



Comandos básicos

Versión de git ----- `git version`

`git config --global user.name "mi nombre"`

Es recomendable utilizar el correo asociado a Github

`git config --global user.email
"myemail@example.com"`

Ayuda

`// Ayuda sobre los comandos
git help`

Mi primer repositorio

`// Iniciar un nuevo repositorio
// Crear la carpeta oculta .git
git init`

// Ver que archivos no han sido registrados
git status

// Agregar todos los archivos para que esté pendiente de los cambios
git add .

// Crear commit (fotografía del proyecto en ese momento)
git commit -m "primer commit"

// Muestra la lista de commit del mas reciente al más antiguo
git log

El comando status verificará que archivos han sido modificados. Cuando deseemos registrar esos cambios tendremos que agregarlos con add . así ya estará listo para poder hacer un commit.

El commit realiza la copia de ese instante para poder volver en el tiempo si es que es necesario.

// Muestra en una línea los commit realizados
git log --oneline

// Muestra en una línea los commit realizados pero más elegante
git log --oneline --decorate --all --graph

// Solo muestra los archivos modificados
git status -s

Movernos por los “commits”

// commit en específico f52f3da
git reset --mixed f52f3da

// commit en específico f52f3da y eliminamos los cambios futuros
git reset --hard f52f3da

// Muestra todos los cambios incluso si borramos los commit
git reflog

- Activar el coloreado de la salida `git config --global color.ui auto`
- Mostrar la configuración `git config --list`

Para no hacer seguimiento de carpetas o archivos, debemos crear el siguiente archivo:

Ignorando Archivos

`.gitignore`

Su estructura de ejemplo sería así:

`archivo.js // Ignora el archivo en cuestion`

`*.js // Ignora todos los archivos con extensión .js`

`node_modules/ // Ignora toda la carpeta`

Ramas o branch

```
// Crea una nueva rama  
git branch nombreRama
```

```
// Nos muestra en que rama estamos  
git Branch
```

```
// Nos movemos a la nueva rama  
git checkout nombreRama
```

Podemos unir la rama master con la nueva, para eso tenemos que estar en la master para ejecutar el siguiente comando:

```
// Nos movemos a la nueva rama  
git merge nombreRama
```

```
// Eliminar una rama  
git branch -d nombreRama
```

Tags

Con los tags podemos hacer versiones de nuestro proyecto.

```
// Crear un tags
git tag versionAlpha -m "versión alpha"

// Listar tags
git tag

// Borrar tags
git tag -d nombreTags

// Hacer una versión en un commit anterior ej: f52f3da
git tag -a nombreTag f52f3da -m "version alpha"

// Mostrar información del tag
git show nombreTag
```

GITHUB

Para subir nuestro proyecto debemos crear un nuevo repositorio, al momento de la creación nos mostrará una serie de comandos para subir el proyecto.

```
git remote add origin https://github.com/igformacion/github.git  
git push -u origin master
```

Al ejecutar estas líneas de comando te pedirá el usuario y contraseña de tu cuenta de github.

// Nos muestra en que repositorio estamos enlazados remotamente.
git remote -v

Subir los tags

Por defecto si creaste un proyecto con diferentes versiones no subirá los tags, para eso tenemos el siguiente comando.

```
git push --tags
```

Push

Al ejecutar el comando git push estaremos subiendo todos los cambios locales al servidor remoto de github, ten en cuenta que tienes que estar enlazado con tu repositorio, para eso puedes utilizar git remote -v luego ejecuta:

```
git push
```

Pull

Cuando realizamos cambios directamente en github pero no de forma local, es esencial realizar un pull, donde descargaremos los cambios realizados para seguir trabajando normalmente.

Es importante estar enlazados remotamente, puedes verificar con: `git remote -v`, luego ejecuta:

```
git pull
```

Fetch

Este comando hace la comparación de nuestros archivos locales con los del servidor, si existiera alguna diferencia nos pediría realizar un get pull para realizar un match de nuestros archivos locales.

```
git fetch
```

Clonar repositorio

Para descargar un repositorio completo basta con tomar la url ej:
`https://github.com/igformacion/github.git`

Y ejecutar el siguiente comando en alguna carpeta del ordenador

```
git clone https://github.com/igformacion/github.git nombreCarpeta
```