GIT LOCAL

Es un software de [control de versiones](https://es.wikipedia.org/wiki/Control_de_versiones) diseñado con el propósito de llevar un registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos.

Cada vez que confirmas un cambio, o guardas el estado de tu proyecto en Git, él básicamente toma una foto del aspecto de todos tus archivos en ese momento, y guarda una referencia a esa copia instantánea. Git maneja sus datos como una secuencia de copias instantáneas.

### Casi todas las operaciones son locales

### Tienes toda la historia del proyecto ahí mismo, en tu disco local, la mayoría de las operaciones parecen prácticamente inmediatas.

### para navegar por la historia del proyecto, Git lee directamente de tu base de datos local, NO NECESITA CONECTARSE A UN SERVIDOR.

* Si quieres ver los cambios introducidos en un archivo entre la versión actual y la de hace un mes, Git puede buscar el archivo hace un mes y hacer un cálculo de diferencias localmente, en lugar de tener que pedirle a un servidor remoto que lo haga u obtener una versión antigua desde la red y hacerlo de manera local.

. Puedes trabajar tranquilamente y luego subir los cambios.

Poner como hacer git pull

### Git tiene integridad

Todo en Git es verificado mediante una suma de comprobación (checksum en inglés) antes de ser almacenado, y es identificado a partir de ese momento mediante dicha suma.

No puedes perder información durante su transmisión o sufrir corrupción de archivos sin

que Git sea capaz de detectarlo

### Git generalmente solo añade información

Es muy difícil conseguir que el sistema haga algo que no se pueda enmendar, o que de algún modo borre información. Después de confirmar una copia instantánea en Git es muy difícil de perderla, especialmente si envías tu base de datos a otro repositorio con regularidad.

Poner git commit que es lo que pasa, captura de pantalla

puedes recuperar datos aparentemente perdidos.

### Los Tres Estados

CONFIRMADO (committed) Confirmado significa que los datos están almacenados de manera segura en tu base de datos local.

MODIFICADO (modified) Modificado significa que has modificado el archivo pero todavía no lo has confirmado a tu base de datos.

PREPARADO (staged). Preparado significa que has marcado un archivo modificado en su versión actual para que vaya en tu próxima confirmación.



El DIRECTORIO DE GIT O REPOSITORIO

# 2.1 Fundamentos de Git - Obteniendo un repositorio Git

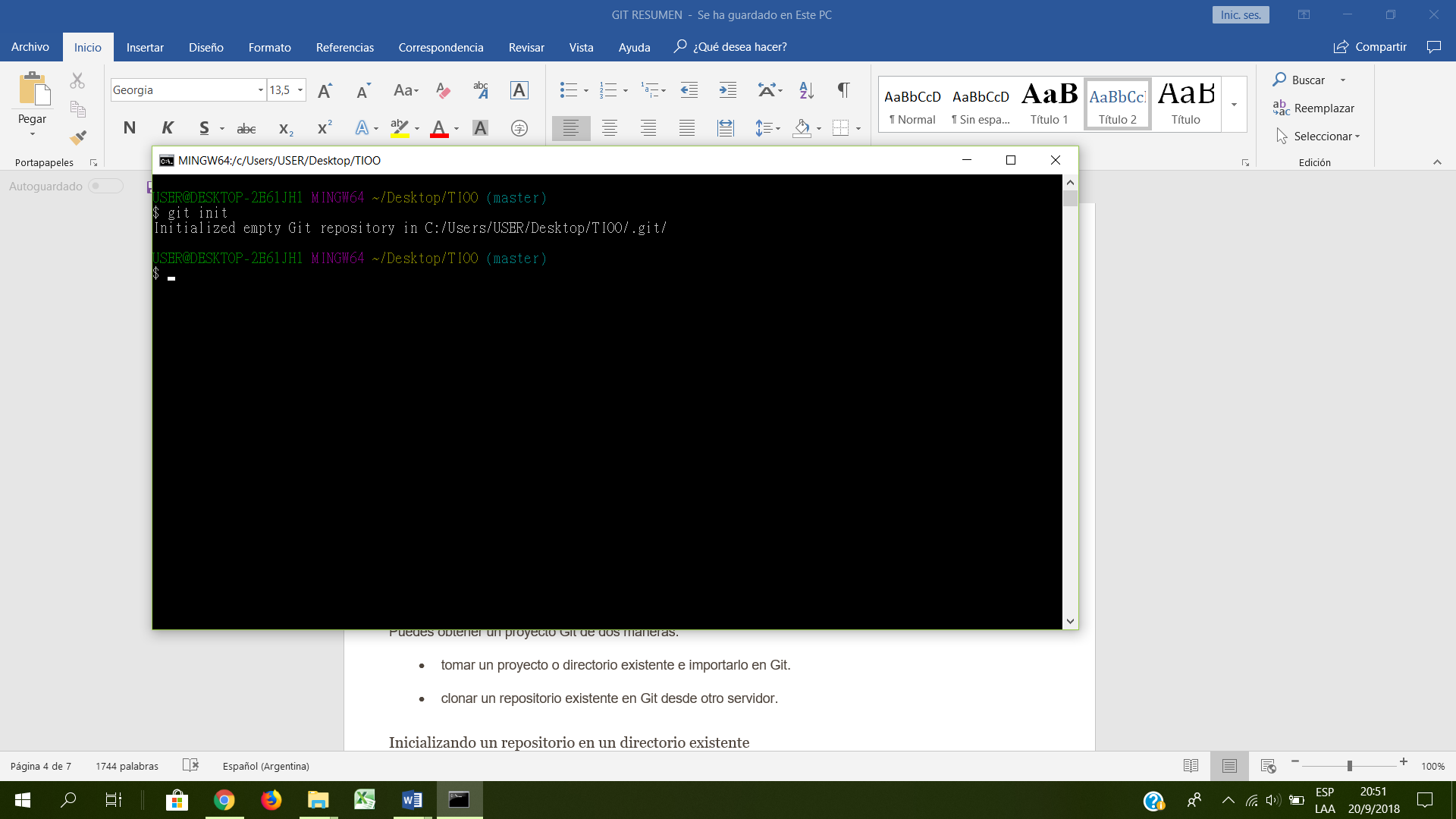
## Obteniendo un repositorio Git

Puedes obtener un proyecto Git de dos maneras.

* tomar un proyecto o directorio existente e importarlo en Git.
* clonar un repositorio existente en Git desde otro servidor.

### Inicializando un repositorio en un directorio existente

Si estás empezando a seguir un proyecto existente en Git, debes ir al directorio del proyecto y usar el siguiente comando:



$ git init

Esto crea un subdirectorio nuevo llamado .git, el cual contiene todos los archivos necesarios del repositorio – un esqueleto de un repositorio de Git. Todavía no hay nada en tu proyecto que esté bajo seguimiento.

Cuando ponemos git init aparece

$ ls -F1

HEAD (guarda el contenido de tu base de datos)

config\* (opciones de configuración específicas de este proyecto)

description

hooks/ (contiene tus scripts, tanto de la parte cliente como de la parte servidor,)

info/ (guarda un archivo global de exclusión con los patrones a ignorar además de los presentes en el archivo)

objects/

refs/ (guarda los apuntadores a las confirmaciones de cambios )

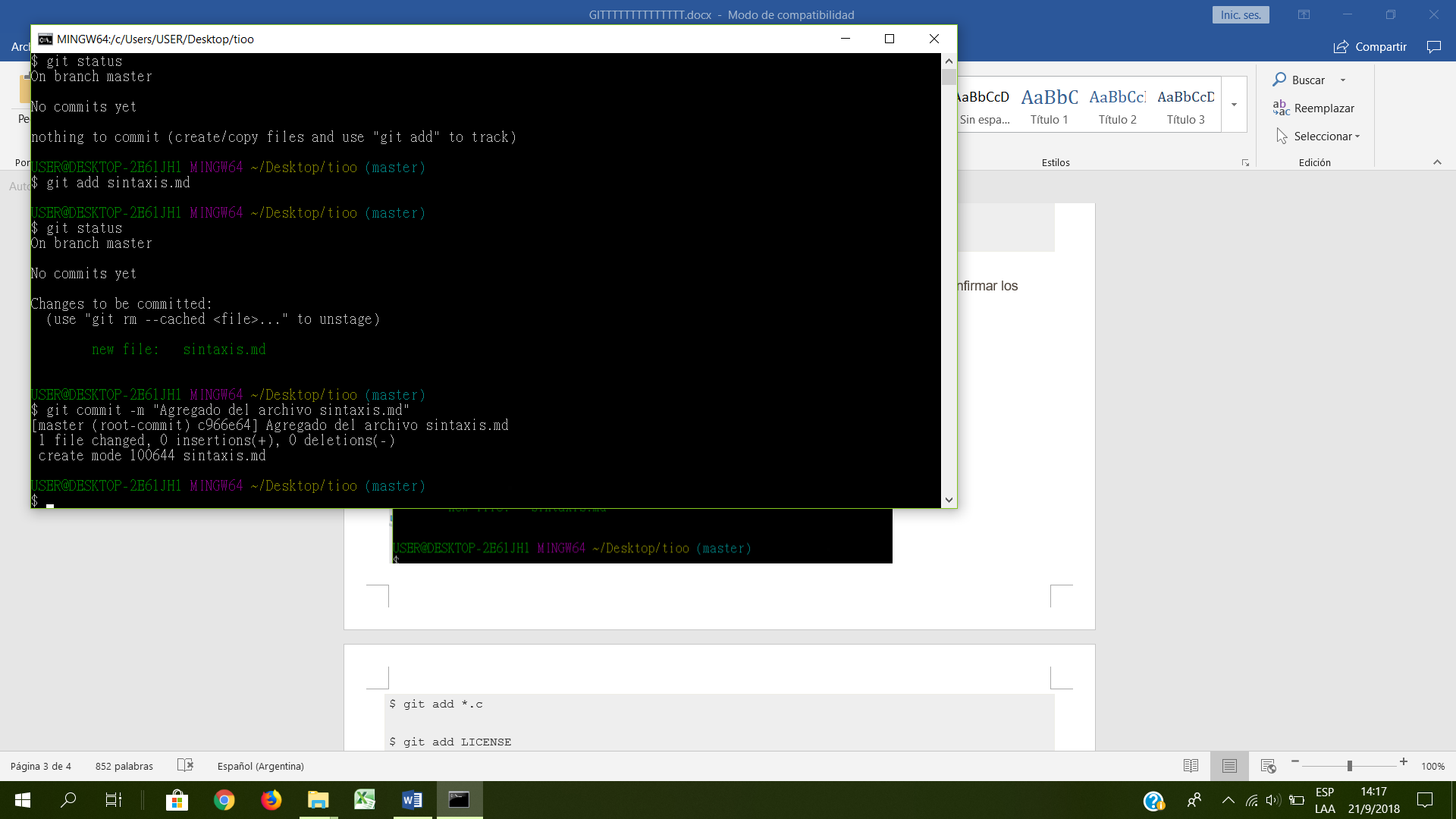
git add para especificar qué archivos quieres controlar, seguidos de un git commit para confirmar los cambios:

al agregar el archivo sintaxis.md, se convierte en verde new file: sintaxis.md

En este momento, tienes un repositorio de Git con archivos bajo seguimiento y una confirmación inicial.

Al confirmar los cambios puedes agregar mensaje, de la cual es una breve descripción cuáles fueron los cambios hechos, para que otros colaboradores puedan saber ¿qué hiciste? Y ¿para qué?

$ git commit -m “agregado del archivo sintaxis.md”

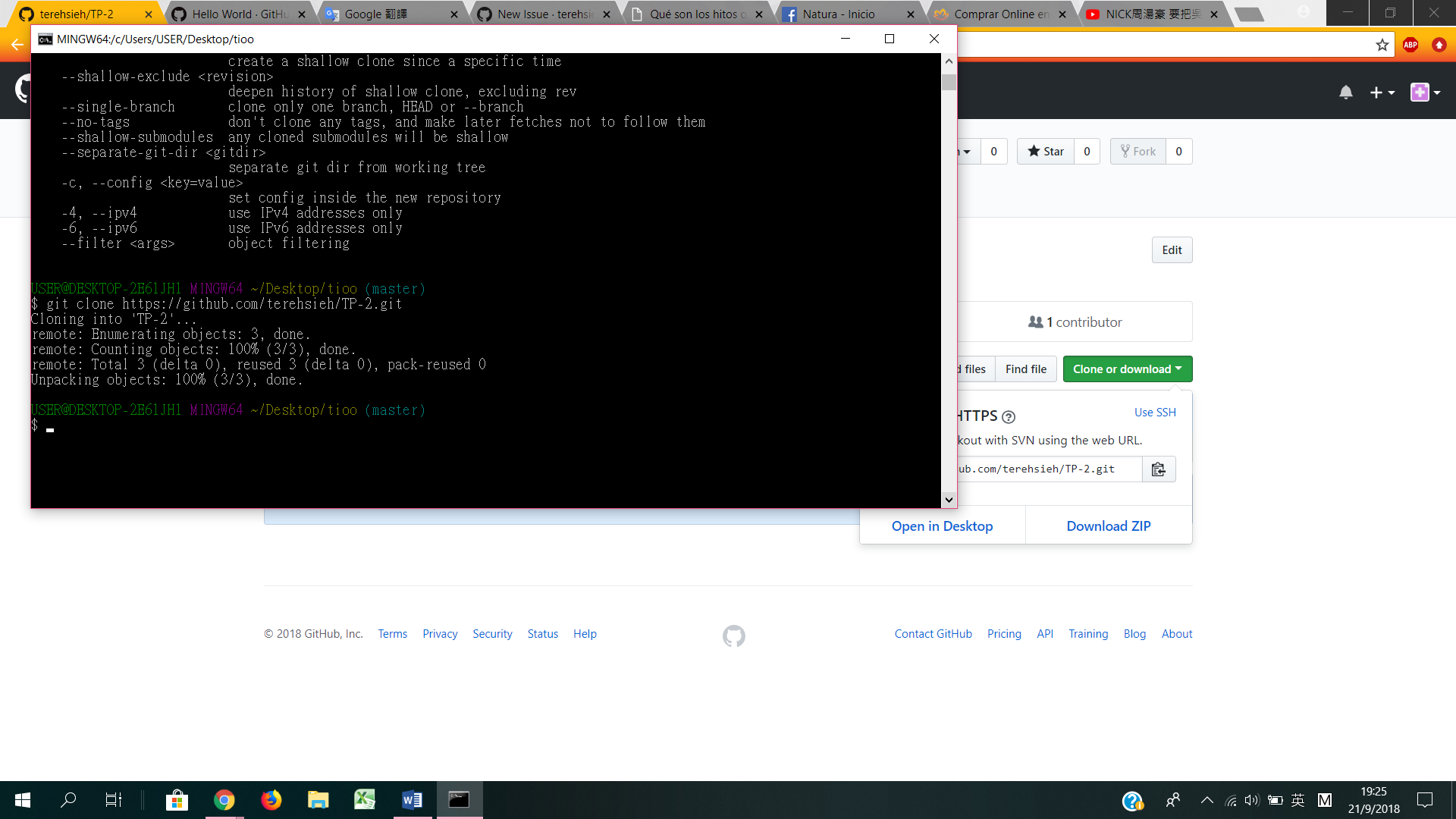


### Clonando un repositorio existente

Si deseas obtener una copia de un repositorio Git existente — por ejemplo, un proyecto en el que te gustaría contribuir — el comando que necesitas es git clone.

git clone [url]

ejemplo: $ git clone https://github.com/terehsieh/TP-2.git



# 2.2 Fundamentos de Git - Guardando cambios en el Repositorio

## Guardando cambios en el Repositorio

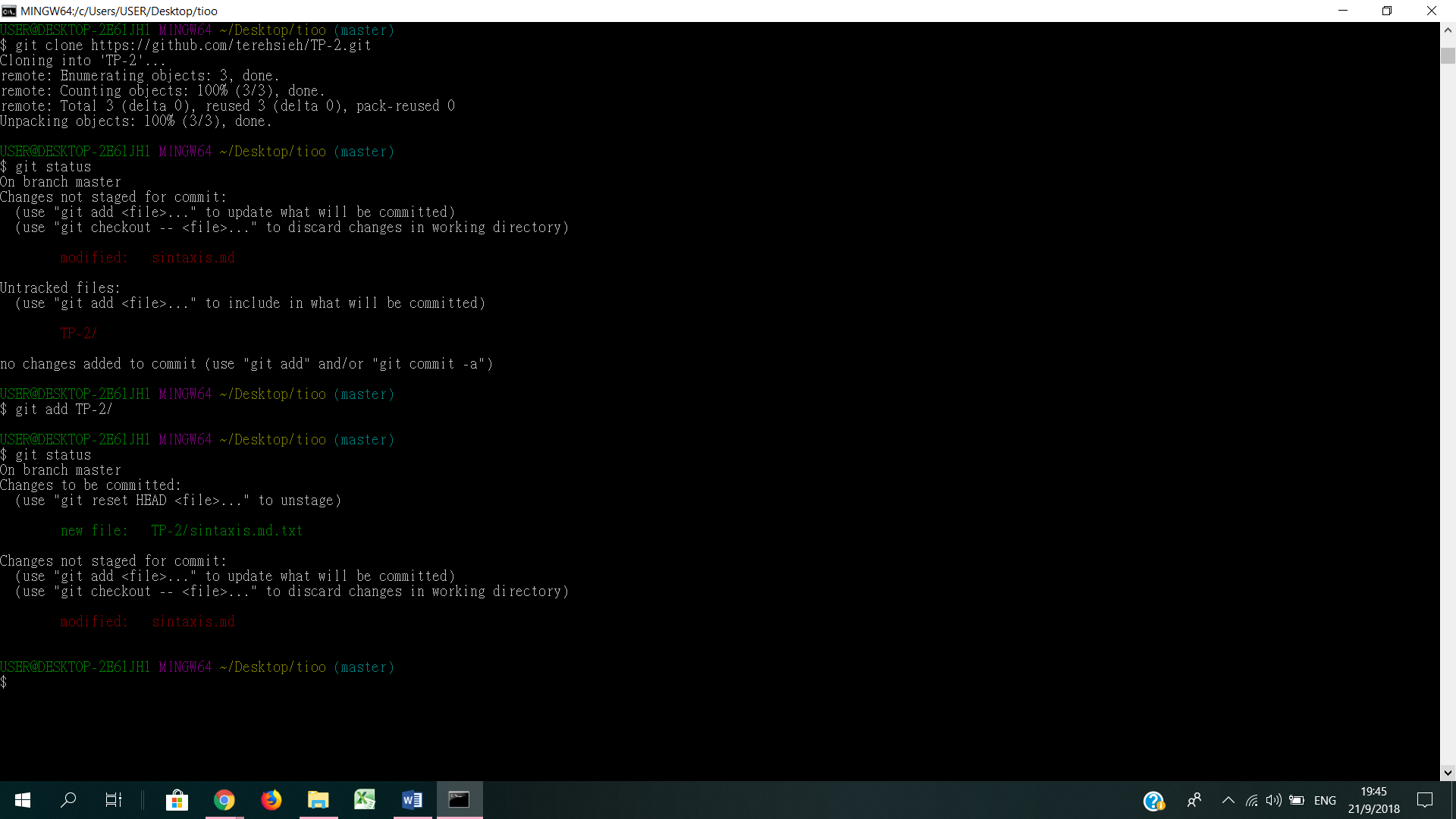
cada archivo de tu repositorio puede tener 2 estados:

* rastreados: TRACKED FILES todos aquellos archivos que estaban en la última instantánea del proyecto; pueden ser archivos sin modificar, modificados o preparados
* sin rastrear: cualquier otro archivo en tu directorio de trabajo que no estaba en tu última instantánea y que no están en el área de preparación (*staging area*).

Cuando CLONAS por primera vez un repositorio TODOS ESTAS RASTREADOS

GIT STATUS: para saber en que estados están los archivos

Si agrego un archivo aparece -----“Untracked files” (“Archivos no rastreados” en inglés), son archivos que no tenias en el commit anterior.



### Rastrear Archivos Nuevo

Para agregarlos tengo q poner git add NOMBRE DEL ARCHIVO

 El comando 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.

### Preparar Archivos Modificados

git add. git add es un comando que cumple varios propósitos - lo usas para empezar a RASTREAR archivos nuevos, PREPARAR archivos, y hacer otras cosas como MARCAR COMO RESUELTO ARCHIVOS en conflicto por combinación.