Lesson 3 Demo1: Deploy to GitHub via Git

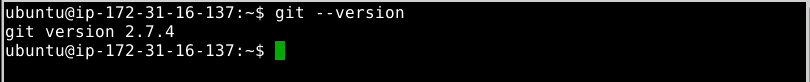
This section will guide you to:

* Install Git and set up your GitHub account
* Execute the most popular commands in Git
* Push all the files from the local repository to GitHub

**Step 1:** Install Git

**Note:** Git is already installed in your lab. You can check the version of Git by executing the below command in the terminal:

**git --version**



* If Git is not installed, then you can follow the below steps to install Git:

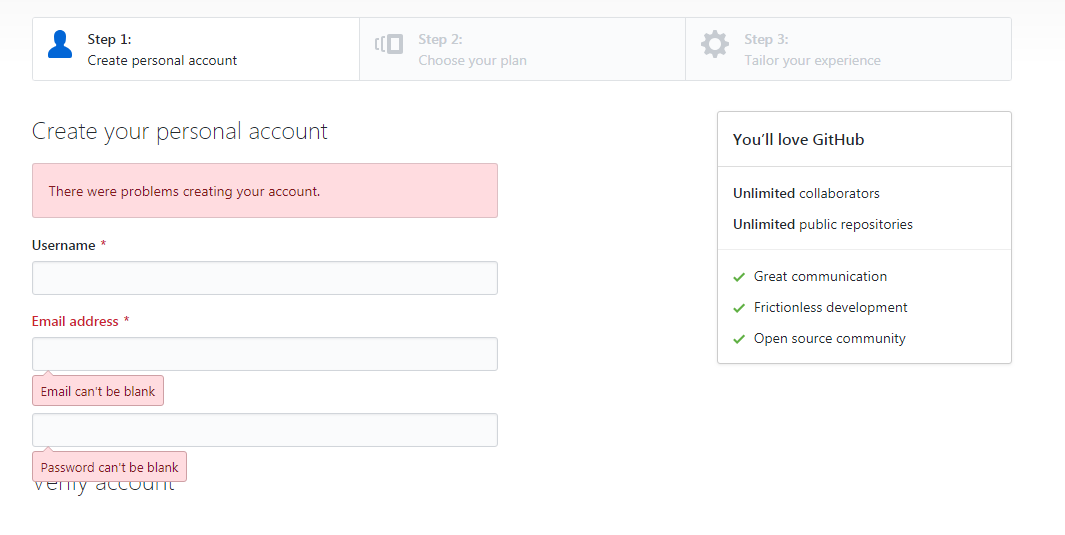
**sudo apt-get install git**



**Step 2:** Set up your GitHub account

**Note: GitHub** 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. This section only covers the public repository and its usage.

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



* In **Choose your personal plan**, select **Free** and click on **Continue**. You can share basic information about yourself or you can **Skip this step**

**Note:** You will receive an email to confirm your account. It is important to confirm your account before you use GitHub. Once confirmed, your GitHub account is set up successfully.

**Step 3:** Log in from Git local to connect remote GitHub

* Open the terminal in your lab and execute the following commands:

**git config --global user.email “your\_Email\_Id”**

**git config --global user.username “your\_Username”**

**Note:** Replace **your\_Email\_Id** with your registered email address in GitHub and **Your\_Username** with your GitHub username.



**Step 4:** Create multiple files and content in each file

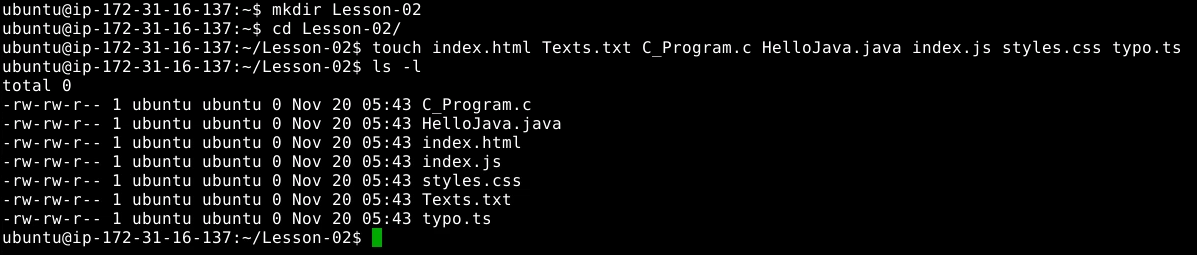
* To create multiple files with different extensions and to create a folder to store all the files in one place, follow the steps shown below:

**mkdir Lesson-02**

**cd Lesson-02**

**touch index.html Texts.txt C\_Program.c HelloJava.java index.js styles.css typo.ts**

**ls -l**



**Note:** You can use any of the text editors available in Linux, but **vi** editor is preferred. To open the **vi** in editor mode, follow the below step. You can execute the same command for any file extension you have created in the above step.

* Execute **vi C\_Program.c** in the terminal andadd the below code in **C\_Program.c** file:

**#include<stdio.h>**

**int main(){**

**printf(“Hello! I am C-Program. Thank you!”);**

**return 0;**

**}**

* Execute **vi HelloJava.java** in the terminal andadd the below code in **HelloJava.java** file:

**class HelloJava {**

**public static void main(String args[]){**

**System.out.Println(“ I am your Java Program. Thank you! ”);**

**}**

**}**

* Execute **vi index.html** in the terminal and add the below code in **index.html** file:

**<html>**

**<head><title>HTML</title></head>**

**<body>**

**<p> I am your HTML Page. Thank you! </p>**

**</body>**

**</html>**

* Execute **vi index.js** in the terminal and add the below code in **index.js** file:

**var texts = “I am your JavaScript Program”;**

**console.log(texts);**

* Execute the below commands to edit the other files:

**vi Styles.css**

**vi Texts.txt**

**vi typo.ts**

**Note:** Please note that these files will be empty.

**Step 5:** Initialize Git

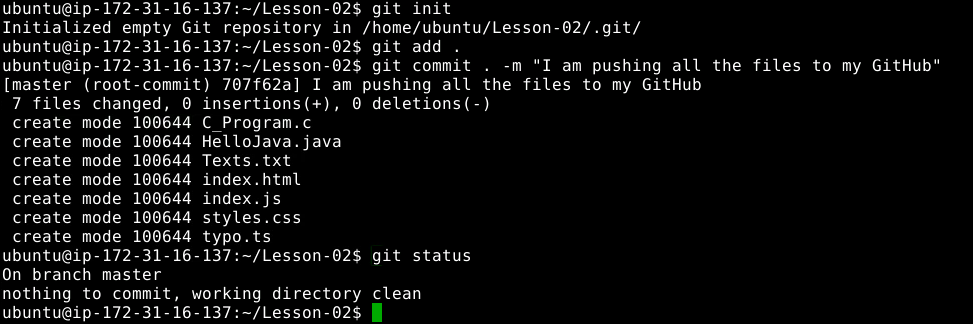
* Since all the files are to be pushed, initialize a **.git** folder inside the directory by executing the following commands:

**git init**

**git add .**

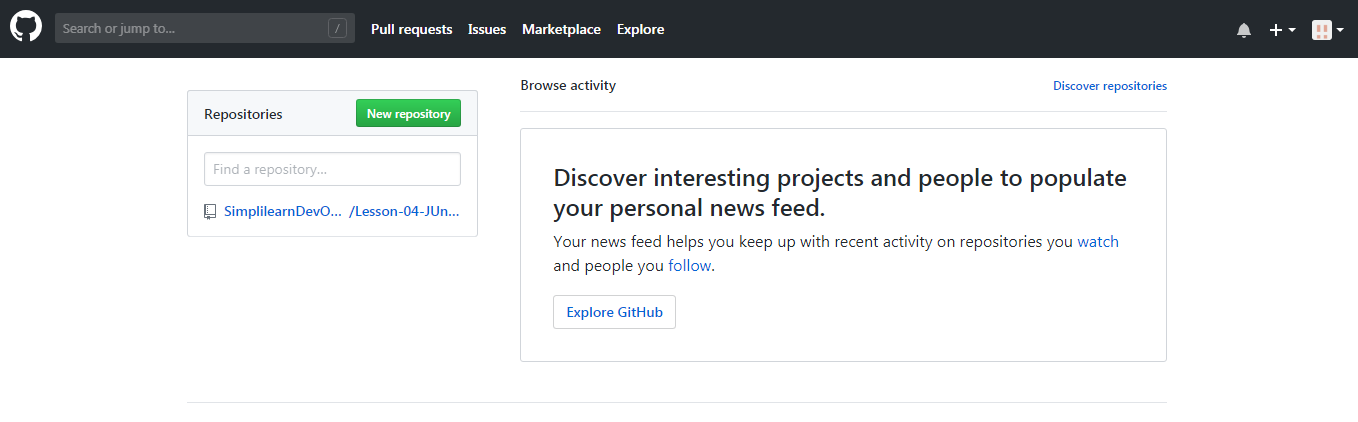
**git commit . -m “I am pushing all the files to my GitHub”**

**git status**

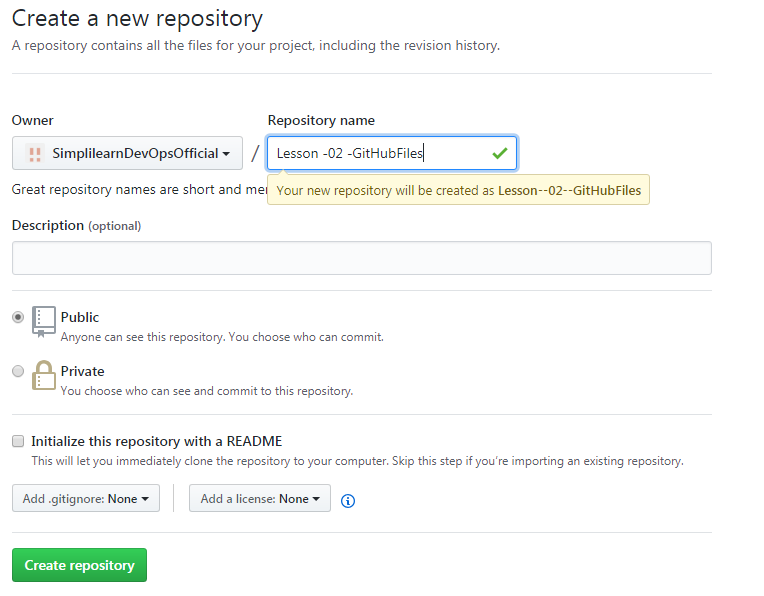


**Step 6:** Create a repository in your GitHub account

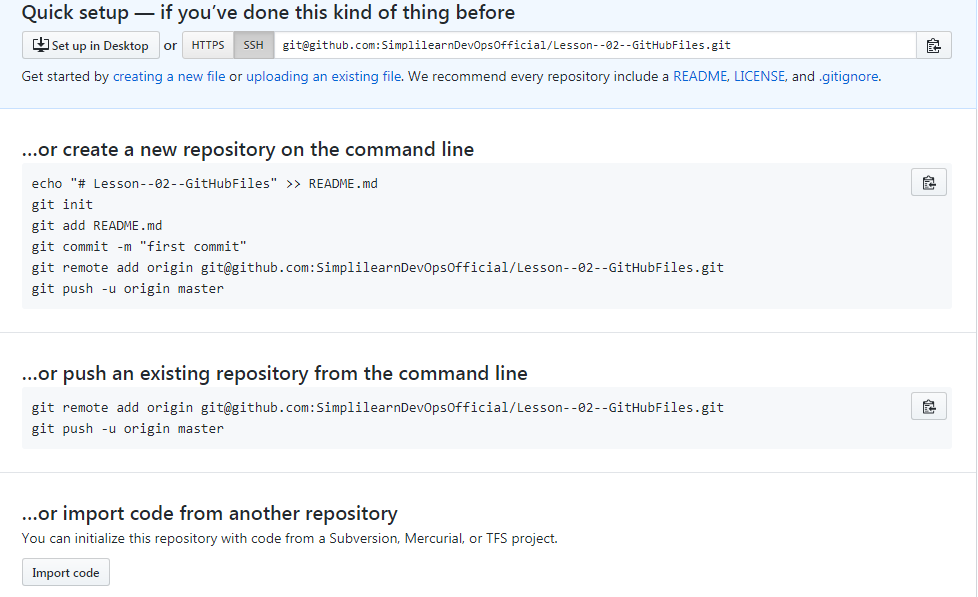
* Go to the homepage of **GitHub.com** and click on **New repository** as shown below



* Enter the name as **Lesson -02- GitHubFiles** and click on **create repository**



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



* 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**
* Copy the Git remote, add origin **<URL\_of\_Your\_GitHub\_Repository>** and execute it in the terminal

**git remote add origin <URL of Your GitHub Repository>**

**git push -u origin master**

**Note:** If you’re unable to push the files to your Github.com account, then follow the below steps:

**Step 7:** Creating SSH Key and adding it to GitHub

* Switch the current directory to ssh by executing below command:

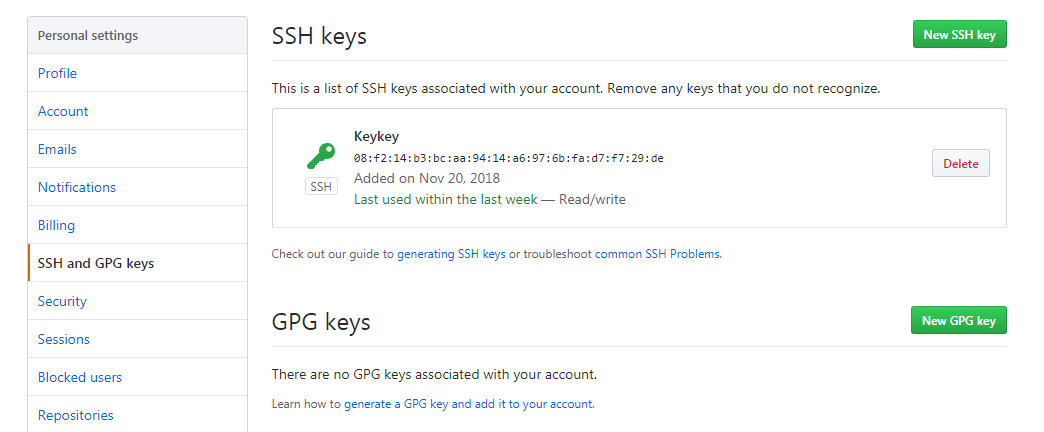
**cd ~/.ssh**

* Generate an RSA key for the registered email Id (An example is available below)

**ssh-keygen -t rsa -C <YourEmailId>**

**vi id\_rsa.pub**

* Copy the entire key from the clipboard. Choose ***Your avatar* > Settings** > **SSH & GPG Keys** and click on **New SSH** **key**, paste the key, and **Save** it.



* In the terminal, execute the following command to save the key and link it with local git:

**ssh-add**

* Copy the git remote, add origin **<URL\_of\_Your\_GitHub\_Repository>**, and execute it in the terminal

**git remote add origin <URL of Your GitHub Repository>**

**git push -u origin master**

* Reload your **GitHub.com** account to confirm the output shown below:

