

# Taller

## Crear un proyecto usando Git y GitHub

---

Objetivo:

Aprender a crear un proyecto local en Git, gestionarlo con ramas y subirlo a GitHub, apoyarse con: <https://www.w3schools.com/git>

### Enunciado del Taller

1. Crear un proyecto llamado miPrimerGit usando Git.
2. Subirlo a GitHub.
3. Realizar las siguientes actividades:
  - a) Crear con Visual Studio Code (VSC) una página web index.html con un saludo e incluir la imagen imagen1.jpg.
  - b) Crear una rama llamada agregar-imagen2, donde se incluya imagen2.jpg.
  - c) Hacer un merge de la rama con main.
  - d) Subir todo a GitHub. Recomendable primero realizar una conexión ssh entre el local y GitHub
  - e) Realizar el ejercicio del siguiente numeral 12.

### Desarrollo Paso a Paso

#### 1. Instalar Git en Windows

a) Descargar desde: <https://git-scm.com/download/win>

verificar que se instaló correctamente: en powershell o en la consola creada por el instalador denominada MINGW64 teclear el siguiente comando:  
`git -version`

Para abrir la consola en Windows: En la lupa buscar **git bash**

Nota: desde esta consola puede teclear todos los comandos Linux

#### 2. Crear una cuenta en GitHub

Ingresar a: <https://github.com>

Registrar: Usuario, correo electrónico y contraseña.

#### 3. Abrir la consola de Git Bash

Buscar Git Bash (se instala con Git). Permite usar comandos de Git y de Linux.

#### 4. Configurar la identidad en el equipo local

```
git config --global user.name "usuario"
```

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

El correo debe ser el mismo usado en GitHub.

#### 5. Crear el proyecto

Usando la consola de Git o PowerShell, puede navegar, crear archivos y directorios similar a Linux; también lo puede hacer con el explorador de windows

```
mkdir miPrimerGit
```

```
cd miPrimerGit
```

Crear index.html y agregar imagen1.jpg en la carpeta.

#### 6. Iniciar el repositorio Git

Estando dentro del directorio creado para el proyecto:

```
git init //crea el repositorio de git para este proyecto
```

```
git add index.html imagen1.jpg
```

```
git commit -m "Primer commit: index.html e imagen1.jpg"
```

#### 7. Crear el repositorio en GitHub

Entrar a GitHub, crear nuevo repositorio: miPrimerGit (vacío, sin README).

#### 8. Vincular proyecto local con GitHub

Se pretende sincronizar el proyecto local con el remoto. Todo o que se realice en el local se refleje en el remoto y viceversa.

**git remote add origin** <https://github.com/usuario/miPrimerGit.git>

remote: mapea repositorios remotos

origin: nombre standard del repositorio remoto

add: añade uno nuevo

<https://github.com/usuario/miPrimerGit.git> repositorio en GitHub

en palabras: mi proyecto local se conecta al repositorio remoto

Nota: usuario debe remplazarse por el usuario real en gitHub

#### **git remote -v**

```
git branch -M main //cambia la rama a main, evita conflictos si la rama se llama master
```

```
git push -u origin main //sube todo el proyecto al repositorio; solo se hace la primera vez,  
después solo usar solamente git push
```

#### 9. Crear una rama para agregar otra imagen

```
git branch
```

```
git branch agregar-imagen2
```

```
git checkout agregar-imagen2
```

Agregar imagen2.jpg y actualizar index.html

```
git add .  
git commit -m "Agrego segunda imagen"
```

## 10. Fusionar la rama con main

```
git checkout main  
git merge agregar-imagen2
```

## 11. Subir cambios a GitHub

```
git push -u origin main
```

## 12. Ejercicio final

Agregar taller.pdf y actualizar repositorio:

```
git add taller.pdf  
git commit -m "Agrego taller.pdf"  
git push  
git pull
```

## Nota 1

```
git push -u origin main
```

Nota: cuando se creó el proyecto Git debe tener cuidado de haber cambiado 'master' por 'main'; de lo contrario use 'master' en lugar de 'main'.

Mucho cuidado: cuando salga la ventana de autorizaciones, inicialmente usar por default el "explorador" y con mucho cuidado y paciencia teclear las credenciales de la cuenta GitHub y esperar hasta que se realice el push. Importante: en este paso no lo vaya a interrumpir.

Si por algún motivo no le funciona, intente hacerlo usando device code o desinstale Git y vuelva a realizar todo el proceso reseteando todo el historial del explorador.

Lo mejor es realizarlo a través de SSH, entonces instalar ssh en el equipo y de esta manera se logra una conexión segura entre el equipo y GitHub.; después todo lo puede manejar gráficamente desde VSC.

## Nota 2: Sincronización entre Git local y GitHub

Existen dos formas de mantener sincronizado el proyecto entre su computadora local y GitHub:

### 1. \*\*Flujo normal (local → GitHub):\*\*

- Realizar cambios en local.
- Ejecutar `git add`, `git commit`.
- Subir con `git push`.

## 2. **\*\*Flujo inverso (GitHub → local):\*\***

- Subir o modificar archivos directamente en GitHub (por ejemplo, crear o subir un PDF desde el navegador).
- Actualizar la copia local con ``git pull``.

### Importante:

- Antes de realizar un ``git push`` desde el local, conviene ejecutar ``git pull`` para traer posibles cambios desde GitHub.
- Si un mismo archivo se modifica en ambos lados sin sincronizar, aparecerá un **\*\*merge conflict\*\***, que se debe resolver manualmente.

# Comandos

## SSH-AGENT

1. Verificar el Servicio ssh-agent
  - a. PowerShell: Get-Service ssh-agent
2. Activar el servicio ssh-agent:
  - a. Debe asegurarse que OpenSSH Client este Instalado
  - b. En Windows: W+R buscar : services.msc  
localizar OpenSSHAuthentication Agent ; activarlo, colocarlo en automatic si esta manual
  - c. Activar el servicio, en Powershell: Start-Service ssh-agent
  - d. Caso de estar instalado en Windows para instalarlo:  
como administrador desde PowerShell:  
Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0

## SSH

### Summary of SSH Concepts and Command desde PowerShell

1. **SSH key pair** - A public and private key for secure access
2. **ssh-keygen** - Generate a new SSH key pair
3. **ssh-add** - Add your private key to the SSH agent
4. **ssh -T git@github.com** - Test SSH connection
  - a. **debe responder algo como:**  
Hi usuariox! You've successfully authenticated
5. **ssh-add -l** - List loaded SSH keys
6. **ssh-add -d** - Remove a key from agent

Realizarlo desde MINGW64 ( consola del git)

Example: Enable SSH Agent

```
eval $(ssh-agent -s)
```

---

## Generating an SSH Key Pair

To create a new SSH key pair, use this command in the terminal (Linux, macOS, or Git Bash for Windows):

### Example: Generate SSH Key

```
ssh-keygen -t rsa -b 4096 -C "your@email.com"
```

Follow the prompts to choose a file location (press Enter to use the default) and set a passphrase (optional, but recommended for extra security).

---

## Adding Your Key to the SSH Agent

After creating your key, add it to the SSH agent so Git can use it:

### Example: Add Key to SSH Agent

```
ssh-add ~/.ssh/id_rsa
```

---

[REMOVE ADS](#)

---

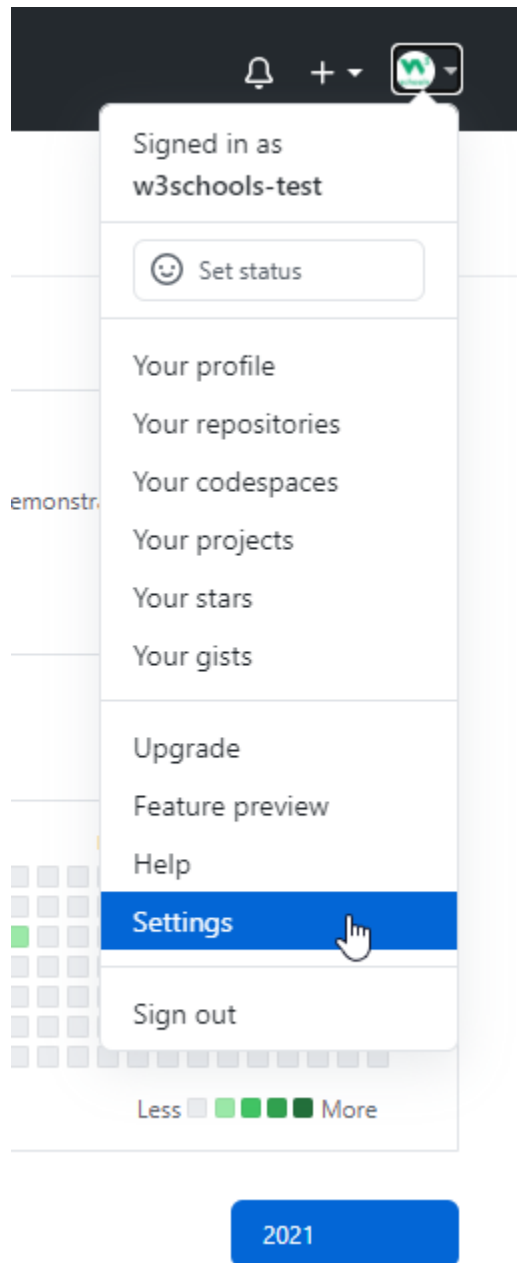
## Copying Your Public Key

To use SSH with Git hosting services, you need to copy your public key and add it to your account settings on GitHub, GitLab, or Bitbucket.

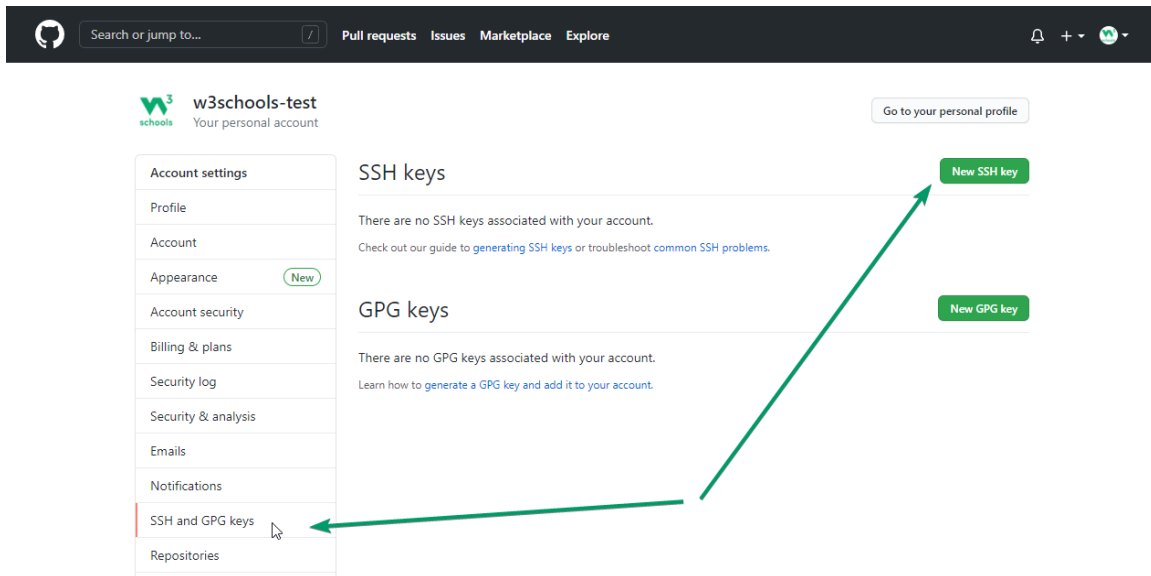
- On macOS: `pbcopy < ~/.ssh/id_rsa.pub`
- On Windows (Git Bash): `clip < ~/.ssh/id_rsa.pub`
- On Linux: `cat ~/.ssh/id_rsa.pub` (then copy manually)

## Add the Key to GitHub

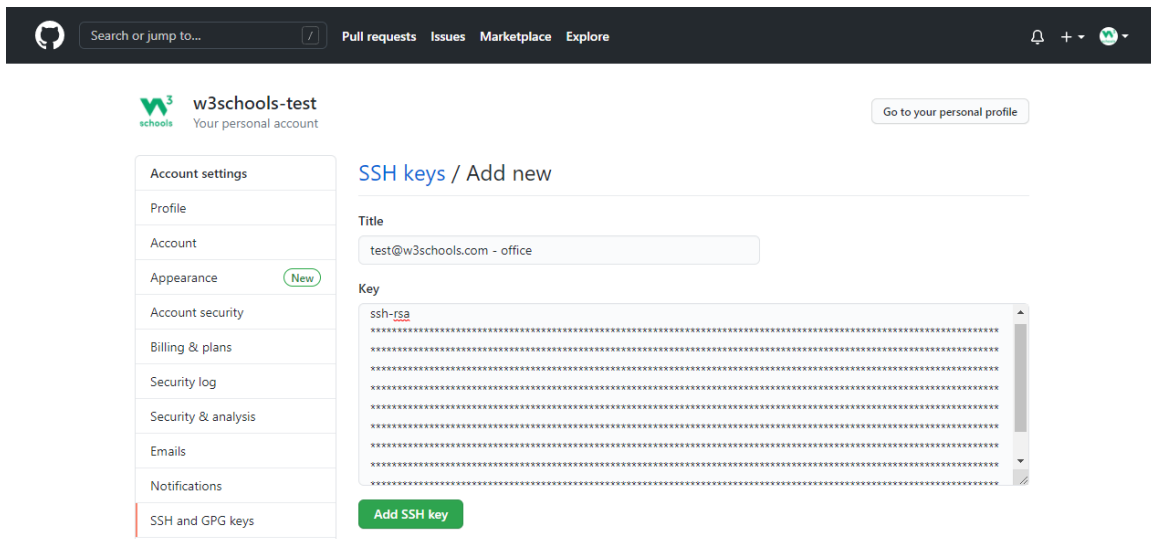
1. Go to GitHub, click your profile in the top right, and select **Settings**:



2. In the sidebar, select **SSH and GPG keys** and click the **New SSH key** button.




3. Give your key a descriptive title, paste your public SSH key into the "Key" field, and click **Add SSH Key**.






You may be prompted to supply your GitHub password or use 2FA to confirm the addition.

You will see your new SSH key listed:





Pull requestsIssuesMarketplaceExplore



**w3schools-test**  
Your personal account

Go to your personal profile

Account settings

Profile

Account

Appearance New

Account security

Billing & plans

Security log

## SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.



**test@w3schools.com - office**  
SHA256: \*\*\*\*\*  
Added on 21 Apr 2021  
Never used — Read/write

Delete

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH problems](#).

Next, you'll test your SSH connection and set your GitHub remote origin using SSH.

## **BIBLIOGRAFIA**

1. <https://www.w3schools.com/>