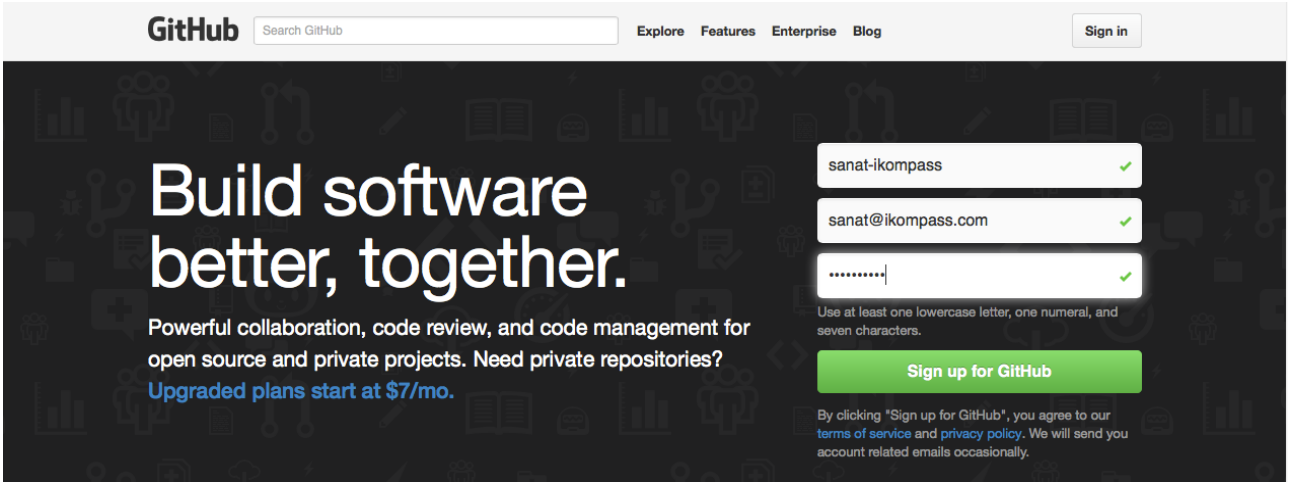
ANNEXTURE - 1

Disclaimer :- The screenshots provided in this tutorial may not be the same when you access the actual webpage as they are frequently updated.

Sign up for Free GitHub account

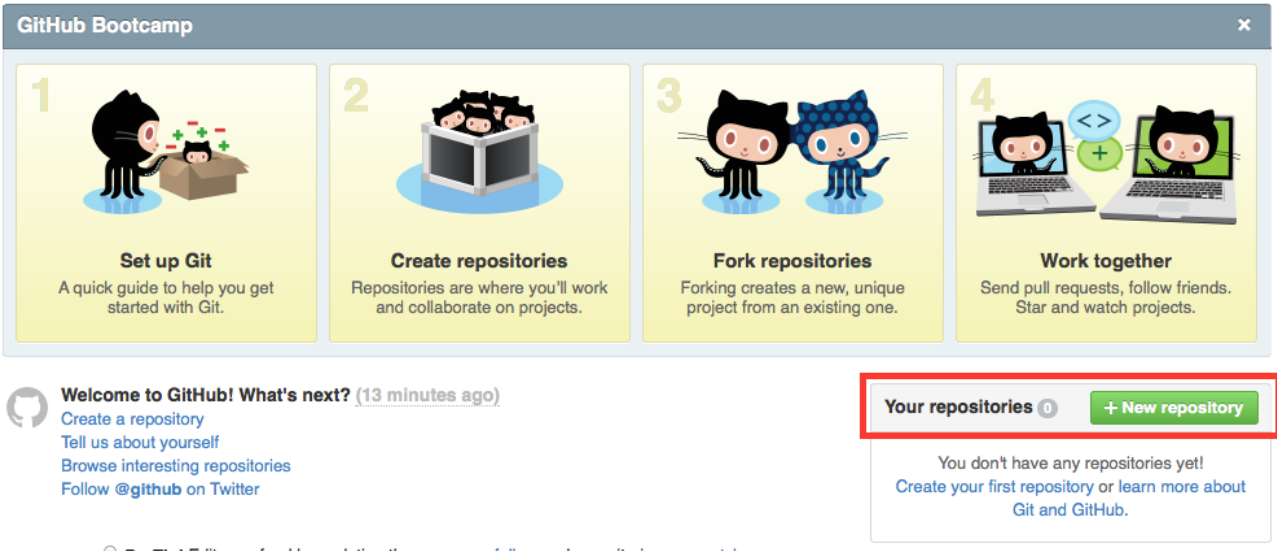
1. Log on to https://github.com and enter your details and click on "Sign up for GitHub" button, to create a GitHub account.



2. On the next page, you will be asked to choose a plan. By default, a free plan is chosen for you. Click "Finish sign up" button and you are done.

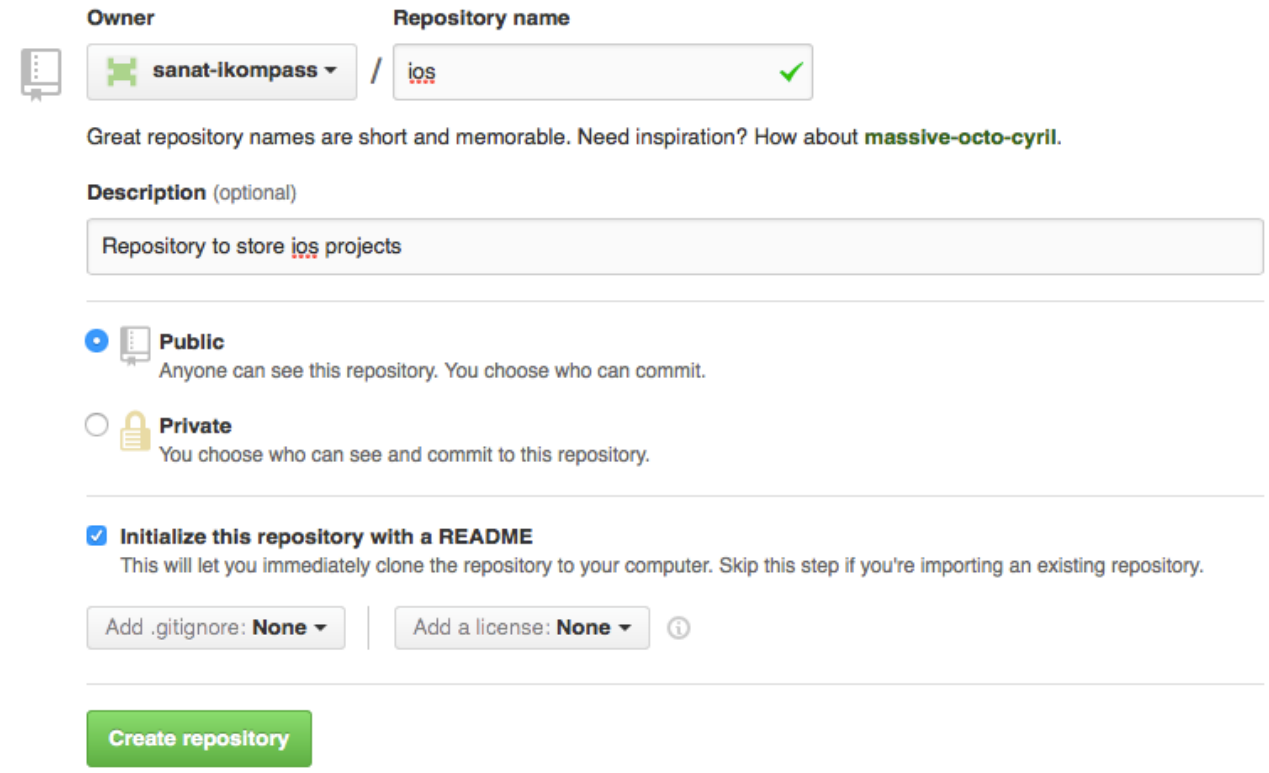
Creating your repository

1. A repository is where your files will be placed. On the GitHub account's welcome page, click "New Repository" button.

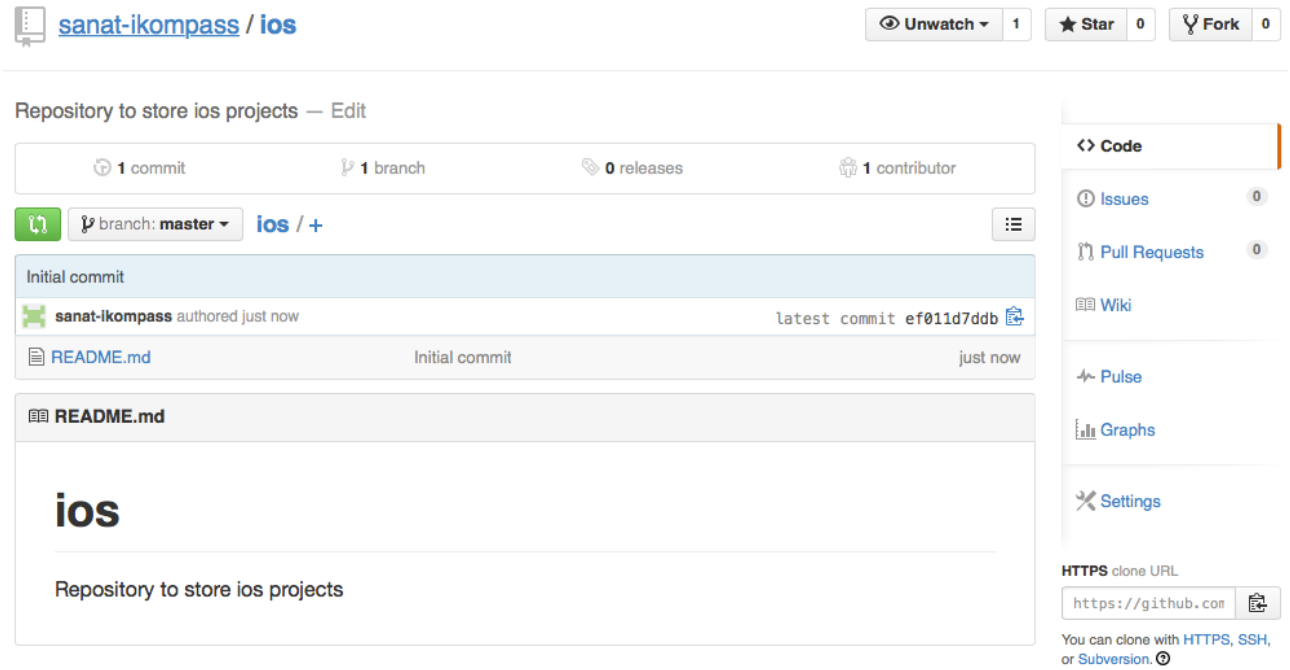


2. On the "Create Repository" page, enter the following details:

   A. Name of the repository
   B. Description for the repository
   C. Make the repository "Public" by choosing that option. If you want to make it "Private", you may need to upgrade your account by paying some money.
   D. Check the option - Initialize the repository with a README



Once done, click the Create repository button. If everything goes well, you should be seeing your repository screen as follows:



Note:- You must verify your email address before you can start using GitHub account.

`Generating SSH keys for your GitHub account`

The GitHub account must trust your computer before it can allow files from it to be added to the repository. SSH keys are a way to identify trusted computers, without involving passwords. Follow the steps below to generate a SSH key:

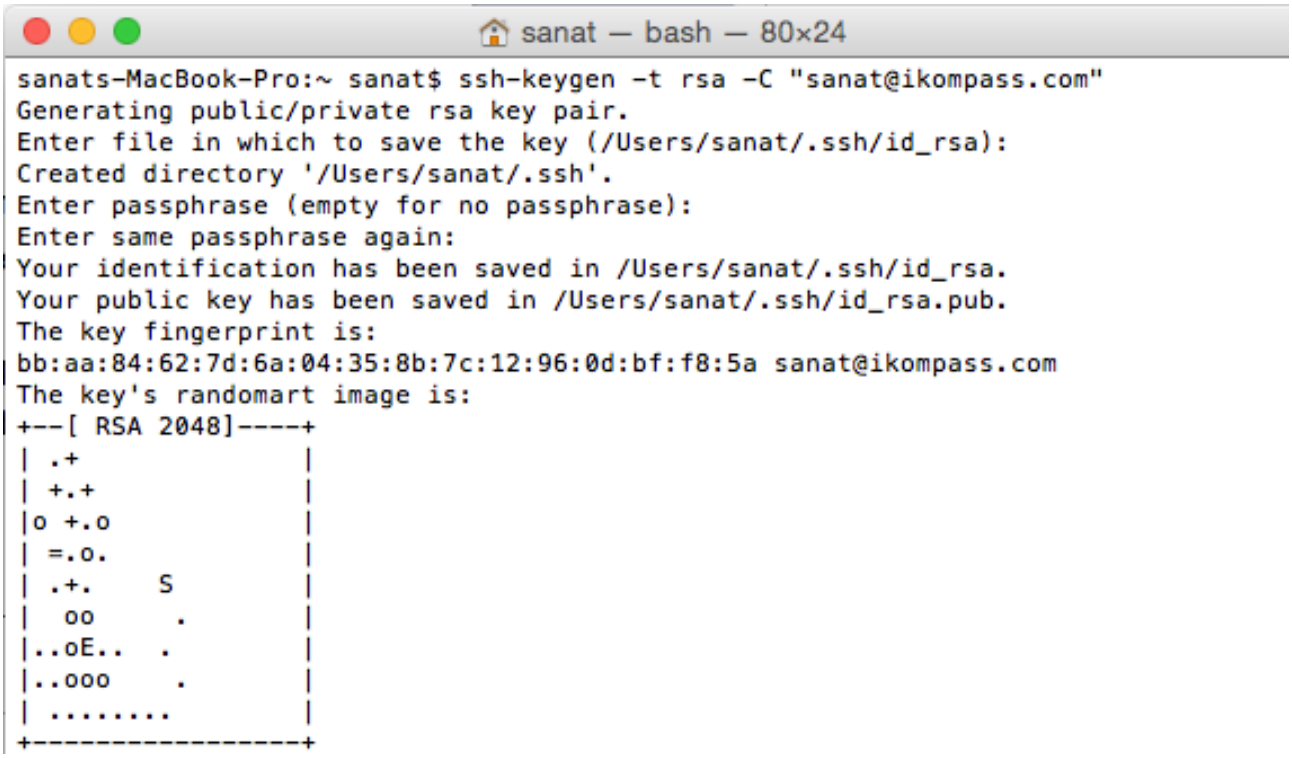
1. On a Unix based system (like Mac OS), the ssh keys are usually stored in ".ssh" folder, within your home directory and path for it looks like: ~/.ssh
2. We can use ssh-keygen tool to generate the ssh key as follows:
   i. Open terminal and type the following command:

```
ssh-keygen -t rsa -C "sanat@ikompass.com"
```

   Forget not to use your own email id in the above command.

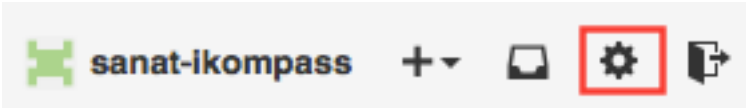
   ii. If asked to choose a location, just hit Enter. We want default settings for this step.
   iii. After this, you will be asked to provide a passphrase. Please use a strong passphrase at this step and don't leave it blank.

Confirm the same passphrase in the next step and hit Enter. An RSA key will be created for your.

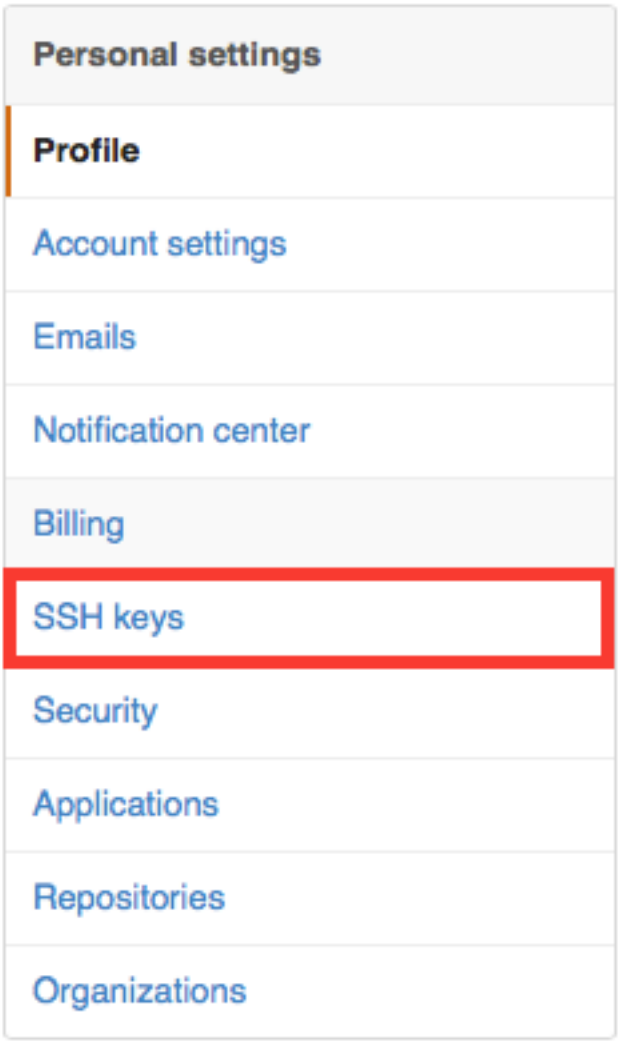


3. Every ssh key has a public-private pair. The public key needs to be added to your GitHub account. To do that open the public key file using your text editor and copy the key from there. It is usually named : **id_rsa.pub** and found in: **~/.ssh** directory

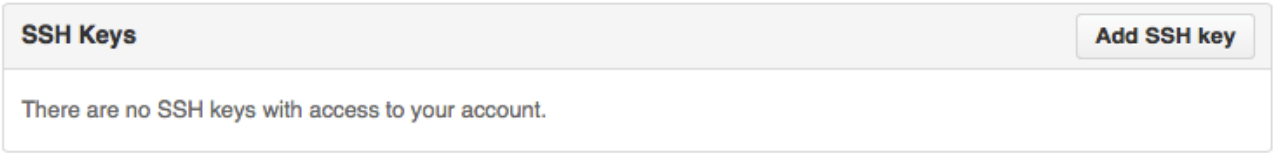
4. In your GitHub account, open the Settings page by clicking its icon (in the top-right corner):



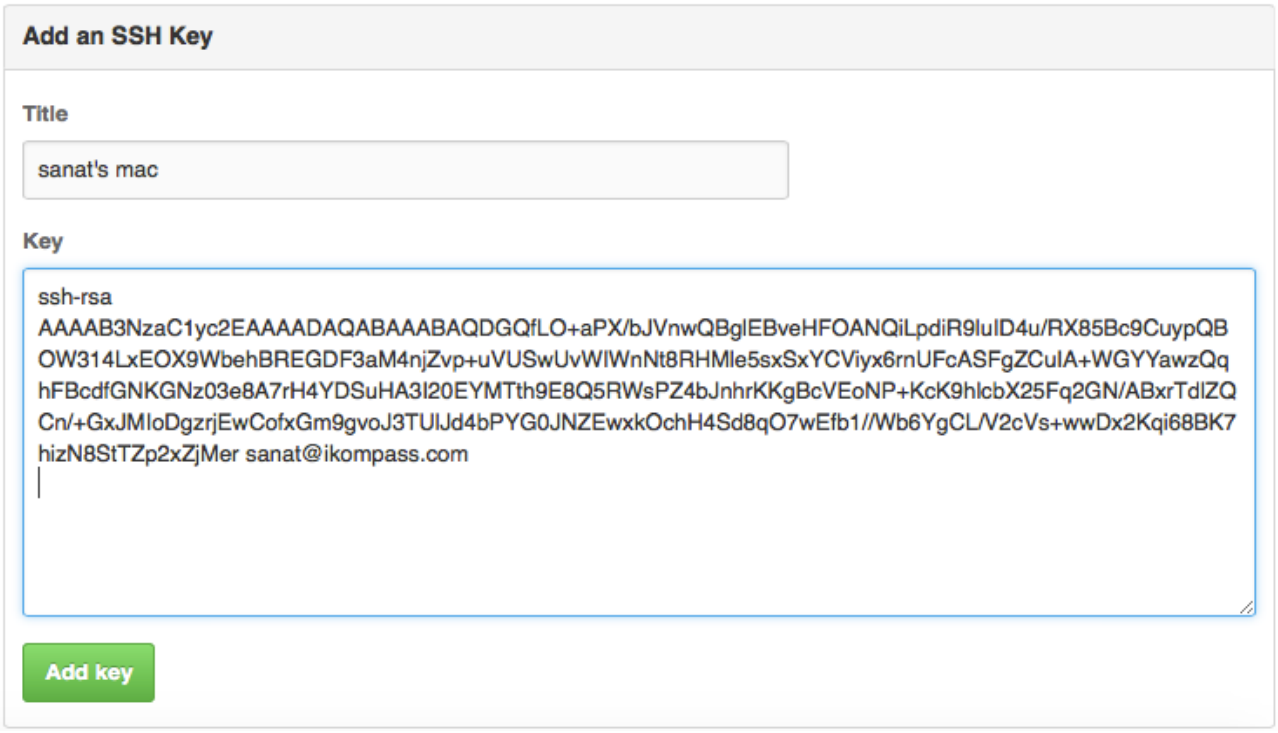
5. In the Settings page, click **SSH keys** :



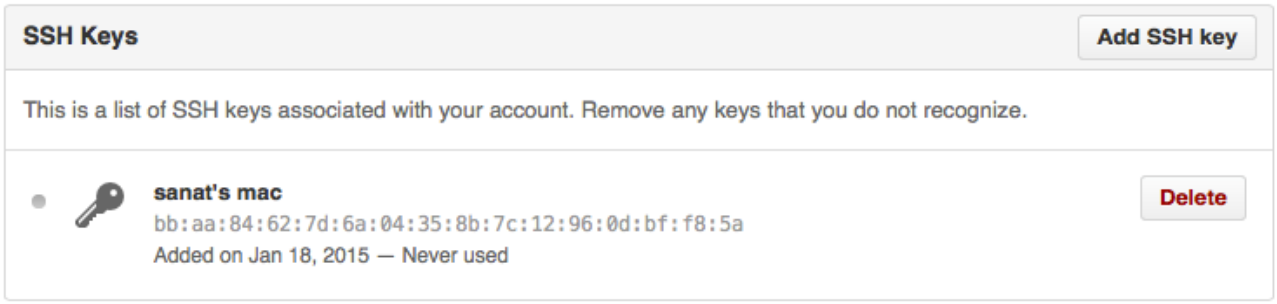
6. Click "Add SSH key" button



7. In the Title field, add a descriptive label for the new key. In the Key field, paste your public key:



8. Click Add key button to confirm your entry. On successful addition of the key, you should see the same in your account as follows:



9. Validate the settings by running the following command in your terminal: **ssh -T git@github.com**

You may be prompted with the following message:

```
The authenticity of host 'github.com (192.30.252.130)' can't be
established.
RSA key fingerprint is 16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.
Are you sure you want to continue connecting (yes/no)?
```

Hit Enter and wait for the confirmation to come from the github server. If you see a message like below, you are all set:

```
Warning: Permanently added 'github.com,192.30.252.130' (RSA) to the list
of known hosts.
Identity added: /Users/sanat/.ssh/id_rsa (/Users/sanat/.ssh/id_rsa)
Hi sanat-ikompass! You've successfully authenticated, but GitHub does
not provide shell access.
```

Configuring Xcode to use GitHub account as File Repository

Xcode offers a GUI tool to manage the source control but unfortunately it treats even the Git repository as Subversion (another popular open source SCM tool). I prefer to use terminal for initial initiation of the repository. The steps and commands are very simple but before you execute them first close Xcode. Once your Xcode is closed, follow the commands below:

1. Browse to the project path where the .xcodeproj file is.
2. Execute the following from your command line:

```
git init
git add .
git commit –m ‘Committing from local to remote repository'
git remote add origin git@github.com:sanat-ikompass/ios.git
git pull origin master
```

3. This will open the VI editor so that you can enter any comment explaining your commit. Add a comment if you wish and save/close the editor. If you are not aware of VI editor and its commands, then the following snippet may help you:

    1. To enter in the add/insert mode hit : i
    2. Once in edit mode, you can add your text
    3. Once done exit the add/insert mode by hitting : Esc

4. To save your changes and exit the editor press : wq!

w = write

q = quit

! = force

4. Once the VI editor is gone, enter the following command:

```
git push origin master
```

That's all it takes. If you find git command's coming from alien land, worry not ! Git is very well documented and as stated earlier, you should read more about them from their online documentation which is available at : https://help.github.com

Finally, open project in XCode and you should be able to see Source Control is configured for GitHub use.