**Github & VSCode:**

Why github ? – version control and sharing capabilities for various types of files.

1. Make sure you install git before proceeding further: <https://git-scm.com/downloads>
2. create your free account on github: <https://github.com/> and create a test repository with the ***custom name.*** Copy the link of the repository and **make a note of it** and we will use it in further steps.

Graphical user interface, text, application

Description automatically generated

if it is already installed you should be able to see it when you search for git from the start menu.

1. make sure git is enabled on your vscode by navigating to settings and the following is checked as shown below.

Graphical user interface, text

Description automatically generated

1. Open an existing project folder and drag it to the VSCode

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, text

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface

Description automatically generated

U – for untracked

Graphical user interface, application

Description automatically generated

We have to go through a 2 step process : 1) commit 2) push

First commit is init and we will click on the check and now the files are ***committed***

Graphical user interface, application

Description automatically generated

* Next step is to ***push*** the changes to github. If we try to do it as shown below we will get the error ***“your repository has no remotes to push to”***

A screenshot of a computer

Description automatically generated with medium confidence

* Now let’s add the repository by navigating on VSCode : View🡪 command pallete🡪 Git: add remote (search for it if you don’t find it straight away) 🡪 provide a name, it is recommended to give the same name so that we will have an idea of what we are dealing with 🡪 Now we will be asked to enter the Remote URL for the repository. Now let’s use the url of our github repository which we made a note of it earlier in step 2 and enter it here.
* Now we should be able to push the changes with the help of Push option provided earlier or we can use the highlighted icon to push the changes to our github repository.

Graphical user interface, application

Description automatically generated

* Now let’s go to our github repository and we should see our files on the repo.

\*\* now let’s say we make some changes to any file on VSCode which is associated with a repository, for example make a small change to a file and save the changes.

* We need to go to the source control menu 🡪 commit the changes by providing a name for the commit and submit.
* Now if we open github if we open the github and the file we should see the changes reflect on the github file.

\*\*\*\*IMPORTANT NOTE:

Git will create a hidden file as shown below in your local folder, DO NOT delete it and DO NOT make changes to it.

Graphical user interface, text, application, email

Description automatically generated

If you want to know how to work with github + PyCharm I recommend making use of this tutorial: <https://www.youtube.com/watch?v=UBLONzkmReE&list=PLsyeobzWxl7poL9JTVyndKe62ieoN-MZ3&index=80>

**Python GitHub code contribution**

Now we will see how to collaboratively work on the code and upload them to the same github repository. EX: we have a team of 3 developers and we will allow them to upload the code to a single repository. Let’s see how.

Imagine bunny is our principal developer and working on his code using and IDE pycharm/VSCode and pushing code to his github repository**(let’s think of it as python\_for\_beginners repository)**.

Now another developer Rishi wants to contribute the code by pushing to bunny’s repository. Rishi uses his own machine and have his own github account.

Steps are:

First Rishi from his github account has to search for bunny’s github as shown below and navigate to the target repository.

Graphical user interface, application

Description automatically generated

Inside bunny’s account there will be multiple repositories and let’s say Rishi wants to upload his code to demopygit repository as shown below. Click on the repository and proceed further as mentioned in the further steps.

Graphical user interface, text, application, email

Description automatically generated

If Rishi want to provide some inputs Rishi needs to click on ***Fork.*** This will create a branch for us after the branch selection.

Graphical user interface, text, application

Description automatically generated

Let’s say as of now bunny has added only two lines as highlighted below and rishi wants to update it or add some more lines to the existing code. To do so Rishi needs to ***pull*** it first from his IDE(Pycharm/VSCode)

Graphical user interface, application, email

Description automatically generated

To pull something from github follow the below given steps.

Graphical user interface, application

Description automatically generated

Copy the Rishi’s github repo’s URL and paste it, further ***login to GitHub*** and ***test the connection*** and ***Clone.*** This will create a directory of Rishi on his machine.

Graphical user interface, website

Description automatically generated

We can see that in Pycharm the code which bunny has created appears and now Rishi can start making changes to it.

Graphical user interface, text, application, website

Description automatically generated

Now Rishi added something as shown below and it should get updated on bunny’s github repo. In other words, rishi want to push this to bunny’s github.

Graphical user interface, text, application

Description automatically generated

1. Perform the local commit which will save the code locally. (**right click on directory🡪 Git🡪 Commit Directory)**
2. Let’s push this to repository as shown below. Options are different for various IDE’s. Here we are looking at PyCharm.

Text

Description automatically generated

* Now we can notice that the changes are done to Rishi’s repo as shown below.

Graphical user interface, application, Teams

Description automatically generated

* Still we haven’t updated or pushed it to Bunny’s repo. So far the changes are only performed on Rishi’s github.
* *Now the question is HOW will bunny’s repo will receive these changes?* Whenever Rishi wants to contribute, what he will do is navigate to his **own repository** and **create a new pull request**

Graphical user interface, text, application, email

Description automatically generated

* After creating the new pull request this will go to bunny’s account which will create a base fork highlighting the changes as shown below. Create a pull request and submit it, this request will be sent to bunny’s repo.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* If we look at bunny’s repository and click on pull requests, we can see a new request has appeared and submitted earlier by Rishi.

Graphical user interface, text, application, website

Description automatically generated

* We can have multiple contributors along with Rishi. Now click on the request verify what the commit is about and and select ***merge pull request*** and ***confirm merge***.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* Now if we take a look at bunny’s file it will be updated with the changes made by Rishi.

Graphical user interface, text, application, email

Description automatically generated

* Now if we take a look at Bunny’s IDE and we can see only 2 lines of code that he has written and the changes made by rishi are not updated to his local. In order to update the code on his IDE, bunny has to perform ***pull*** as highlighted below.

Graphical user interface, text, application

Description automatically generated