

Como empezar a usar Git

Crea tu primer repositorio Git

1 - Inicializa un repositorio vacío

Para poder usar Git lo primero que tienes que hacer es abrir la terminal y ubicarte en el directorio donde están ubicadas las fuentes de tu proyecto y crear un repositorio vacío:

```
cd <directorio-proyecto>  
git init
```

Acabas de crear el repositorio Git local correspondiente a tu proyecto.

2 - Añade archivos al repositorio

Una vez tenemos el repositorio vacío tenemos que añadir todos los archivos y directorios del proyecto al repositorio, es muy fácil:

```
git add .
```

3 - Haz el primer commit

El comando anterior solo añade archivos al repositorio, no hace commit de los cambios realizados. Por tanto, debes hacer el primer commit:

```
git commit -m "Mi primer commit"
```

4 - Configura el repositorio remoto

Para colaborar con otros programadores no es suficiente con tener un repositorio Git local, necesitas un repositorio ubicado en algún servidor al cual todos los desarrolladores puedan acceder. Esto es precisamente lo que te ofrece la **Forja**.

Lo primero es configurar el repositorio que has creado en la Forja como *origin* o repositorio remoto de tu repositorio local:

```
git remote add origin git@forja.guebs.com:nueva-web-empresa-sl.git
```

IMPORTANTE: El comando superior no es más que un ejemplo en el que suponemos que tienes un proyecto llamado **nueva-web-empresa-sl** en la Forja.

5 - Tu primer push

Una vez configurado el repositorio remoto ya puedes hacer el primer *push* de cambios:

```
git push origin master
```

Como continuar trabajando con Git

Ahora ya tienes un repositorio Git local, has hecho tu primer commit y también has hecho tu primer push al repositorio alojado en la Forja. ¿Y ahora que?

Ahora debes continuar con el desarrollo de tu proyecto y de vez en cuando hacer commit al repositorio local, de esta forma tendrás un historico de cambios y podrás revertir cualquier cambio que hagas.

Hacerlo es muy fácil, es casi igual que la primera vez, solo que ahora debes pasar la opción **-a** al hacer commit:

```
git add .  
git commit -a -m "Descripcion del commit"
```

Por otro lado, cada vez que completes alguna característica o corrijas algún bug conviene que hagas *push* al repositorio remoto, para que el resto de programadores puedan obtener el cambio. El comando a ejecutar es igual que la primera vez:

```
git push origin master
```