

PRÁCTICA_11 : GIT

1. Trabajaremos con git desde consola

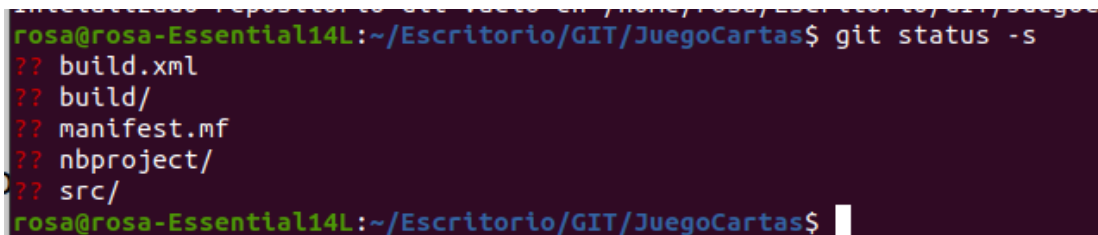
a) Creamos el repositorio

- Creamos una carpeta en nuestra máquina llamada GIT
- Hacemos un nuevo proyecto java que escriba por consola “Esto es una prueba GIT”
- Abre un terminal y ve a la carpeta GIT
- Ejecuta `git init`
- Comprueba que ha creado la carpeta para el repositorio (estará como archivos ocultos `ls -a`)

b) Indicamos que archivos seguiremos

Comprueba el estado de “respaldo” del contenido de la carpeta GIT

Listamos los archivos que hay en la carpeta : `git status -s` . Las interrogaciones a la izquierda nos indican que no se está haciendo seguimiento



```
rosa@rosa-Essential14L:~/Escritorio/GIT/JuegoCartas$ git status -s
?? build.xml
?? build/
?? manifest.mf
?? nbproject/
?? src/
rosa@rosa-Essential14L:~/Escritorio/GIT/JuegoCartas$
```

Probamos a añadir al repositorio el proyecto java y comprobamos el estado de nuevo

Ya no aparecen las ?? delante del archivo, está en el área de ensayo

c) Añadimos todo el proyecto al staging área y comprobamos de nuevo. ¿Ha habido algún cambio?

d) Movemos al repositorio git

`git commit -m "descripción de la instantánea"`

```
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$ git commit -m "inicio del ejemplo"
[master (commit-raiz) ec299dc] inicio del ejemplo
11 files changed, 2027 insertions(+)
create mode 100644 build.xml
create mode 100644 build/built-jar.properties
create mode 100644 build/classes/ejemploGIT_1/EjemploGIT_1.class
create mode 100644 manifest.mf
create mode 100644 nbproject/build-impl.xml
create mode 100644 nbproject/genfiles.properties
create mode 100644 nbproject/private/config.properties
create mode 100644 nbproject/private/private.properties
create mode 100644 nbproject/project.properties
create mode 100644 nbproject/project.xml
create mode 100644 src/ejemploGIT_1/EjemploGIT_1.java
```

Si utilizas la consola git por primera vez es probable que te pida username y email

- Muestra el estado del proyecto y observa qué ha pasado
- Comprueba que se ha hecho la primera instantánea

d) Edita el proyecto, añade una variable string con tu nombre y escríbelo por pantalla. Muestra el estado del proyecto

Esa M delante del archivo indica que se ha modificado y no se ha “respaldado”

e) Haz un nuevo respaldo con la descripción “añadido nombre”. Antes de hacer el commit debes añadirlo de nuevo al área de ensayo

```
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$ git add .
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$ git commit -m "añadido nombre"
[master 0b5c0c8] añadido nombre
3 files changed, 6 insertions(+), 2 deletions(-)
rewrite build/classes/ejemploGIT_1/EjemploGIT_1.class (88%)
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$
```

f) Para listar todas las copias o instantáneas `git log --oneline`

```
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$ git log --oneline
0b5c0c8 (HEAD -> master) añadido nombre
ec299dc inicio del ejemplo
rosa@rosa-Essential14L:~/Escritorio/GIT/EjemploGIT_1$
```

g) Para restaurar a la primera instantánea y desechar definitivamente el cambio `git reset --hard` código de la instantánea. Vuelve a la primera y comprueba que el archivo ha sido restaurado, tu proyecto está como al inicio.

h) Lista todas las instantáneas, observa qué ha pasado

i) Añade algo de código a tu proyecto, haz un commit con este cambio. Vuelve a añadir más código y haz de nuevo un commit. Para la siguiente práctica debemos tener al menos tres commit.

j) Elimina el archivo java de tu repositorio. Ve a un commit anterior, vuelve al último commit y comprueba que has recuperado el archivo con el último commit que hayas hecho, sin eliminar ningún otro.