



2022

Contenido

¿Qué es Git?



Instalación



Comandos - Locales



Comandos - Remotos



GitHub



Recursos



■ Qué es Git?

What is Git?

Git es controlador de versiones, diseñado para manejar desde proyectos pequeños hasta proyectos más grandes, con rapidez y eficiencia.



Contenido

¿Qué es Git? ☐

Instalación ☐

Comandos - Locales ☐

Comandos - Remotos ☐

GitHub ☐

Recursos ☐

Instalacion

Instalación.

- **Homebrew**

`$ brew install git`

** Si no tienes homebrew, puedes instalarlo desde el siguiente [link](#);

- **MacPorts**

`$ sudo port install git`

** Si no tienes MacPorts, puedes instalarlo desde el siguiente [link](#).

Windows

- **Instalador**

- <https://git-scm.com/download/win>

Contenido

¿Qué es Git? ☐

Instalación ☐

Comandos - Locales ☒

Comandos - Remotos ☐

GitHub ☐

Recursos ☐

Comandos – Locales

Primeros comandos.

- **git --version**

Muestra la version instalada de git.

- **git help**

Muestra la información/descripción de los comandos.

- **git help commit**

- **git config**

Sirve para agregar configuraciones a la cuenta de git.

- **git config --global user.name "user name"**

Configura el nombre de usuario.

- **git config --global user.email "user.name@email.com"**

Configura el correo electrónico

- **git config --global init.defaultBranch main**

Configuración para que el nombre de la rama principal sea "main" en lugar de "master".

Comandos fundamentales

- **git init**

Crea un repositorio de git vacío o reinicializa uno ya existente.

- **git status**

Muestra el estatus del ambiente de trabajo. (working tree)

- **git add**

Agrega uno o más archivos al área de preparación.(staging area)

- `git add nombre_archivo.ext`
- `git add *.ext`
- `git add carpeta/*.ext`
- `git add carpeta/`
- `git add carpeta/.gitkeep`

- **git reset nombre_archivo.ext**

Remueve los archivos indicados del área de preparación al área de trabajo.

Comandos fundamentales

- **git commit -m "initial commit"**

Registra los cambios del repositorio; crea una captura del proyecto en un momento dado.

- **git commit -am "new message"**

Tiene la funcionalidad de un "git add." y un "git commit" combinadas. (solo para archivos con seguimiento)

- **git commit --amend -m "message correction"**

Corrige el mensaje del último commit

- **git commit --amend -C HEAD**

Agrega los cambios en el área de preparación al último commit en la línea del tiempo.

- **git log**

Muestra el historial de commits que se han realizado en el branch.

Contenido

¿Qué es Git? ☐

Instalación ☐

Comandos - Locales ☐

Comandos - Remotos ☒

GitHub ☐

Recursos ☐

Comandos – Remotos



Comandos remotos

- `git clone /path/repository/`
Clona un repositorio en un nuevo directorio.
- `git push origin <branch_name>`
Actuliza las referencias remotas junto con los objetos asociados
- `git pull origin <branch_name>`
Obtiene e integra las referencias y los cambios del repositorio o rama remota al local.
- `git fetch origin`
Descarga los cambios y las referencias de un repositorio o rama remota al local.

Contenido

¿Qué es Git? ☐

Instalación ☐

Comandos - Locales ☐

Comandos - Remotos ☐

GitHub ☒

Recursos ☒



Creación de una cuenta

- Requerimientos:
 - Correo electrónico
- PASOS A SEGUIR:
 1. Ir al sitio oficial de [GitHub](https://github.com)
 2. Ingresar tu correo electrónico.
 3. Ingresar una contraseña de al menos 15 caracteres u 8 incluyendo un número y una minúscula.
 4. Ingresar un nombre de usuario único
 5. Indicar si quieres recibir anuncios y actualización por email con una “Y” para sí y una “N” para no.
 6. Verificar tu cuenta.
 7. Dar clic en crear la cuenta.
 8. Ingresar el código que se te envió al correo que diste de alta.

Doble autenticación (2FA)

Dentro de tu cuenta de GitHub sigue los siguientes pasos para habilitar la doble autenticación:

1. Ir a la configuración de tu cuenta:
 - a. Dirígete a la imagen de tu usuario para desplegar un menú
 - b. Selecciona configuración (Settings)
2. Dentro de las configuraciones selecciona la opción de “Contraseña y autenticación” (Password and authentication)
3. En la sección de “Doble autenticación” (Two-Factor Authentication) dar clic en “Habilitar doble autenticación” (Enable two factor authentication)
4. Selecciona la opción que mejor te convenga:
 - a. Usando una aplicación
 - i. [1Password](#), [Authy](#), [LastPass Authenticator](#), or [Microsoft Authenticator](#).
 - b. Por mensaje de texto (SMS)
5. Una vez seleccionada una opción vas a tener que seguir la configuración que se te presenta
 - a. Para una aplicación tienes que escáner un QR
 - b. SMS: tienes que ingresar un número de teléfono válido
6. Guardar los códigos que se te muestran en la pantalla.
7. Si ya llegaste hasta este paso tienes activada la doble autenticación.

Clave SSH

Primero tienes que crear una clave SSH, sigue los siguientes pasos para conseguirlo.

1. Ve a la terminal de tu SO
2. Pega el siguiente comando en tu termina y sustituye el correo por el que usas en tu cuenta de GitHub
> `$ ssh-keygen -t ed25519 -C "your_email@example.com"`
3. Te va a pedir ingresar en donde quieres almacenar tu clave, por defecto es en `"/Usuarios/TuUsuario/.ssh/id"`
4. Te va a pedir un frase de seguridad, puedes dejarla vacia
5. Te va a aparecer un mensaje indicando donde se almaceno tu clave, algo asi:
6. Listo ya tienes un clave de SSH generada.

```
Your identification has been saved in /Users/georgina.fraire/.ssh/id_ed25519
Your public key has been saved in /Users/georgina.fraire/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:rt3KW/A7WjZLesbKA/WJu5MDTDqcccCDYOS9UozytYU your_email@example.com
The key's randomart image is:
+--[ED25519 256]--+
| .                 |
|O+ .              |
|O+ +E .           |
|*...O. .          |
|O+...+ .So .      |
|.O O +..OO        |
|. . . O.+B        |
|O.  =*B=+         |
|..  .@%+          |
+----[SHA256]-----
```

Clave SSH

En tu terminal realiza lo siguiente:

1. Copia el contenido de tu clave pública al portapapeles con el siguiente comando, recuerda reemplazar el `id_ssh` por el que tu generaste.
`> pbcopy < ~/.ssh/id_ssh.pub`
1. También puedes abrir el archivo con algún editor de texto y copiar el contenido; por ejemplo, nano con el siguiente comando:
`> nano < ~/.ssh/id_ssh.pub`

Ahora dentro de tu cuenta de GitHub haz lo siguiente:

1. Ir a la configuración de tu cuenta:
 - a. Dirígete a la imagen de tu usuario para desplegar un menú
 - b. Selecciona configuración (Settings)
2. Dentro de las configuraciones selecciona la opción de “Claves SSH y GPU” (SSH Y GPU Keys)
3. Dentro de la sección SSH da clic en el botón “Nueva Clave SSH” (New SSH Key)
4. Agrega un título
5. Pega el contenido de la clave SSH dentro de la sección “Clave” (key)
6. Da clic en “Agregar clave ssh” (Add ssh key)
7. LISTO.

Contenido

¿Qué es Git? ☐

Instalación ☐

Comandos - Locales ☐

Comandos - Remotos ☐

GitHub ☐

Recursos ☒



Recursos

- GIT: <https://git-scm.com/>
- GITHUB: <https://docs.github.com/es>
- CLAVE SSH: <https://docs.github.com/es/authentication/connecting-to-github-with-ssh>
- DOBLE AUTENTIFICACIÓN: <https://docs.github.com/es/authentication/securing-your-account-with-two-factor-authentication-2fa>
- GIT - ATlassian <https://www.atlassian.com/git>



THANK YOU