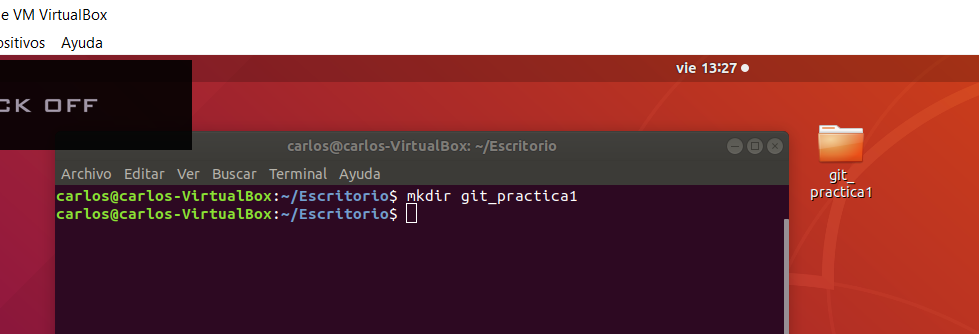
# **Apartado 1: Conociendo los comandos básicos de git.**

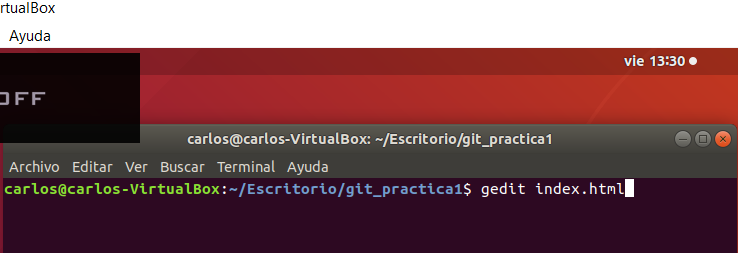
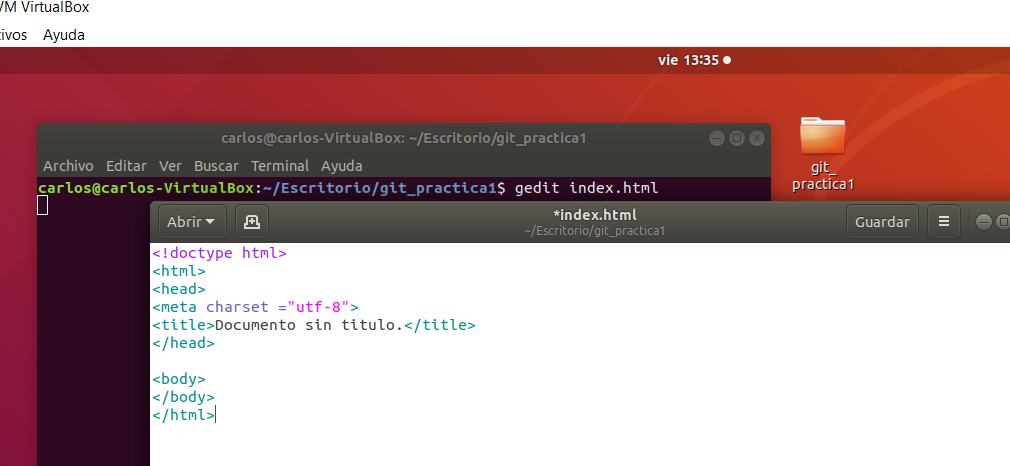
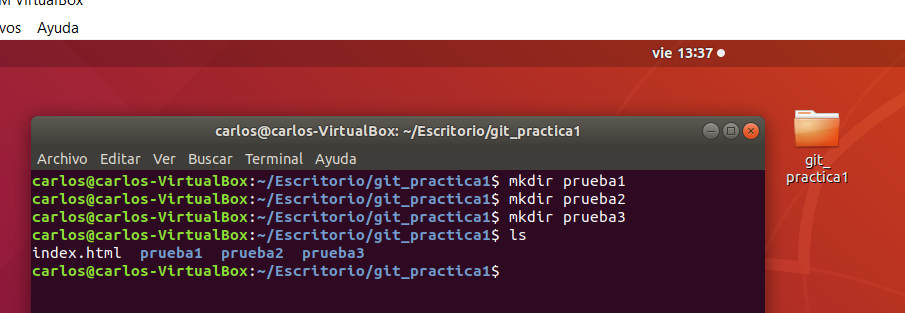
1. ¿Para qué sirven estos comandos de git?
   * git init: Es un comando que se utiliza una sola vez durante la configuración inicial de un repositorio nuevo. Al ejecutar este comando, se creará un nuevo subdirectorio. git en tu directorio de trabajo actual. También se creará una nueva rama maestra.
   * git add: Puede recibir tanto una ruta de archivo como de un directorio; si es de un directorio, el comando añade recursivamente los archivos que están dentro de él.
   * git commit: El comando git commit captura una instantánea de los cambios preparados en ese momento del proyecto. Las instantáneas confirmadas pueden considerarse como versiones "seguras" de un proyecto: Git no las cambiará nunca a no ser que se lo pidas expresamente.
   * git commit -m: Hace commit a los archivos que indiquemos, de esta manera quedan guardados nuestras modificaciones.
   * git status: Nos indica el estado del repositorio, por ejemplo, cuales están modificados, cuales no están siendo seguidos por GIT, entre otras características.
2. ¿Dónde puedo consultar la función de todos los comandos de git?

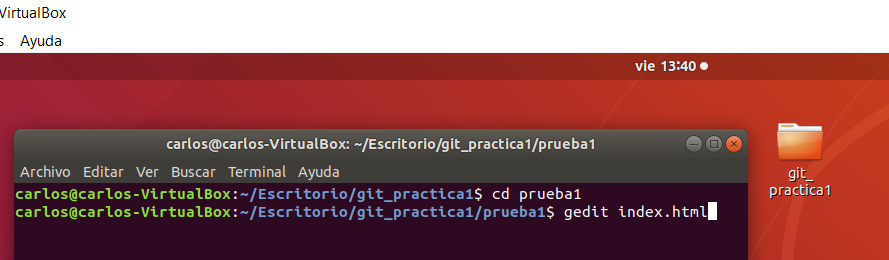
En Google buscamos la página oficial de git. Dentro de git nos iremos a documentación, luego reference manual y después en complete list f all comands, y ahí te aparecen todos los comandos y que es lo que hacen.

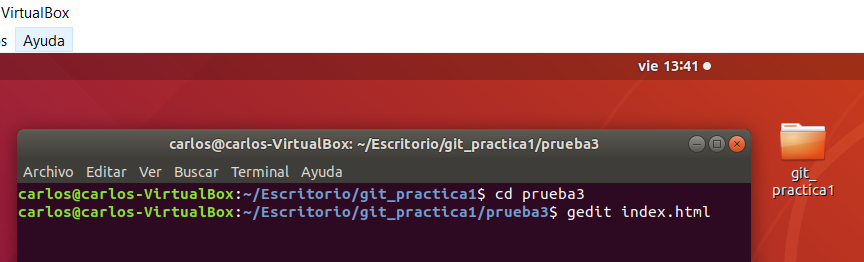
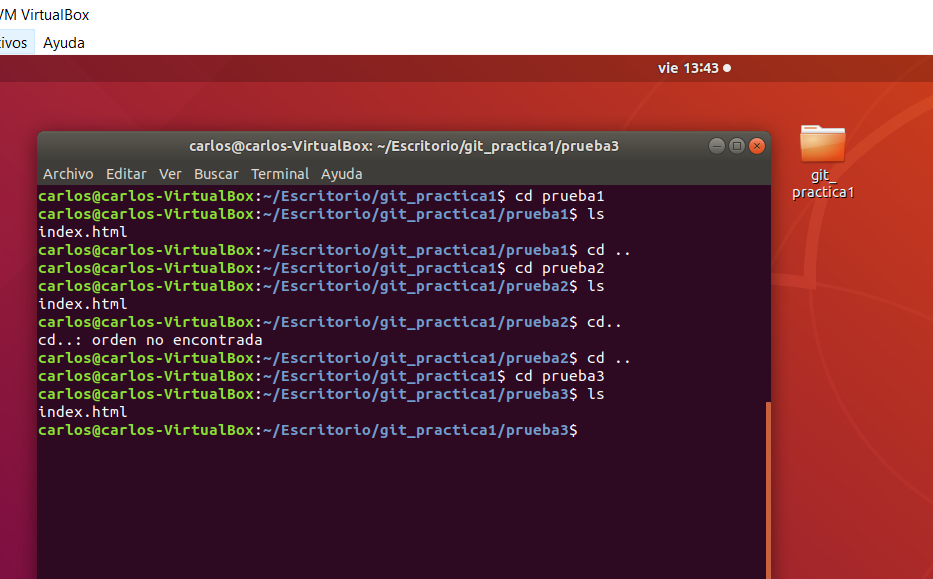
# **Apartado 2: poniendo en práctica los comandos básicos**

1. Crea un directorio en tu ordenador “git\_practica1”

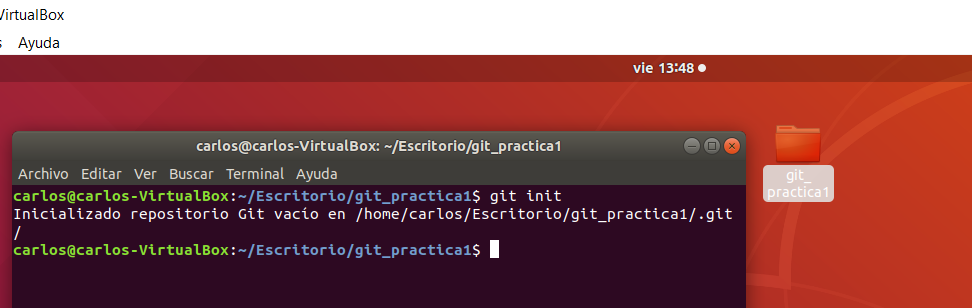


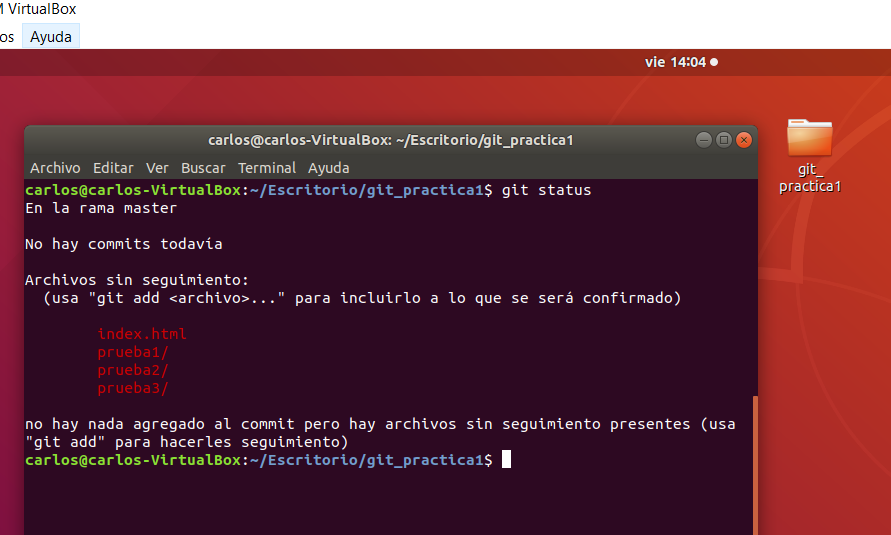
1. Añade un archivo “index.html” y escribe en él algún código (por ejemplo, un encabezado)
2. Crea varios directorios más (dentro de “git\_practica1” y guarda en ellos varios archivos html más (no hace falta que tengan código).



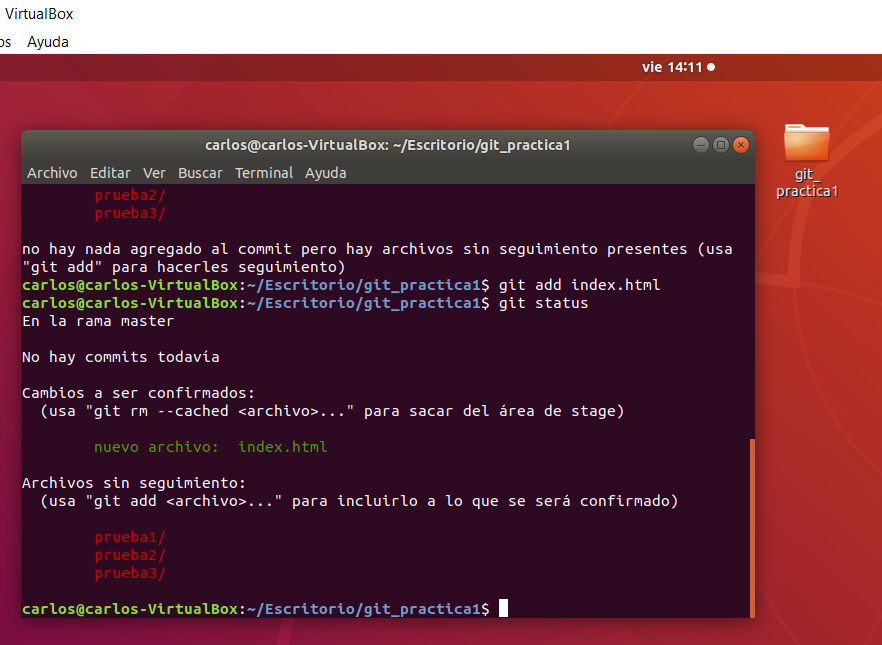


1. Inicia git en este directorio

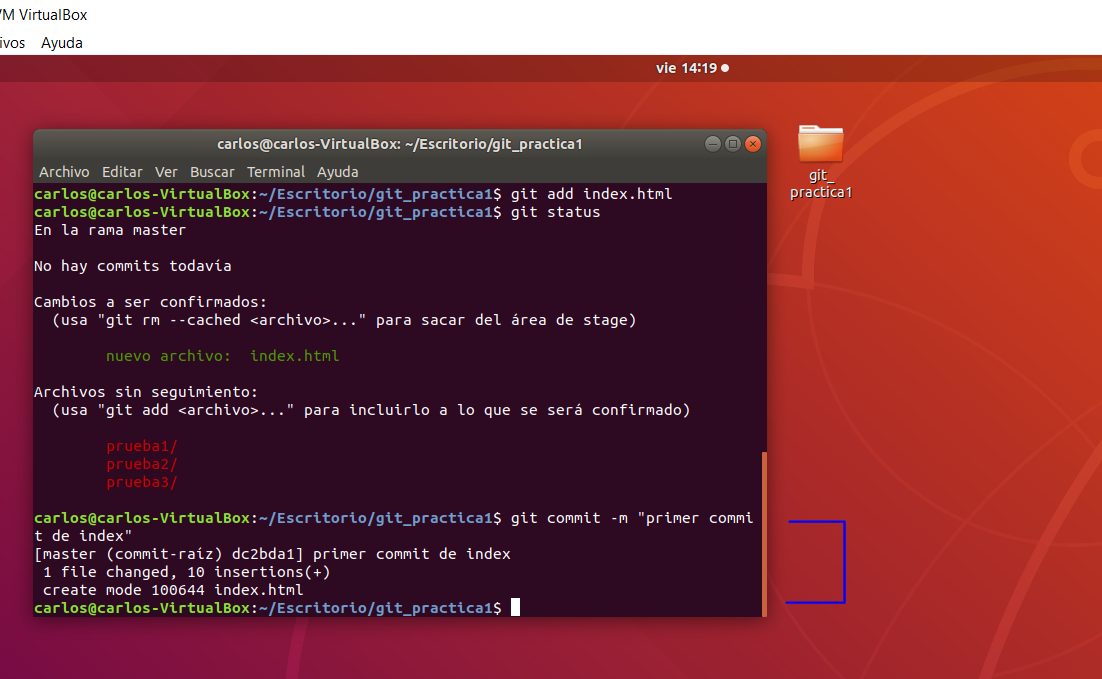
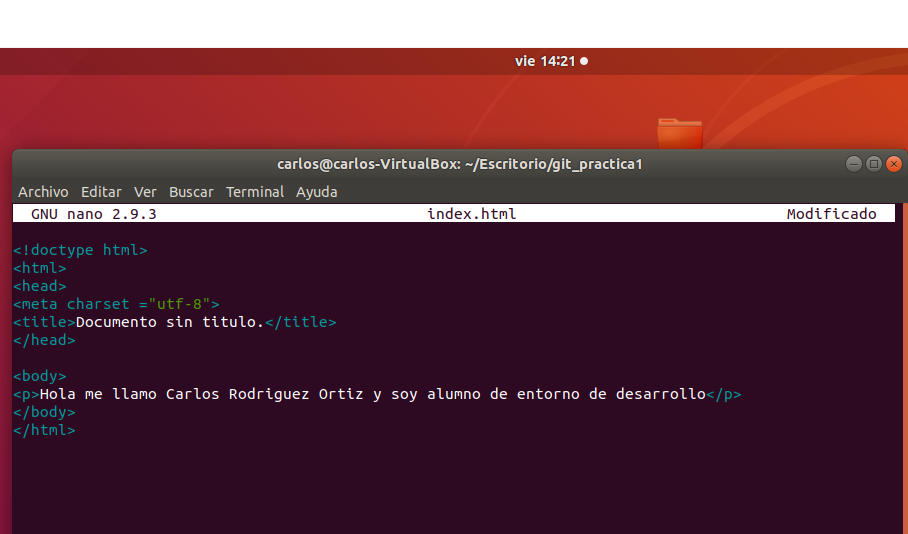


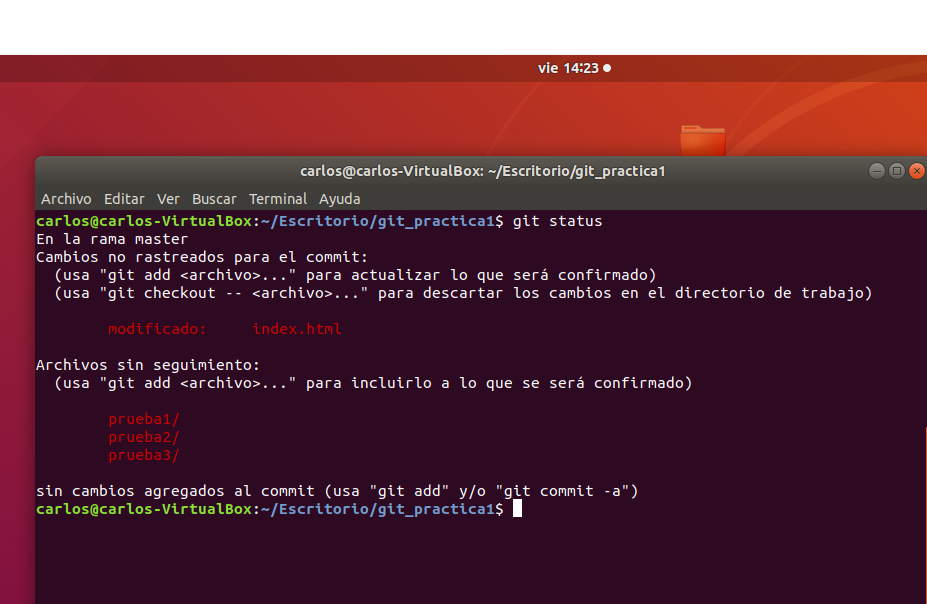
1. Comprueba el estado de todos los archivos y directorios del proyecto git. Explica en qué área de git está cada archivo

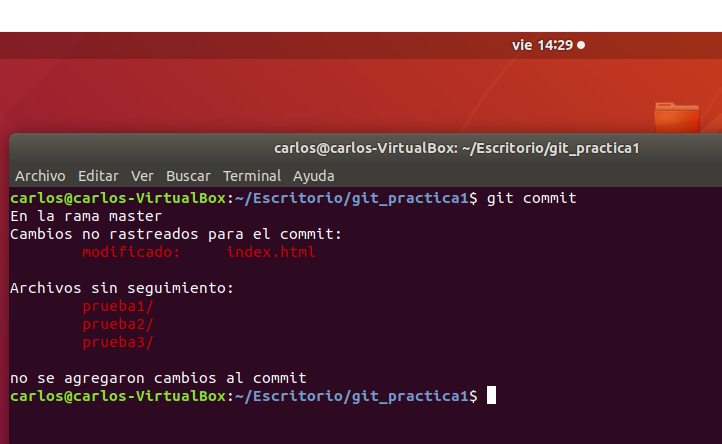
Están todos en el repositorio local ahora mismo, ya que no hemos añadido ningún archivo al área de ensayo. No se le está realizando ningún seguimiento a los archivos.

1. Añade el archivo index a git y vuelve a comprobar el estado de los archivos, ¿qué diferencia hay con la anterior comprobación? Explícalo

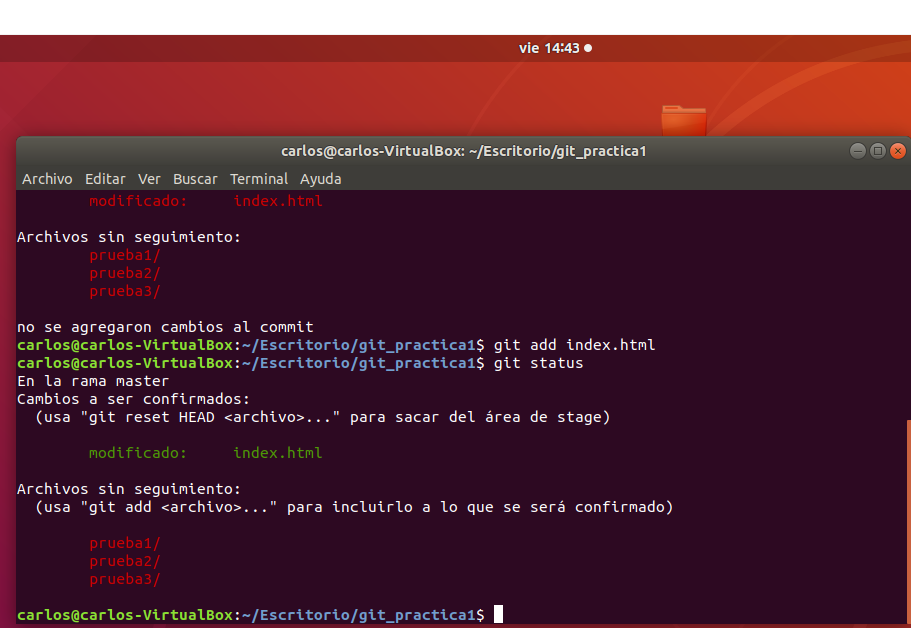
La diferencia que hay con el paso anterior es que ahora hemos pasado el archivo index.html del repositorio local a el área de ensayo, y le hace un seguimiento.

1. Confirma a git que el código del archivo index está ok (de momento). Para ello, indica esto en el mensaje (por ejemplo “primer commit de index”)
2. Añade más código a tu archivo index, por ejemplo, varios párrafos con texto y vuelve a comprobar el estado de los archivos, ¿qué diferencia hay con la anterior comprobación? Explícalo

La diferencia es que git detecta que el archivo index.html ha sido modificado en cuanto a la última versión que tiene el subido a el área de ensayo, por lo cuál nos lo notifica para que lo volvamos a subir al área de ensayo con las modificaciones pertinentes.

1. Intenta hacer un commit (no creo que puedas), explica qué pasa y soluciona el error para poder hacerlo.

No se puede hacer un commit ya que el archivo que esta en el área de ensayo no esta modificado como el del repositorio local. Lo tenemos que subir primero para después poder hacer el commit



Ya estaría listo para hacer el git commit y el gitpush

1. Comprueba cuántas copias de index tienes en tu repositorio
2. Restaura tu proyecto con el primer index que codificaste