

1.3 Set Up and Configure GitHub Account

This section will guide you to:

- Create your GitHub account
- Connect from local Git repository to remote GitHub via SSH
- Initiate the first push

This guide has four subsections, namely:

- 1.3.1 Setting up your GitHub account
- 1.3.2 Creating SSH-key and adding it to GitHub
- 1.3.3 Logging at local Git to connect with remote GitHub
- 1.3.4 Creating a repository in your GitHub account

Step 1.3.1: Setting up your GitHub account.

About GitHub: It is a web-based hosting service for version control using Git. It offers plans for public and private repositories. You can add multiple projects by creating multiple public repositories. In this section, we will only demonstrate how public repository works.

Navigate to <https://github.com/> and click on **Sign up for GitHub**. Enter the details and click on **Create an account**.

Step 1:
Create personal account

Step 2:
Choose your plan

Step 3:
Tailor your experience

Create your personal account

There were problems creating your account.

Username *

Email address *

Email can't be blank

Password can't be blank

You'll love GitHub

Unlimited collaborators
Unlimited public repositories

✓ Great communication
✓ Frictionless development
✓ Open source community

Once you're at **Step 2: Choose your personal plan**, Select **Free**, and click on **Continue**. You can share basic information about yourself or you can select **skip this step**.

You must have received an email to confirm your account. It is important to confirm your account before you use GitHub. Once you have confirmed, you are successfully signed for GitHub.

Step 1.3.2: Creating an SSH Key and adding it to GitHub.

You can create ssh-key via **Git bash** by following the steps:

- Open your **Git bash**
- Execute the command:

```
ssh-keygen -t rsa -b 4096 -C "<your email address>" → press [enter]
```

- Do not enter anything but [enter] until the setup is complete
- Start the ssh-agent in the background:

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

- Add your SSH private key to the ssh-agent

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

- Copy the SSH key to your clipboard

```
clip < ~/.ssh/id_rsa.pub
```


- Copy the entire key from the clipboard. Choose **Your avatar > settings > SSH & GPG Keys** and click on **New SSH key** and paste the key and **save** it

Personal settings
Profile
Account
Emails
Notifications
Billing
SSH and GPG keys
Security
Sessions
Blocked users
Repositories

SSH keys

New SSH key

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



Keykey
08:f2:14:b3:bc:aa:94:14:a6:97:6b:fa:d7:f7:29:de
Added on Nov 20, 2018
Last used within the last week — Read/write

SSH

Delete

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

GPG keys

New GPG key

There are no GPG keys associated with your account.

Learn how to [generate a GPG key and add it to your account](#).


Step 1.3.3: Logging at local Git to connect with remote GitHub.

Open the Git terminal and execute the commands below by replacing **your_Email_Id** with your registered email address with GitHub and replace **Your_Username** with your GitHub username.

```
#git config --global user.email "your_Email_Id"
#git config --global user.username "Your_Username"
```

Step 1.3.4: Creating a repository in your GitHub account.

Go to the homepage of GitHub.com and click on **New Repository** as shown below:


Search or jump to...
Pull requests
Issues
Marketplace
Explore

Repositories
New repository

Find a repository...
SimplelearnDevO... /Lesson-04-JUn...

Browse activity
Discover repositories

Discover interesting projects and people to populate your personal news feed.
Your news feed helps you keep up with recent activity on repositories you watch and people you follow.
Explore GitHub

Enter the name file and click on **Create repository** as shown in the example screenshot mentioned below:

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: **SimplilearnDevOpsOfficial** / Repository name: **Lesson -02 --GitHubFiles** ✓

Great repository names are short and meaningful. Your new repository will be created as **Lesson--02--GitHubFiles**

Description (optional):

☒ **Public**
Anyone can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

Create repository

You will be redirected to a quick guide page and inside the directory you have created.

Quick setup — if you've done this kind of thing before

Set up in Desktop or **HTTPS** **SSH** `git@github.com:SimplilearnDevOpsOfficial/Lesson--02--GitHubFiles.git`

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# Lesson--02--GitHubFiles" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:SimplilearnDevOpsOfficial/Lesson--02--GitHubFiles.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:SimplilearnDevOpsOfficial/Lesson--02--GitHubFiles.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Since a repository is already created, “**...or create a new repository on the command line**” should be skipped. Click on **SSH** to change the instructions from **HTTPS** to **SSH** (if you're using Ubuntu).

Copy the `git remote add origin <URL_of_Your_GitHub_Repository>` and execute it in the terminal as shown in the example command mentioned below:

```
git remote add origin git@github.com:SimplilearnDevOpsOfficial/Lesson--02--GitHubFiles.git  
git push -u origin master
```