# CONNECTING BASH TO GITHUB USING SSH

Great question! Setting up Git Bash and linking it to your GitHub account is a straightforward process. Here's a step-by-step guide to help you get started:

---

Step 1: Configure Git in Git Bash\*\*

1. \*\*Open Git Bash\*\*:

- Launch Git Bash from your Start menu or desktop shortcut.

2. \*\*Set Your Username\*\*:

**-** *Configure your Git username (this should match your GitHub username):*

*```bash*

**git config --global user.name "Your GitHub Username"**

**```**

3. \*\*Set Your Email\*\*:

*- Configure your Git email (this should match the email associated with your GitHub account):*

*```bash*

**git config --global user.email "your-email@example.com**"

```

4. \*\*Verify Your Configuration\*\*:

*- To check if your settings are correct, run:*

*```bash*

**git config --list**

```

- You should see your `user.name` and `user.email` in the output.

Step 2: Generate an SSH Key (for Secure Authentication)\*\*

To securely connect Git Bash to GitHub, you'll need to generate an SSH key and add it to your GitHub account.

1. \*\*Generate a New SSH Key\*\*:

*- Run the following command in Git Bash (replace `your-email@example.com` with your GitHub email):*

*```bash*

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

```

- *If your system doesn't support `ed25519`, use:*

*```bash*

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

```

2. \*\*Save the SSH Key\*\*:

*- When prompted to "Enter a file in which to save the key,"* **press `Enter` to accept the default location (`~/.ssh/id\_ed25519` or `~/.ssh/id\_rsa`).**

3. \*\*Set a Passphrase (Optional)\*\*:

- **You can add a passphrase for extra security, or press `Enter` to skip**.

4. \*\*Start the SSH Agent\*\*:

- *Run the following command to start the SSH agent:*

*```bash*

**eval "$(ssh-agent -s)**"

```

5. \*\*Add Your SSH Key to the SSH Agent\*\*:

- *Add your private key to the SSH agent:*

*```bash*

**ssh-add ~/.ssh/id\_ed25519**

```

- ***If you used `rsa`, replace `id\_ed25519` with `id\_rsa`***.

6. \*\*Copy the SSH Key\*\*:

- *Copy the contents of your public key to your clipboard:*

*```bash*

**cat ~/.ssh/id\_ed25519.pub**

```

- ***Highlight the output and copy it (it starts with `ssh-ed25519` or `ssh-rsa`).***

---

Step 3: Add the SSH Key to Your GitHub Account\*\*

1. \*\*Go to GitHub\*\*:

- **Log in to your GitHub account and go to \*\*Settings\*\* > \*\*SSH and GPG keys\*\***.

2. \*\*Add a New SSH Key\*\*:

**- Click \*\*New SSH Key\*\* or \*\*Add SSH Key\*\*.**

**- Give it a title (e.g., "My Laptop").**

**- Paste the SSH key you copied earlier into the "Key" field**.

3. \*\*Save the Key\*\*:

- **Click \*\*Add SSH Key\*\* and confirm your GitHub password if prompted.**

---

### \*\*Step 4: Test the SSH Connection\*\*

1. \*\*Test the Connection\*\*:

*- Run the following command in Git Bash to test the SSH connection:*

*```bash*

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

```

- *If everything is set up correctly, you'll see:*

```

**Hi <your-username>! You've successfully authenticated, but GitHub does not provide shell access**.

```

---

Step 5: Clone a Repository (Optional)\*\*

Now that Git Bash is linked to GitHub, you can clone a repository to your local machine.

1. \*\*Find a Repository on GitHub\*\*:

- **Go to the repository you want to clone and click the \*\*Code\*\* button.**

**- Copy the SSH URL (e.g., `git@github.com:username/repository-name.git`)**.

2. \*\*Clone the Repository\*\*:

- *In Git Bash, navigate to the directory where you want to clone the repository:*

*```bash*

**cd /path/to/your/directory**

```

- *Clone the repository using the SSH URL:*

*```bash*

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

```

3. \*\*Verify the Clone\*\*:

- *Navigate into the cloned repository:*

*```bash*

**cd repository-name**

```

- *Check the files:*

*```bash*

**ls**

```

---

Step 6: Make Changes and Push to GitHub\*\*

1. \*\*Make Changes\*\*:

- **Edit files in your repository using your preferred text editor or IDE.**

2. \*\*Stage Changes\*\*:

*- Stage the changes you want to commit:*

*```bash*

**git add** .

3. \*\*Commit Changes\*\*:

- *Commit the changes with a message:*

*```bash*

**git commit -m "Your commit message"**

4. \*\*Push Changes\*\*:

*- Push the changes to GitHub:*

*```bash*

**git push origin main**

- If your default branch is `master`, replace `main` with `master`.

Done!