## Objetivo del ejercicio

El objetivo del presente ejercicio es que comencemos a practicar algunas funcionalidades de Git para adquirir soltura. A partir de la próxima sesión pasaremos a trabajar con GitHub

## Pasos a seguir

Para abordar el ejercicio hemos de seguir los siguientes pasos:

### Descarga el fichero que encontrarás en mudle junto con este ejercicio (Owin\_Covid\_Data.xlsx).

### Crea un directorio en tu escritorio llamado ‘Ejercicio\_git’ y almacena ahí el fichero descargado.

### Abre el terminal y sitúate en el directorio ‘Ejercicio\_git’ en el que vamos a trabajar.

### Inicializa Git en ese directorio.

### Configura como parámetros globales de Git tu nombre y tu dirección de email.

### Comprueba con el comando ‘git status’ el estado del fichero del directorio.

### Crea en el directorio un fichero llamado README.md

### Genera un fichero .gitignore y especifica que no se guarden ficheros con extensión .xlsx

### Genera un código R donde haces una importación del fichero y guárdalo en el fichero donde estamos trabajando

### Añade en .gitignore que tampoco quieres considerar ficheros .Rhistory

### Añade los ficheros al stage o escenario.

### Crea un commit con el código R, llamado “Añado el código de carga de datos”.

### Entra en el fichero README.md y modifícalo, añadiendo el siguiente texto:

# Descripción del proyecto

Utilizaremos este proyecto para familiarizarnos con Git.

### Crea un commit con este fichero, llamado “Añadimos el fichero README.md”.

### Añade en el código de R la generación de un fichero con datos agregados de contagios por país y exporta ese fichero a la ruta donde estamos trabajando.

### Con el comando “git status” comprueba cómo Git reconoce que los ficheros README.md y .R ha sido modificados.

### En un único paso, añade al stage y crea un commit con las modificaciones. Recuerda que esto lo podríamos hacer con el comando “git commit –am …”. El título del commit será “Modificamos el fichero README.md y generamos datos agregados”.

### Utiliza el comando “git log” para ver todos los commits que hemos creado.

### Utilizar el comando “git diff” para analizar las diferencias entre los dos últimos commits que has realizado.