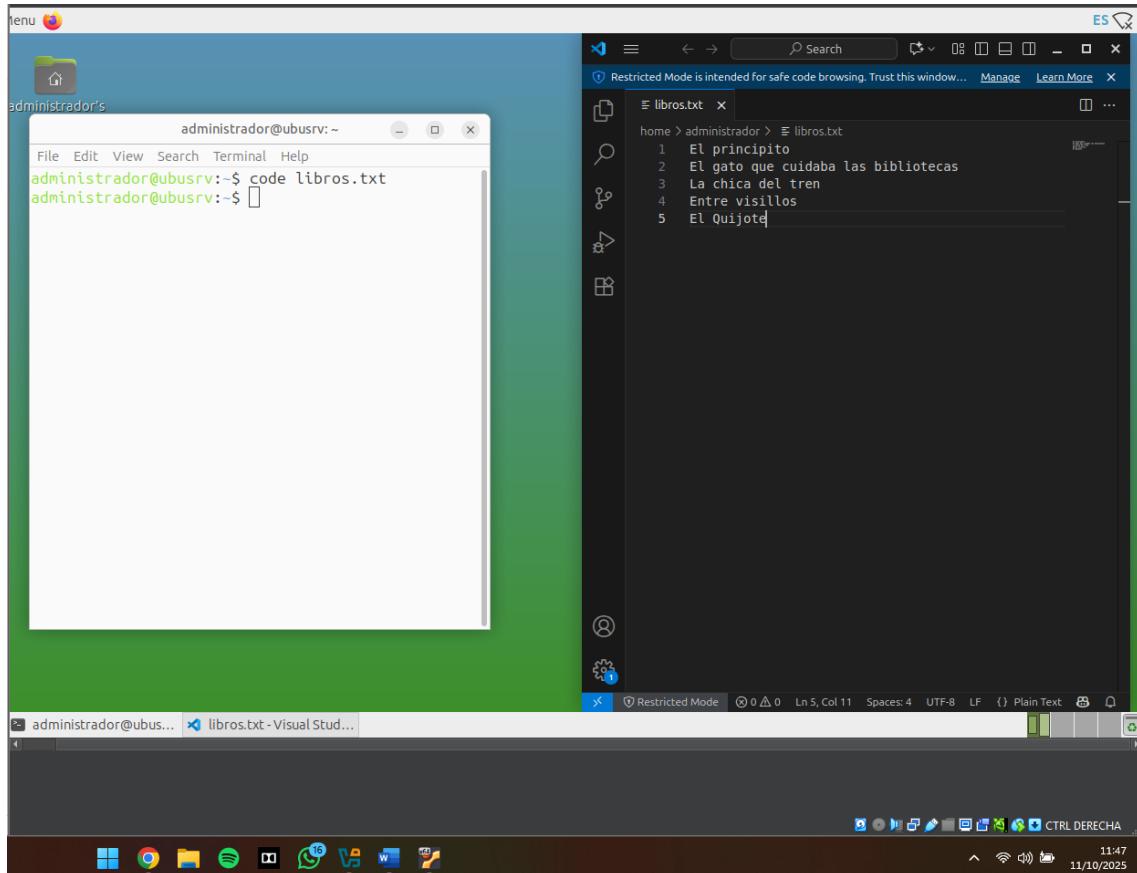
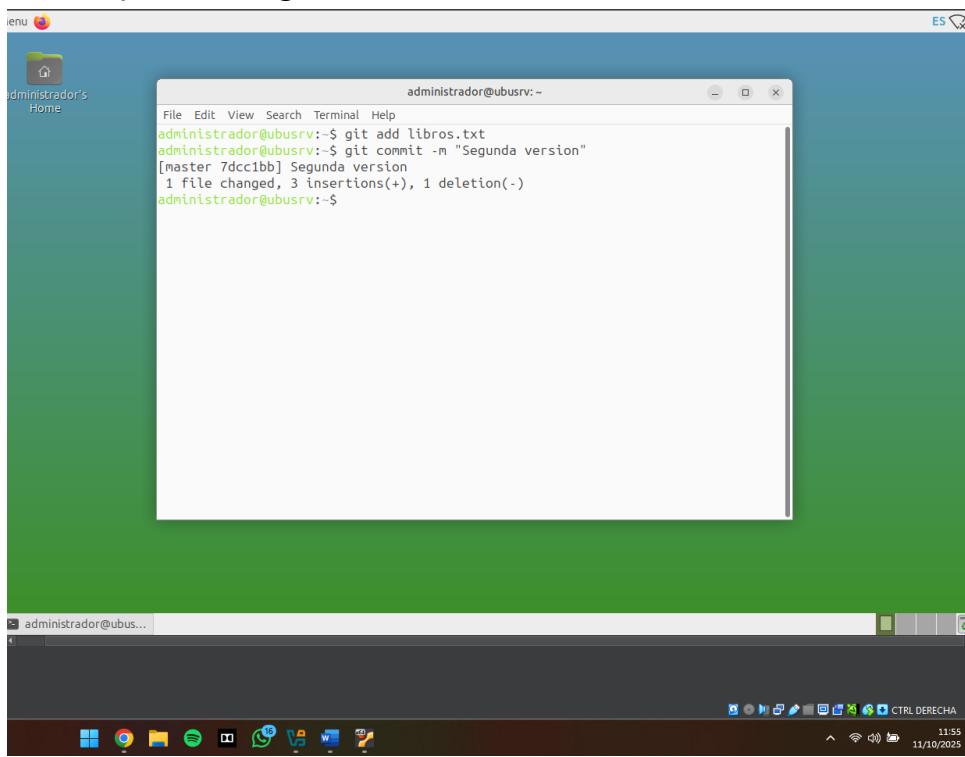
Previamente, antes de realizar el commit, ejecuté “git add libros.txt” para añadirlo al área de preparación. Le llamé, “Primera versión”



terminal

administrador@ubusrv:~

```
File Edit View Search Terminal Help
administrador@ubusrv:~$ git commit -m "Primera version"
[master (root-commit) 961d022] Primera version
 1 file changed, 3 insertions(+)
 create mode 100644 libros.txt
administrador@ubusrv:~$
```

d) Añade dos libros al archivo libros.txt.**e) Haz un segundo commit.**

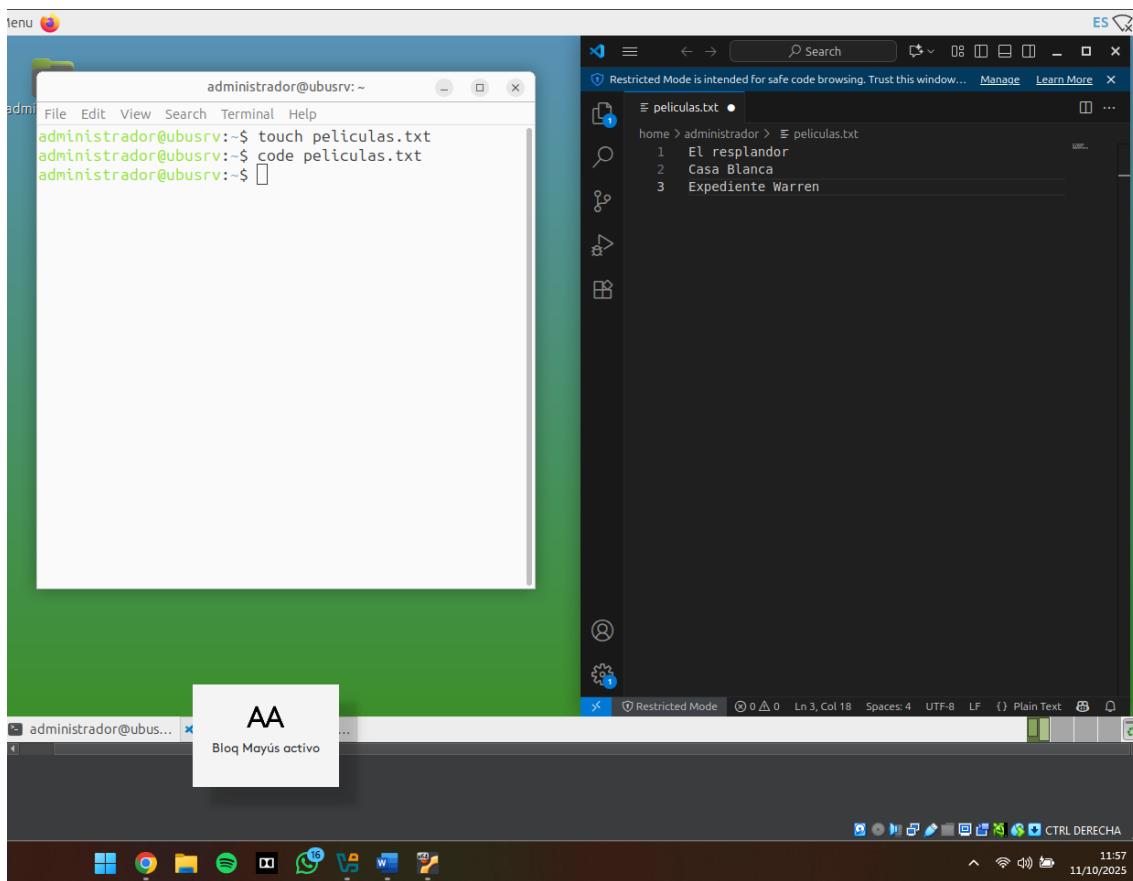
```
administrador@ubusrv:~$ git log
commit 7dcc1bb6a9f6f64eec72cfe27aa498e8a028a0d0 (HEAD -> master)
Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
Date:   Sat Oct 11 09:54:52 2025 +0000

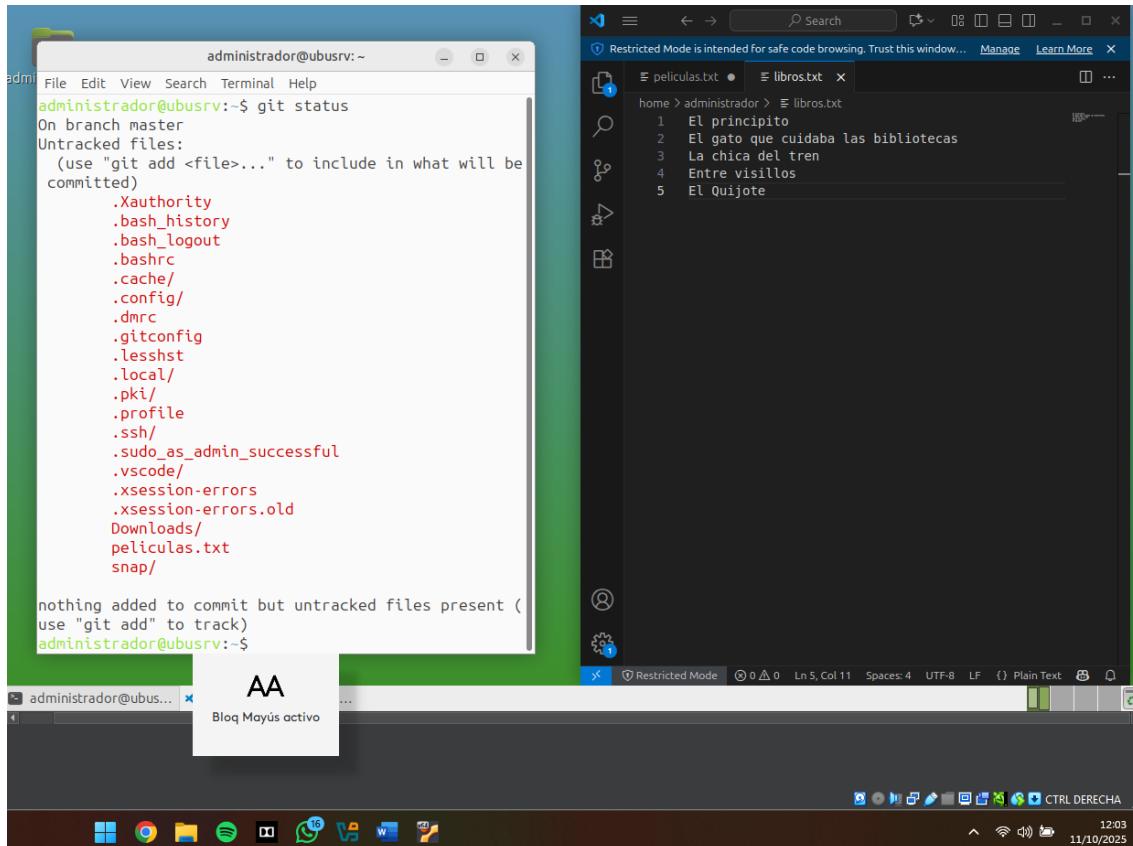
    Segunda version

commit 961d022762077c081098b612709140067f36b97d
Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
Date:   Sat Oct 11 09:46:23 2025 +0000

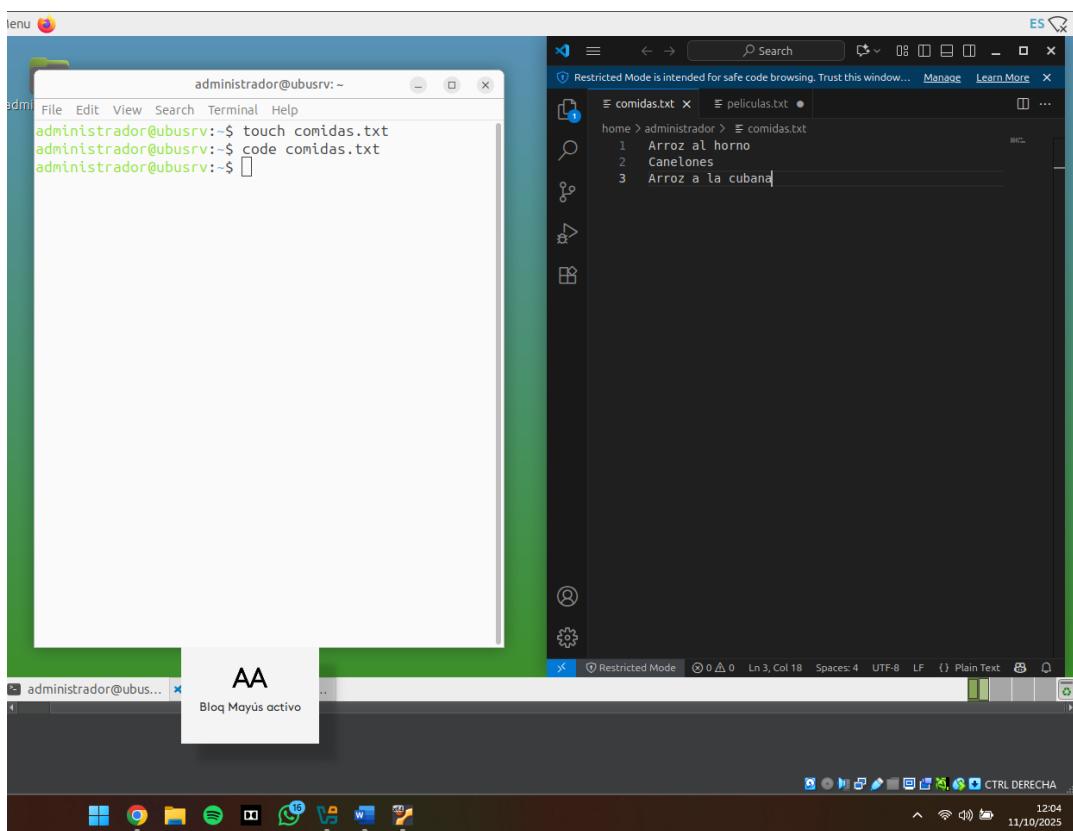
    Primera version
administrador@ubusrv:~$
```

f) Crea un fichero denominado películas.txt. Añade tres títulos de películas a dicho archivo.



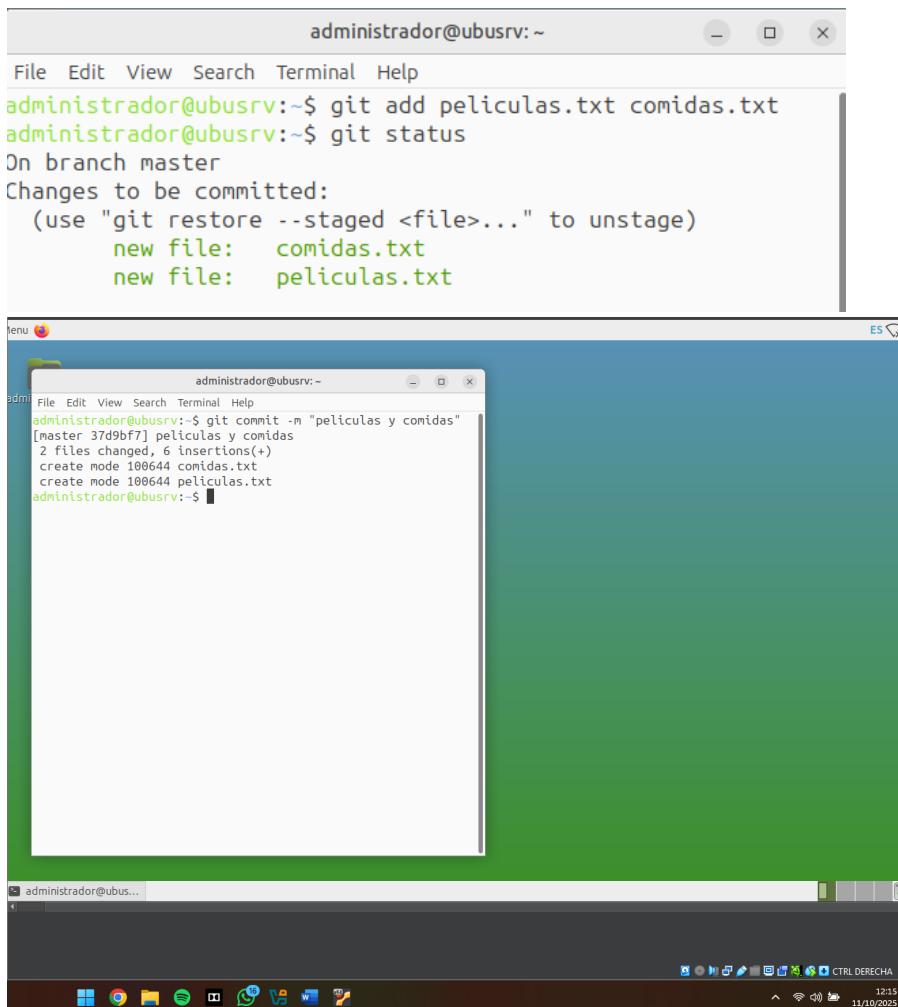
g) Haz una captura de pantalla del comando git status.

The screenshot shows a terminal window titled "administrador@ubusrv:~" displaying the output of the command "git status". The terminal shows a list of untracked files, including ".Xauthority", ".bash_history", ".bash_logout", ".bashrc", ".cache/", ".config/", ".dmrc", ".gitconfig", ".lesshtf", ".local/", ".pki/", ".profile", ".ssh/", ".sudo_as_admin_successful", ".vscode/", ".xsession-errors", ".xsession-errors.old", "Downloads/", "peliculas.txt", and "snap". It also indicates that nothing has been added to commit but there are untracked files present. Below the terminal is a file manager window showing two files: "peliculas.txt" and "libros.txt". The "peliculas.txt" file contains a list of five books: "El principito", "El gato que cuidaba las bibliotecas", "La chica del tren", "Entre visillos", and "El Quijote". The desktop taskbar at the bottom shows various application icons.

h) Crea un fichero denominado comidas.txt. Añade tres nombres de comidas a dicho archivo.

The screenshot shows a terminal window titled "administrador@ubusrv:~" with the command "code comidas.txt" entered. Below the terminal is a file manager window showing three files: "comidas.txt", "peliculas.txt", and "libros.txt". The "comidas.txt" file contains a list of three items: "Arroz al horno", "Canelones", and "Arroz a la cubana". The desktop taskbar at the bottom shows various application icons.

i) Haz un tercer commit que incluya los archivos películas.txt y comidas.txt.



```
administrador@ubusrv:~
```

```
File Edit View Search Terminal Help
```

```
administrador@ubusrv:~$ git add peliculas.txt comidas.txt
```

```
administrador@ubusrv:~$ git status
```

```
On branch master
```

```
Changes to be committed:
```

```
(use "git restore --staged <file>..." to unstage)
```

```
  new file:   comidas.txt
```

```
  new file:   peliculas.txt
```



```
menu ES
```

```
administrador@ubusrv:~
```

```
File Edit View Search Terminal Help
```

```
administrador@ubusrv:~$ git commit -m "películas y comidas"
```

```
[master 37d9bf7] películas y comidas
```

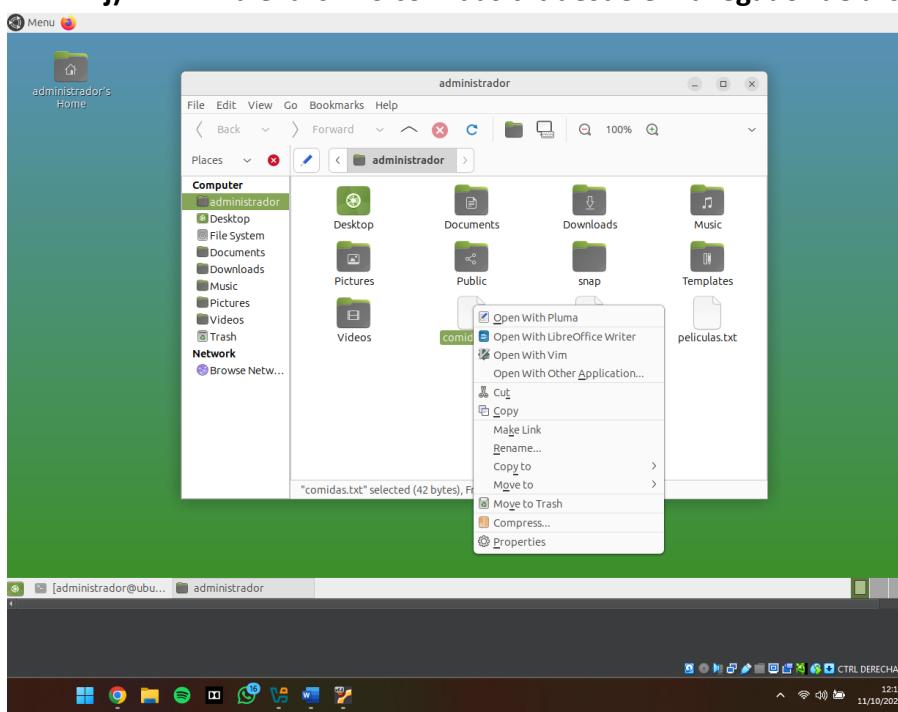
```
2 files changed, 6 insertions(+)
```

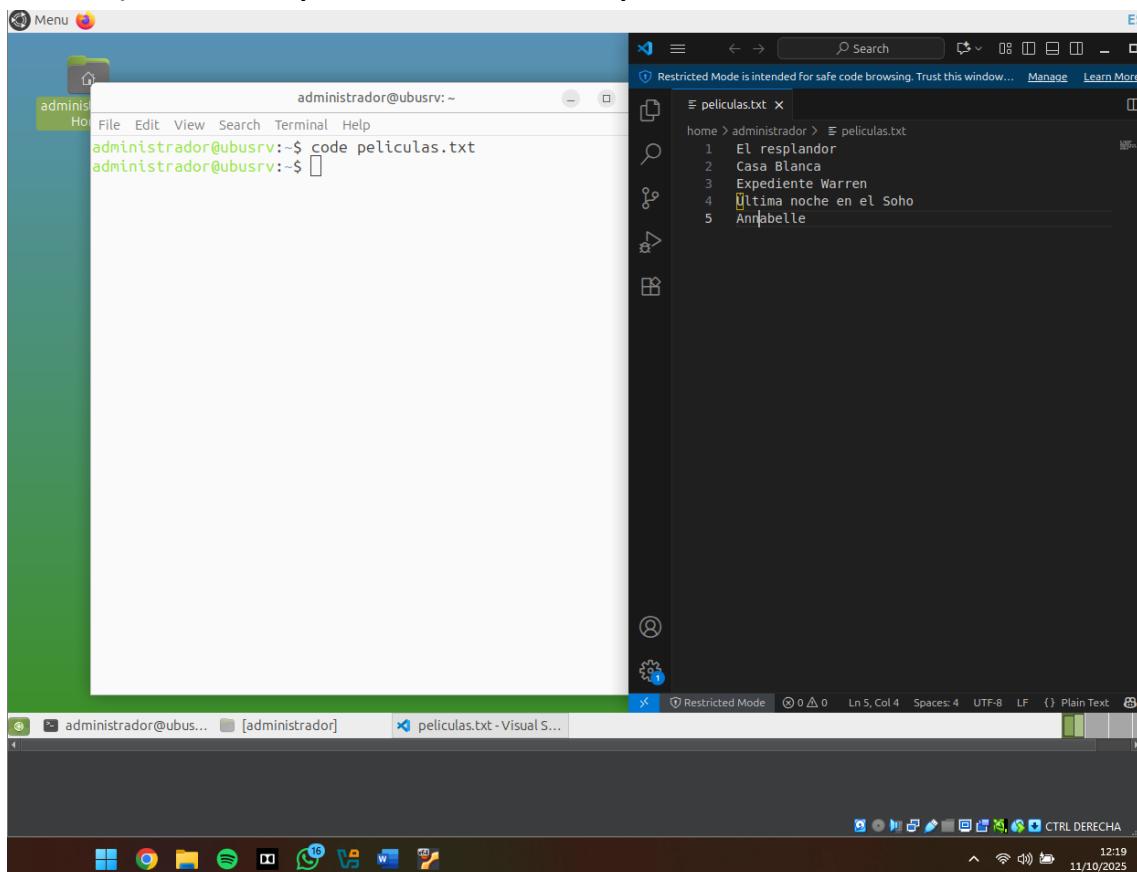
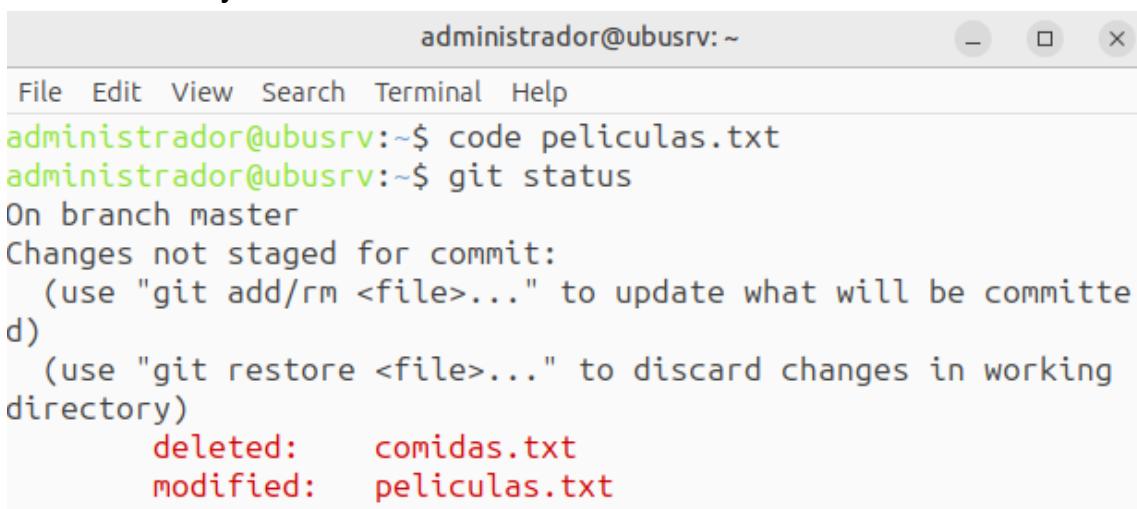
```
create mode 100644 comidas.txt
```

```
create mode 100644 películas.txt
```

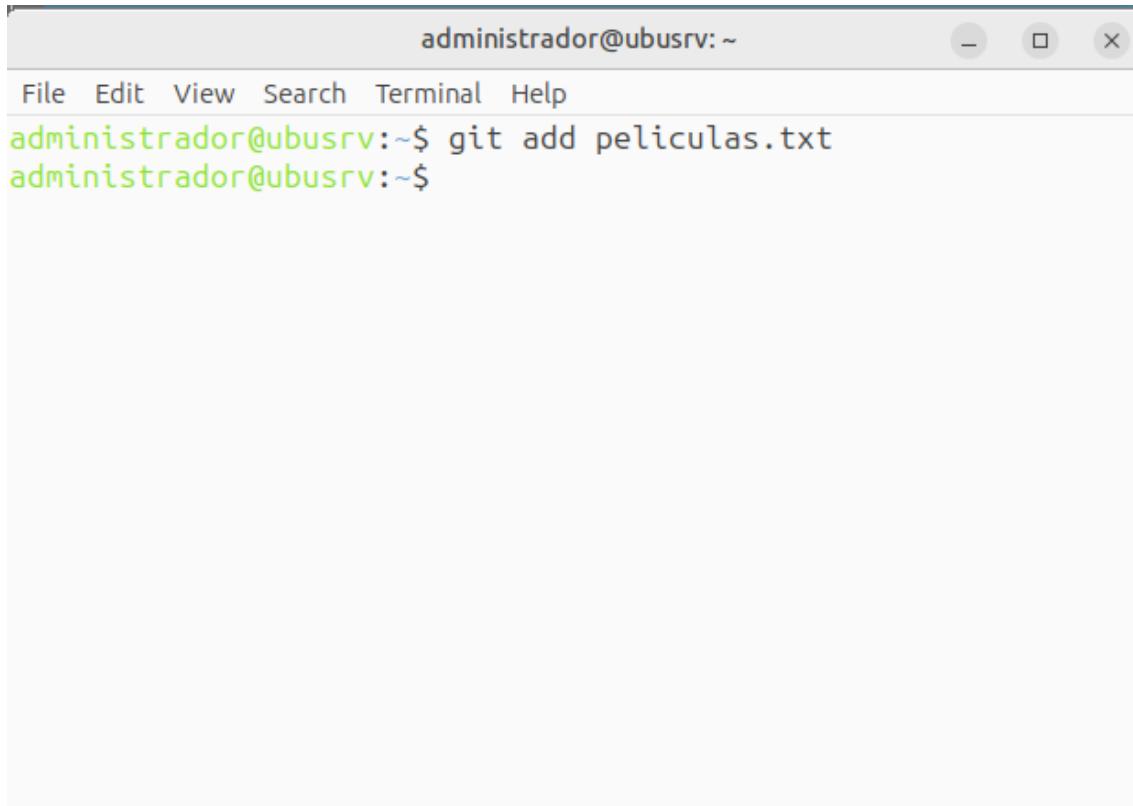
```
administrador@ubusrv:~$
```

j) Elimina el archivo comidas.txt desde el navegador de archivos.

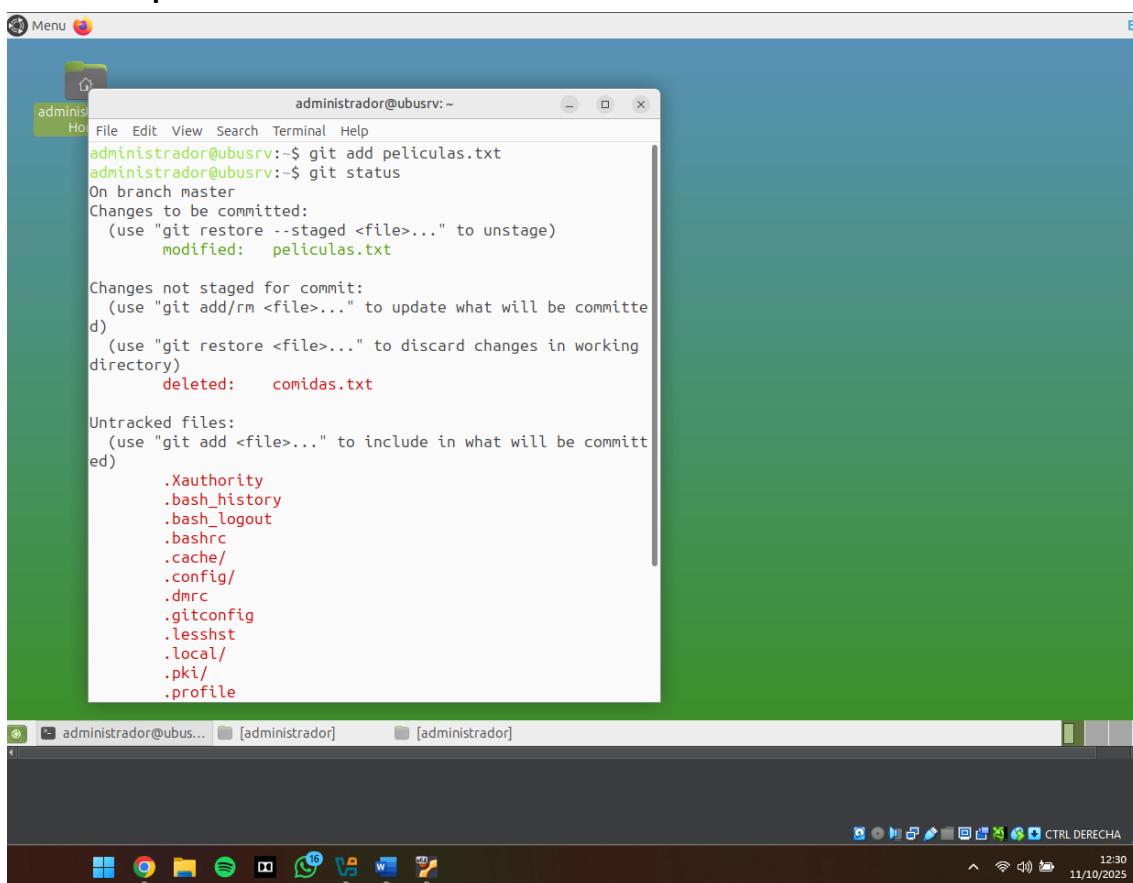


k) Añade dos películas más al archivo peliculas.txt.**l) Haz una captura de pantalla que muestre los cambios en el directorio de trabajo.**

```
administrador@ubusrv: ~
File Edit View Search Terminal Help
administrador@ubusrv:~$ code peliculas.txt
administrador@ubusrv:~$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      deleted:    comidas.txt
      modified:   peliculas.txt
```

m) Añade los cambios al área de preparación.

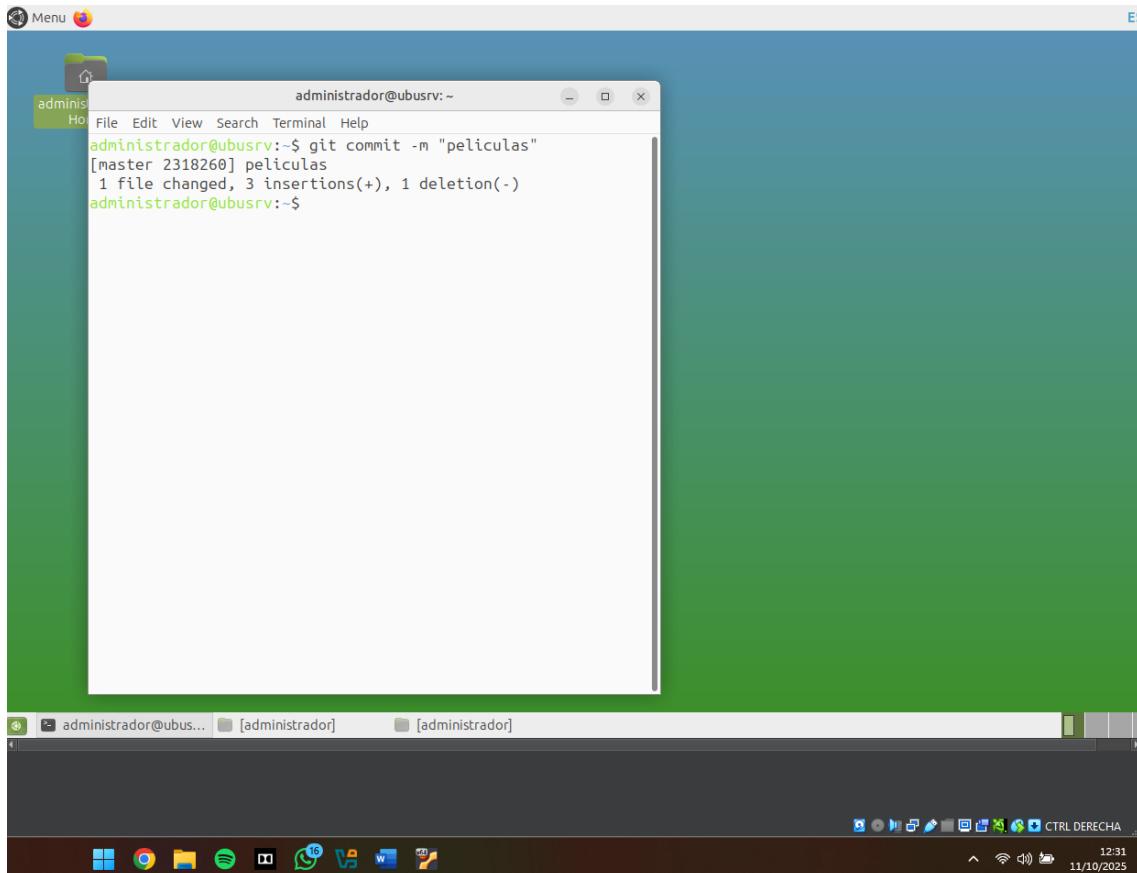
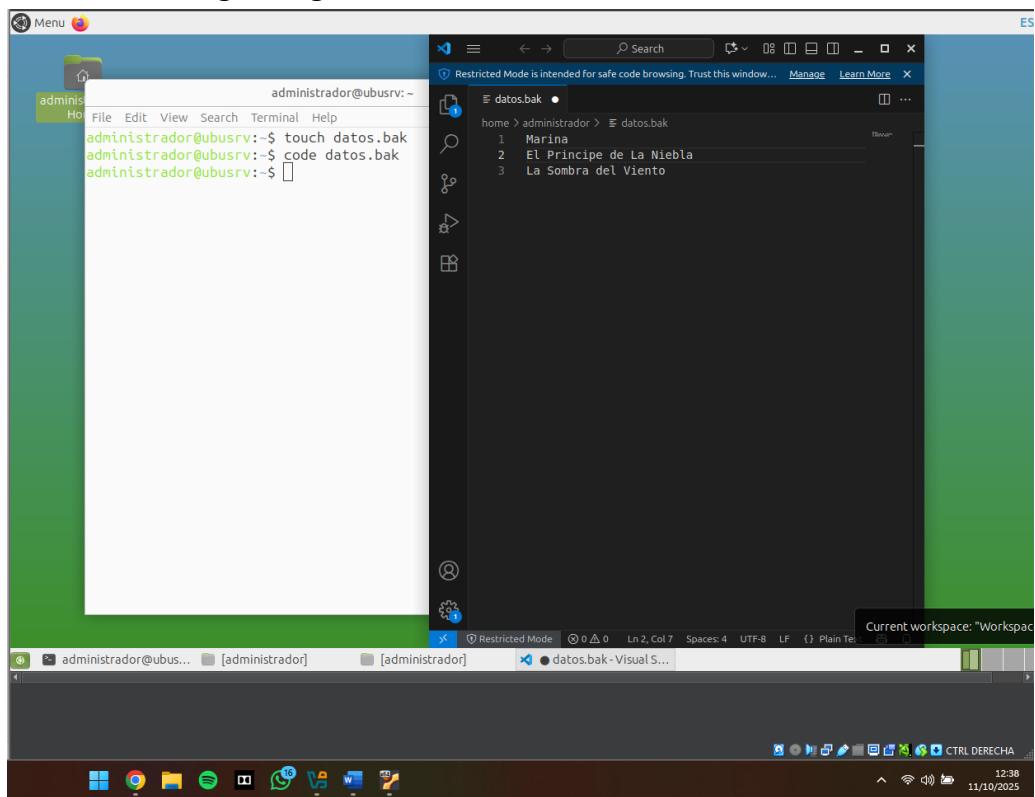
```
administrador@ubusrv: ~
File Edit View Search Terminal Help
administrador@ubusrv:~$ git add peliculas.txt
administrador@ubusrv:~$
```

n) Haz una captura de pantalla del comando git status. Debe indicar que se ha borrado el archivo comidas.txt y que se ha modificado el archivo peliculas.txt.

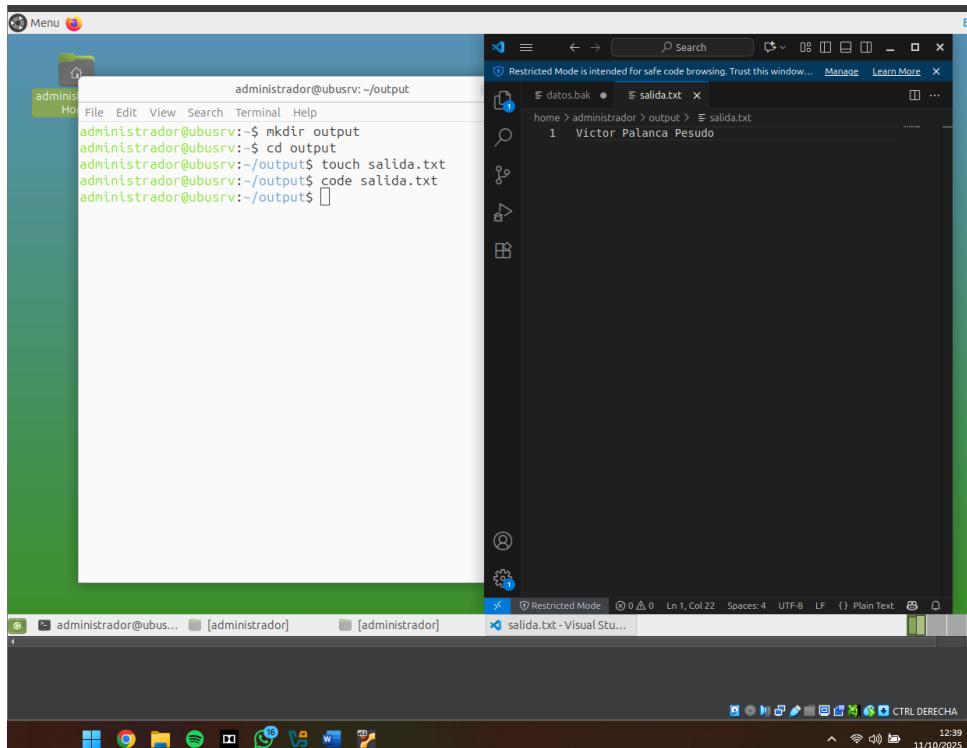
```
administrador@ubusrv: ~
File Edit View Search Terminal Help
administrador@ubusrv:~$ git add peliculas.txt
administrador@ubusrv:~$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   peliculas.txt

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      deleted:    comidas.txt

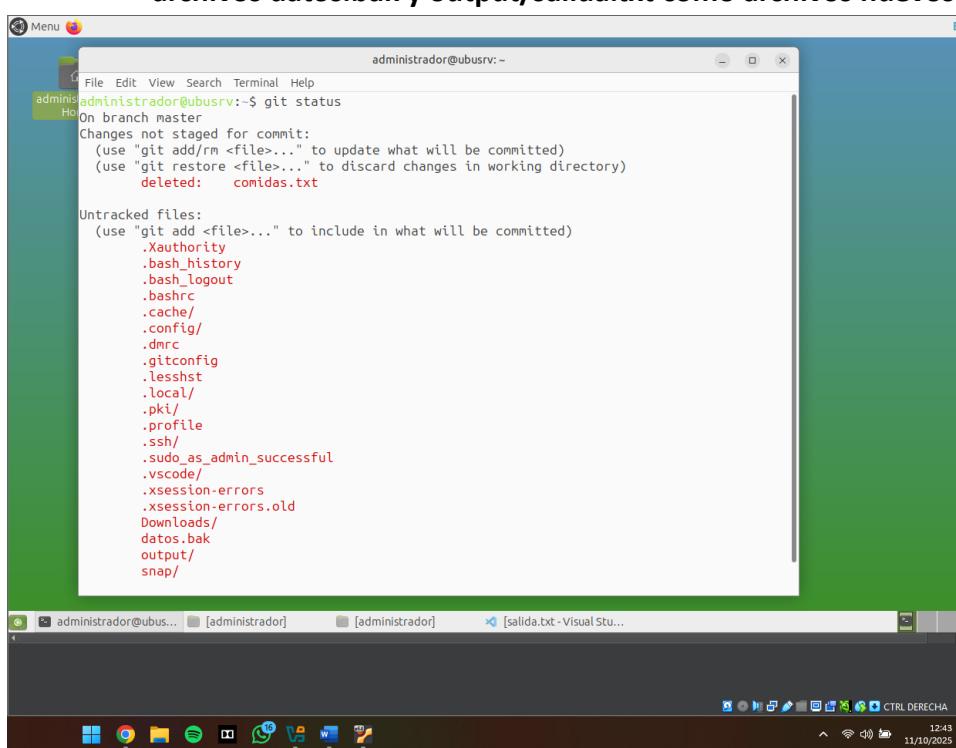
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    .Xauthority
    .bash_history
    .bash_logout
    .bashrc
    .cache/
    .config/
    .dmrc
    .gitconfig
    .lessht
    .local/
    .pki/
    .profile
```

o) Haz un cuarto commit.**p) Crea un archivo denominado datos.bak. Añade tres títulos de libros a dicho archivo. ¡IMPORTANTE! No añadas el archivo al área de preparación ni hagas ningún commit.**

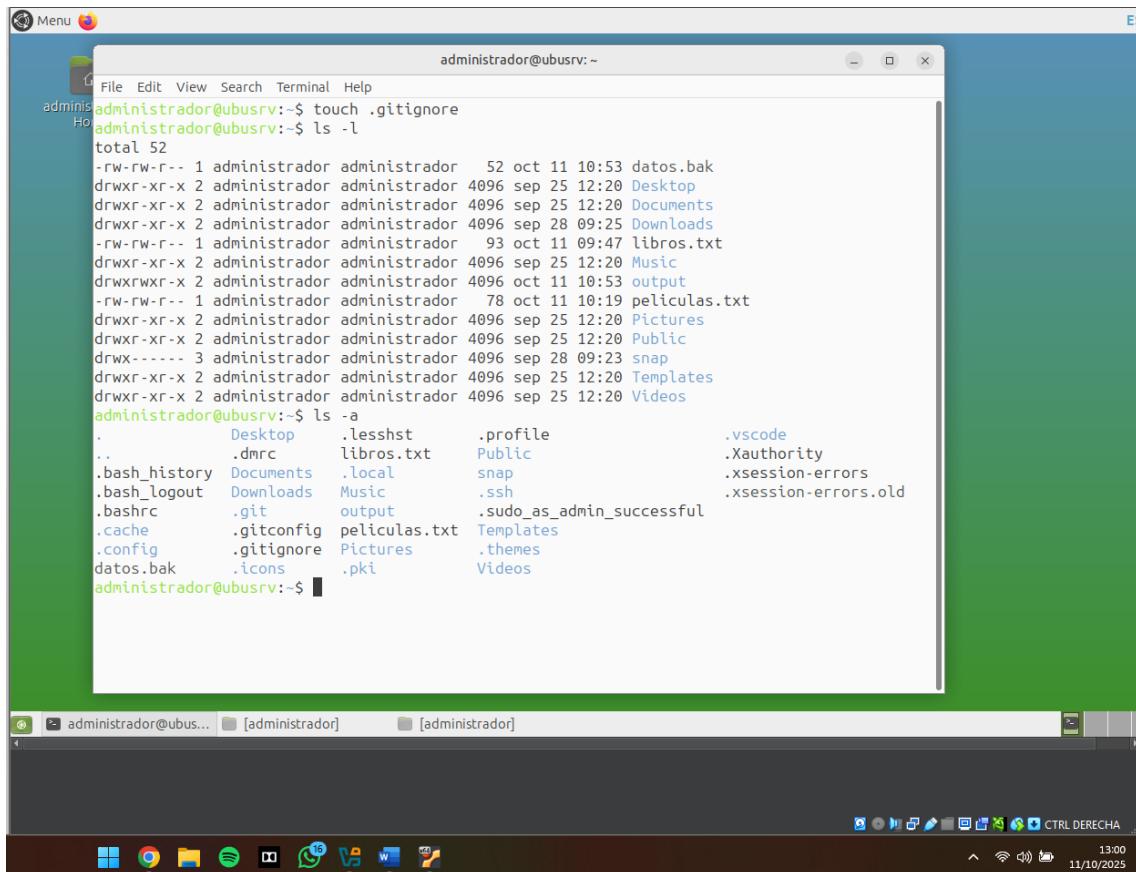
- q) Crea una subcarpeta denominada output. Crea un archivo denominado salida.txt en su interior. Escribe tu nombre y apellidos en dicho archivo. ¡IMPORTANTE! No añadas los archivos al área de preparación ni hagas ningún commit.**



- r) Haz una captura de pantalla del comando git status. Deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos (color rojo).**

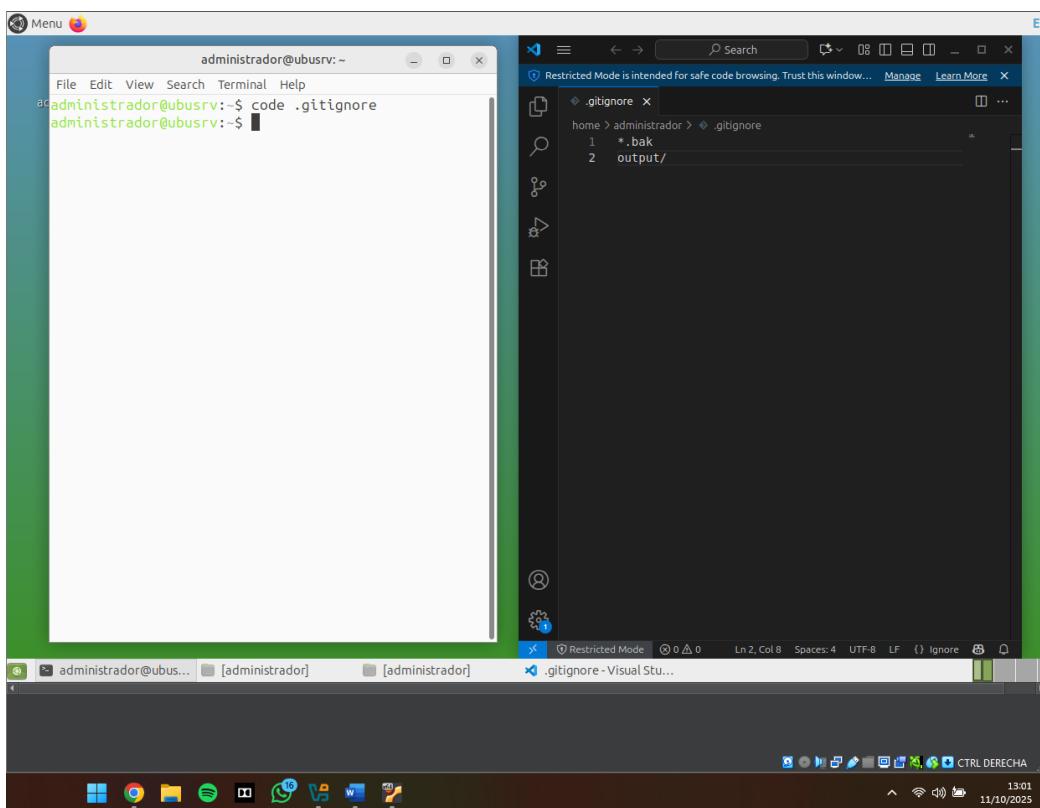


s) Crea un archivo .gitignore para que los ficheros con extensión .bak y el contenido de la carpeta output/ no se incluyan en el repositorio.



The screenshot shows a Linux desktop environment. In the top-left corner, there's a terminal window titled "administrador@ubusrv:~". Inside, the user runs two commands: "touch .gitignore" and "ls -l". The output of "ls -l" shows a directory structure with files like "datos.bak", "Desktop", "libros.txt", etc., and a file named ".gitignore". In the bottom-right corner, there's a file manager window showing a ".gitignore" file with the following content:

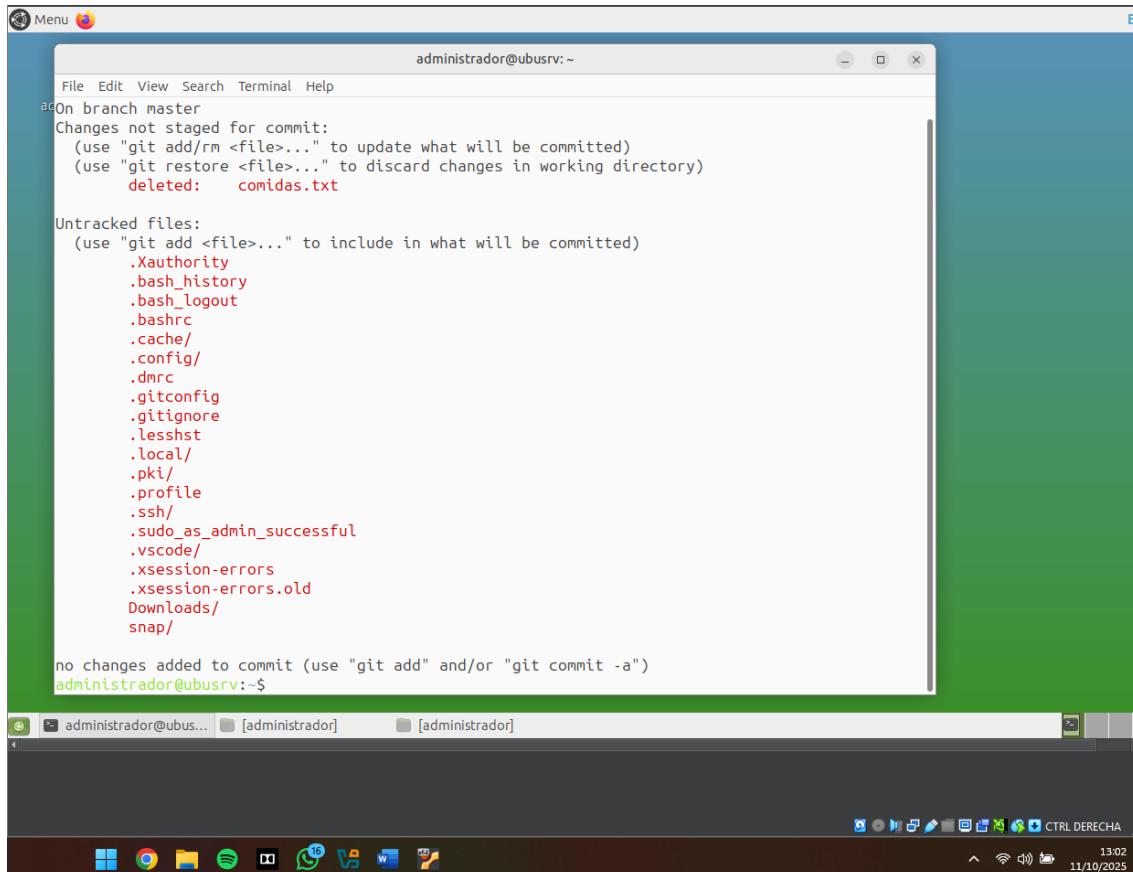
```
home > administrador > .gitignore
1 *.bak
2 output/
```



The screenshot shows a Windows desktop environment. In the top-left corner, there's a terminal window titled "administrador@ubusrv:~". Inside, the user runs the command "code .gitignore". In the bottom-right corner, there's a file manager window showing a ".gitignore" file with the following content:

```
home > administrador > .gitignore
1 *.bak
2 output/
```

- t) Haz una nueva captura de pantalla del comando git status. Ahora no deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos, sino que en su lugar debe aparecer únicamente el archivo .gitignore.**



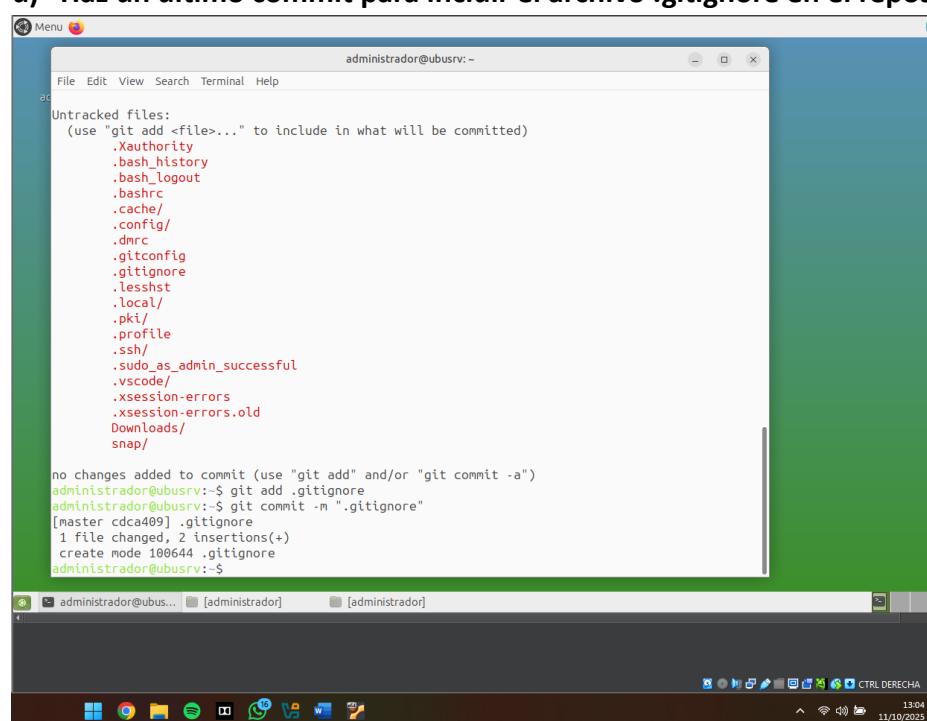
```
Menu ES
administrador@ubusrv: ~
File Edit View Search Terminal Help
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:      comidas.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .Xauthority
  .bash_history
  .bash_logout
  .bashrc
  .cache/
  .config/
  .dmrc
  .gitconfig
  .gitignore
  .lessht
  .local/
  .pki/
  .profile
  .ssh/
  .sudo_as_admin_successful
  .vscode/
  .xsession-errors
  .xsession-errors.old
  Downloads/
  snap/

no changes added to commit (use "git add" and/or "git commit -a")
administrador@ubusrv:~$
```

The terminal window is titled "administrador@ubusrv: ~". The output shows the current branch is "master". It lists changes not staged for commit, specifically the deletion of "comidas.txt". It also lists untracked files including ".Xauthority", ".bash_history", ".bash_logout", ".bashrc", ".cache/", ".config/", ".dmrc", ".gitconfig", ".gitignore", ".lessht", ".local/", ".pki/", ".profile", ".ssh/", ".sudo_as_admin_successful", ".vscode/", ".xsession-errors", ".xsession-errors.old", "Downloads/", and "snap/". A message at the bottom indicates "no changes added to commit (use "git add" and/or "git commit -a")".

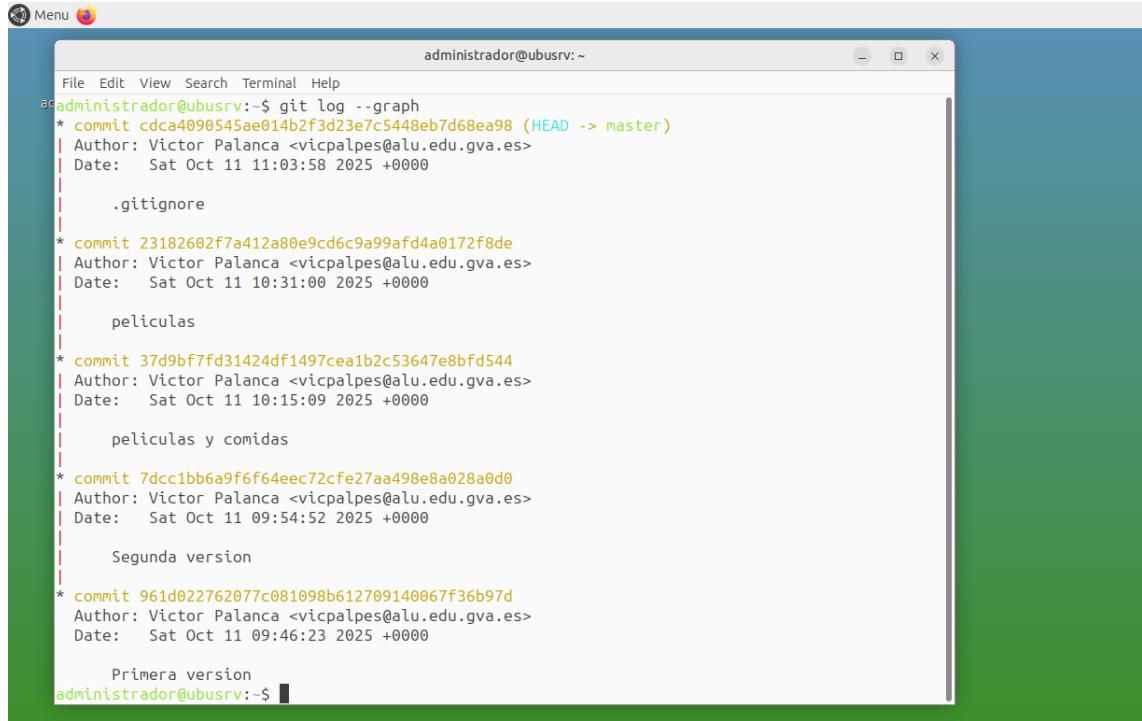
- u) Haz un último commit para incluir el archivo .gitignore en el repositorio.**



```
Menu ES
administrador@ubusrv: ~
File Edit View Search Terminal Help
Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .Xauthority
  .bash_history
  .bash_logout
  .bashrc
  .cache/
  .config/
  .dmrc
  .gitconfig
  .gitignore
  .lessht
  .local/
  .pki/
  .profile
  .ssh/
  .sudo_as_admin_successful
  .vscode/
  .xsession-errors
  .xsession-errors.old
  Downloads/
  snap/
no changes added to commit (use "git add" and/or "git commit -a")
administrador@ubusrv:~$ git add .gitignore
administrador@ubusrv:~$ git commit -m ".gitignore"
[master cdca409] .gitignore
 1 file changed, 2 insertions(+)
 create mode 100644 .gitignore
administrador@ubusrv:~$
```

The terminal window is titled "administrador@ubusrv: ~". The output shows the current branch is "[master]". It lists untracked files including ".Xauthority", ".bash_history", ".bash_logout", ".bashrc", ".cache/", ".config/", ".dmrc", ".gitconfig", ".gitignore", ".lessht", ".local/", ".pki/", ".profile", ".ssh/", ".sudo_as_admin_successful", ".vscode/", ".xsession-errors", ".xsession-errors.old", "Downloads/", and "snap/". A message at the bottom indicates "no changes added to commit (use "git add" and/or "git commit -a")". The user then runs "git add .gitignore" and "git commit -m ".gitignore"" which creates a new commit with the message ".gitignore" and adds the ".gitignore" file to the repository.

v) Haz una captura de pantalla que muestre el histórico de cambios del repositorio.



```
File Edit View Search Terminal Help
administrador@ubusrv:~$ git log --graph
* commit cdca4090545ae014b2f3d23e7c5448eb7d68ea98 (HEAD -> master)
| Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
| Date:   Sat Oct 11 11:03:58 2025 +0000
|
|       .gitignore
|
* commit 23182602f7a412a80e9cd6c9a99af4a0172f8de
| Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
| Date:   Sat Oct 11 10:31:00 2025 +0000
|
|       peliculas
|
* commit 37d9bf7fd31424df1497cea1b2c53647e8bfd544
| Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
| Date:   Sat Oct 11 10:15:09 2025 +0000
|
|       peliculas y comidas
|
* commit 7dcc1bb6a9f6f64eec72cfe27aa498e8a028a0d0
| Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
| Date:   Sat Oct 11 09:54:52 2025 +0000
|
|       Segunda version
|
* commit 961d022762077c081098b612709140067f36b97d
| Author: Victor Palanca <vicpalpes@alu.edu.gva.es>
| Date:   Sat Oct 11 09:46:23 2025 +0000
|
|       Primera version
administrador@ubusrv:~$
```