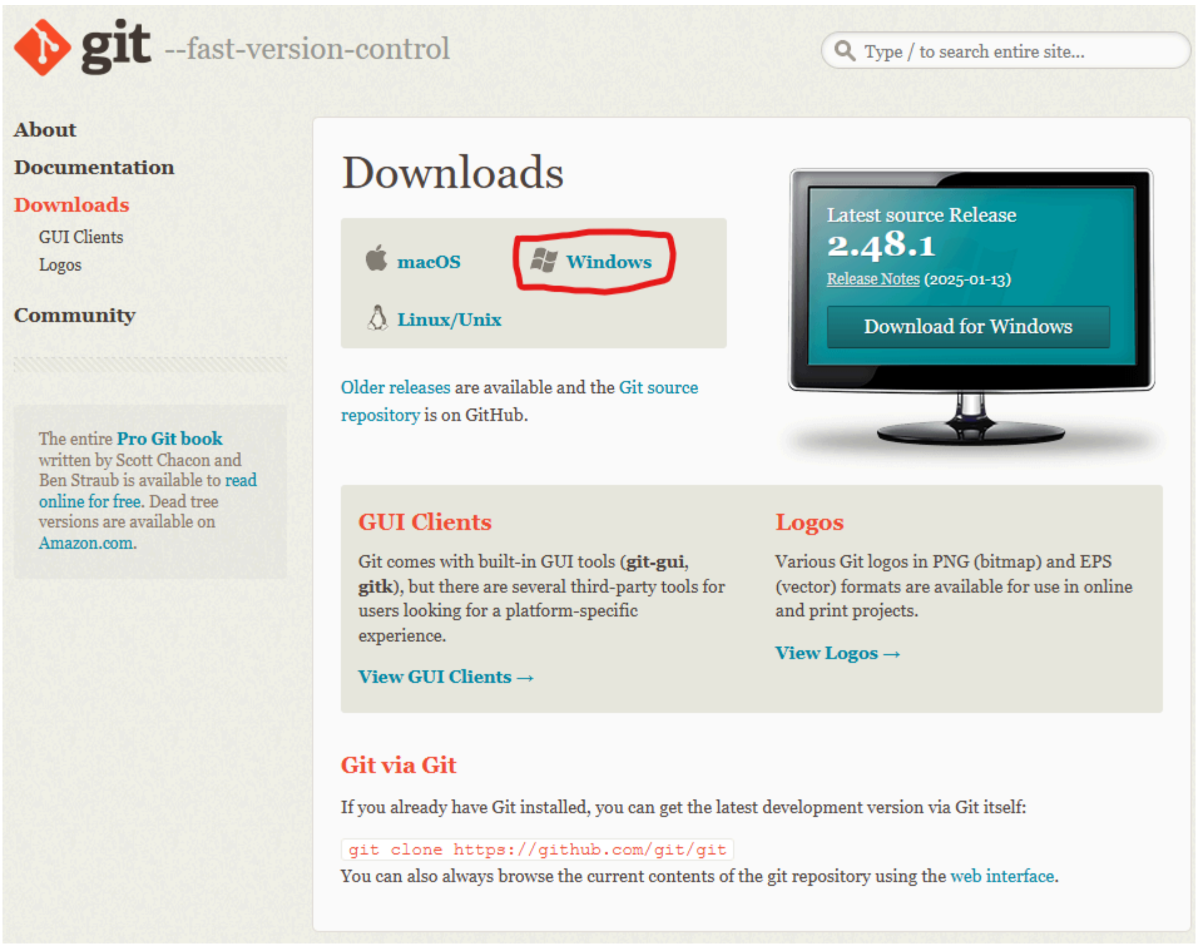
How to Setup Git on Windows and link to GitHub (Step-by-Step Guide)

To use **Git Bash** with **GitHub**, follow these steps:

---

1. Installed Git on your machine

First go to git download page (if you don't have git installed on your machine). Then click on Windows since this is a windows guide after all. Follow through the steps and install git on your machine, unless you know what you are doing don't change any settings along the way.



The entire Pro Git book written by Scott Chacon and Ben Straub is available to read online for free. Dead tree versions are available on Amazon.com.

Downloads

Latest source Release 2.48.1 Release Notes (2025-01-13)

Older releases are available and the Git source repository is on GitHub.

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

View GUI Clients →

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

View Logos →

Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

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

You can also always browse the current contents of the git repository using the web interface.

Open **Git Bash** and run:

```
git --version
```

If you see a version number (e.g., `git version 2.x.x`), Git is installed.
If not, download and install Git.

---

2. Configure Your Git Username & Email

Git needs to know your **GitHub username and email**.
Set them using these commands (replace with your actual details):

```
git config --global user.name "YourGitHubUsername"
git config --global user.email "your-email@example.com"
```

Verify:

```
git config --global --list
```

---

3. Generate & Add SSH Key to GitHub (Recommended)

(A) Check If You Already Have an SSH Key

Run:

```
ls -al ~/.ssh
```

If you see files like `id_rsa.pub` or `id_ed25519.pub`, **skip to Step 3C**.

(B) Generate a New SSH Key

Run:

```
ssh-keygen -t ed25519 -C "your-email@example.com"
```

- If

  ```
  ed25519
  ```

  isn't supported, use:

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

- When prompted for a file location, **press Enter** (default: `~/.ssh/id_ed25519`).
- If asked for a **passphrase**, you can set one or leave it blank.

(C) Add Your SSH Key to GitHub

1. Copy the SSH key:

   ```
   cat ~/.ssh/id_ed25519.pub
   ```

2. Go to **GitHub** → **Settings** → **SSH and GPG keys** (direct link).
3. Click **"New SSH key"**, paste the copied key, and save.

(D) Test the SSH Connection

Run:

```
ssh -T git@github.com
```

If successful, you'll see:

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

---

4. Clone a Repository (optional)

If you want to download a GitHub repository:

```
git clone git@github.com:your-username/repository-name.git
```

or using HTTPS (if SSH isn't set up):

```
git clone https://github.com/your-username/repository-name.git
```

---

5. Link an Existing Project to GitHub

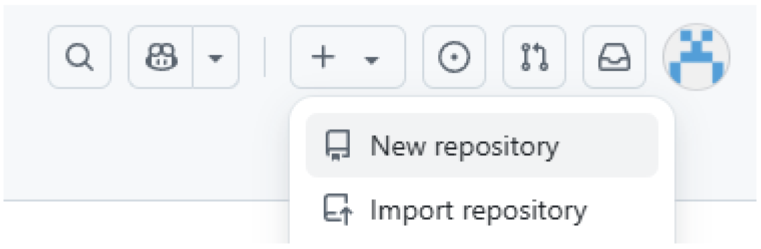
(A) Initialize Git in Your Project

```
cd /d/path/to/your/project
git init
```

You should see `(master)` highlighted in blue after your repository, if you see anything else relate to this document.

(B) Add Remote Repository

Start a new repository in Github by click the "+" on the top right of your Github page, and follow through the steps.



If you already created a repo on GitHub, link it:

```
git remote add origin git@github.com:your-username/repository-name.git
```

Verify:

```
git remote -v
```

---

6. Push Your Code to GitHub

```
git add .
git commit -m "Initial commit"
git push -u origin main  # (or master, depending on your branch)
```

If using HTTPS, GitHub may ask for your username/password or a **personal access token**.

Additional Notes

1. Rejected Push

If you see this error when you `push` a change to a branch on GitHub:

```
$ git push -u origin main
To github.com:ctrlA-ctrlC-ctrlV/E-Commerce-Platform.git
 ! [rejected]        main -> main (fetch first)
error: failed to push some refs to 'github.com:ctrlA-ctrlC-ctrlV/E-Commerce-
Platform.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

this happens because **your local `main` branch is behind the remote `main` branch**. This means:

- Someone else has pushed changes to GitHub, or
- You initialized your repo after some files were already added to GitHub.

Git is preventing you from **overwriting changes on GitHub**.

How to Fix It

✅ Step 1: Pull the Latest Changes from GitHub

Run:

```
git pull origin main --rebase
```

- This fetches and **merges** remote changes into your local branch.
- The `--rebase` option keeps your commits on top of the new updates.

✅ Step 2: Resolve Any Merge Conflicts (If Needed)

If Git reports **merge conflicts**, manually edit the conflicting files to keep the right changes.
Once resolved, run:

```
git add .
git commit -m "Enter Your Message Here"
```

✅ Step 3: Push Your Changes

Now that your local branch is up to date, push again:

```
git push origin main
```

Alternative (Force Push - ⚠️ Dangerous)

If you **don't** want to merge changes from GitHub and want to **overwrite** everything, you can force push:

```
git push --force origin main
```

⚠️ **Warning:** This will **overwrite** the existing files on GitHub, potentially deleting work done by others.