**1. Set Up Git in WSL**

If you haven't already installed Git, you can do so by running:

sudo apt update

sudo apt install git

Verify the installation:

git --version

**2. Configure Git**

Set up your Git username and email (these will be associated with your commits):

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

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

You can check your Git configuration with:

git config --list

**3. Generate an SSH Key (Optional but Recommended)**

Using SSH keys allows you to interact with GitHub without entering your credentials every time.

1. Generate a new SSH key:

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

Press Enter to accept the default file location and passphrase.

1. Add the SSH key to the SSH agent:

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

1. Copy the SSH key to your clipboard:

cat ~/.ssh/id\_ed25519.pub

Highlight and copy the output.

1. Add the SSH key to your GitHub account:
   * Go to GitHub > Settings > SSH and GPG keys.
   * Click "New SSH key," give it a title, and paste the key.
2. Test the SSH connection:

ssh -T git@github.com

You should see a success message.

**4. Create a New GitHub Repository**

1. Go to GitHub and log in to your account.
2. Click the "+" icon in the top-right corner and select "New repository."
3. Fill in the repository name, description, and choose public/private visibility.
4. Click "Create repository."

**5. Clone the Repository to WSL**

1. Copy the repository URL (HTTPS or SSH) from GitHub.
2. Clone the repository to your WSL environment:

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

Replace git@github.com:username/repository-name.git with your repository's URL.

1. Navigate into the cloned repository:

cd repository-name

**6. Add Files and Commit Changes**

1. Create or modify files in the repository directory.
2. Stage the changes:

git add .

Or stage specific files:

git add filename

1. Commit the changes:

git commit -m "Your commit message"

**7. Push Changes to GitHub**

Push your changes to the remote repository:

git push origin main

(Replace main with the name of your default branch if it's different.)

**8. Pull Changes from GitHub**

To update your local repository with changes from GitHub:

git pull origin main

**9. Manage Branches**

* Create a new branch:

git branch new-branch-name

* Switch to a branch:

git checkout branch-name

* Push a new branch to GitHub:

git push origin branch-name

**10. Additional Tips**

* Use git status to check the status of your repository.
* Use git log to view the commit history.
* Use .gitignore to exclude files from being tracked by Git.