**✅ Step 1: Download Git for Windows**

1. Go to the official Git website:  
   👉 <https://git-scm.com>
2. Click on the **“Download for Windows”** button.
3. The download should begin automatically (usually a .exe installer).

**✅ Step 2: Install Git**

1. Run the downloaded .exe file.
2. During installation, you can go with most of the **default options**, but here are key screens to pay attention to:
   * **Select Components**: Leave default settings as is.
   * **Choosing the editor**: Select **Visual Studio Code**, **Notepad++**, or keep the default **Vim**.
   * **Adjusting your PATH environment**: Choose  
     🔘 *“Git from the command line and also from 3rd-party software”*.
   * **Choosing HTTPS transport**: Choose  
     🔘 *“Use the OpenSSL library”*.
   * **Configuring line ending conversions**: Choose  
     🔘 *“Checkout Windows-style, commit Unix-style line endings”*.
3. Finish the installation.

**✅ Step 3: Verify Installation**

1. Open **Command Prompt** or **Git Bash**.
2. Type:

git --version

1. You should see the installed Git version like:

git version 2.44.0.windows.1

**✅ Step 4: Configure Git (First-Time Setup)**

Set your user name and email, which will be used for your commits:

git config --global user.name "Your Name"

git config --global user.email "you@example.com"

To verify your config:

git config --list

**✅ Step 5: Using Git Bash or Terminal**

* **Git Bash** is installed automatically. It provides a UNIX-style terminal for using Git and other commands.
* You can also use **Command Prompt**, **PowerShell**, or **Windows Terminal**, but Git Bash is more versatile.

**Check if OpenSSH is Installed**

Open **PowerShell** and run:

ssh -V

You should see something like:

OpenSSH\_for\_Windows\_8.1p1

If it's **not installed**, you can add it from **Settings > Apps > Optional Features > Add a feature** → search for **"OpenSSH Client"** and install it.

**✅ Generate SSH Key using OpenSSH (Windows)**

1. Open **PowerShell** or **Command Prompt**.
2. Run this command:

ssh-keygen -t ed25519 -C "your\_email@example.com"

1. If you're using an older system, use:

ssh-keygen -t rsa -b 4096 -C "your\_email@example.com"

1. Press **Enter** to accept the default path:

C:\Users\YourName\.ssh\id\_ed25519

1. Choose a **passphrase** (optional but recommended), or just press Enter to skip.

**✅ Add SSH Key to GitHub or GitLab**

1. Copy your public key to the clipboard:

Get-Content ~/.ssh/id\_ed25519.pub | Set-Clipboard

(Use type instead of Get-Content if using Command Prompt.)

1. Go to:
   * GitHub: <https://github.com/settings/keys>
   * GitLab: <https://gitlab.com/profile/keys>
2. Paste the key and save it.

**✅ Test SSH Connection**

ssh -T git@github.com

Expected response (first time):

The authenticity of host 'github.com' can't be established...

Are you sure you want to continue connecting (yes/no)? yes

Hi your-username! You've successfully authenticated...

To **push a file to GitHub**, you need to follow a few steps using Git. Here’s a clear, step-by-step guide:

**✅ Prerequisites**

1. **Git is installed** and configured (git config --global user.name etc.).
2. You have a **GitHub account**.
3. You’ve created a **repository** on GitHub (public or private).
4. Optional: SSH key is set up (for secure push), or you use HTTPS with your credentials/token.

**✅ Steps to Push a File to GitHub**

**🔹 Step 1: Create or Navigate to Your Project Folder**

cd path/to/your/project-folder

Or create one:

mkdir my-project

cd my-project

**🔹 Step 2: Initialize Git**

git init

**🔹 Step 3: Add the File**

Let’s say you have a file named example.txt:

git add example.txt

Or to add everything:

git add .

**🔹 Step 4: Commit the Change**

git commit -m "Initial commit"

**🔹 Step 5: Connect to Your GitHub Repository**

You can use **HTTPS** or **SSH**.

**For HTTPS:**

git remote add origin https://github.com/your-username/your-repo.git

**For SSH:**

git remote add origin [git@github.com:your-username/your-repo.git](mailto:git@github.com:your-username/your-repo.git)

**🔹 Step 6: Push to GitHub**

If it's your **first push**:

git push -u origin main

Replace main with master if that’s your default branch.

For later pushes:

git push

**✅ Example**

echo "Hello GitHub" > readme.txt

git init

git add readme.txt

git commit -m "Add readme"

git remote add origin https://github.com/your-username/my-repo.git

git push -u origin main

**💡 Tip**

If you see an error like:

fatal: remote origin already exists.

Run:

git remote set-url origin <new-url>