**Git Hub**

By understanding how to manage different versions of project files, team members can collaborate more efficiently and avoid conflicts.

1. Learning how to **create repositories** and **switch between file states in Git** allows for better organisation and tracking of changes throughout the project's development.
2. Understanding how to utilise **Git's branching feature** enables developers to experiment with new ideas without affecting the main project, providing a safe space for innovation and exploration.
3. Git provides a version control system that allows team members to **easily revert back to previous versions** of files if needed, ensuring that no important changes are lost or overwritten.

|  |
| --- |
| Lab Task – 1 **Creating repository in Git Hub** |

|  |  |
| --- | --- |
| a. | Create an account in Git Hub and login into Git Hub |
|  | <https://github.com/login> |
| b. | Create a new repository for your project in Git Hub |
|  | Create a profile picture and chose your repositories    Click new  button and give repository name as “TestProj”  [Note: If name not available, give other suitable name for the repo]    Then click create repository button |

|  |
| --- |
| Lab Task – 2 **Working with Git** |

Open the windows power shell(or command prompt) and type the following commands

|  |  |
| --- | --- |
| a. | Check whether git is installed |
|  | Note: if it is not installed, you will get message like in the picture |
| b. | Download git if it is not installed |
|  | Download Git <https://git-scm.com/downloads>  Install the git from the downloaded file with default options |
| c. | Now Check whether git is installed using power shell |
|  |  |
| d | Check git version |
|  |  |
| e. | Configure git with username and password |
|  | git config --global user.name "username"  git config --global user.email "emailid@xyz.com"  git config --global user.password "password"    Note: username, pwd and email of GitHub |
| f. | Goto Git Hub and copy the URL of your repop |
|  |  |
| h. | Clone git hub repo to Git |
|  | Create a folder called HCT using windows or power shell. In this folder, you are going to bring the repo of Git Hub  Note: the folder should have read and write permission when you clone a repo. Right click the folder—properties---general---uncheck the read-only.    Write the following command in power shel to clone. |
| i. | After cloning, Check the content of repo in the Git |
|  | Note: By Default, a folder will be present in the repo name |
| j. | Create a python file in the Testproj folder |
|  | Note: Save and Download the file in Testproj folder and check it in windows    Note: we have save the file in local repo only in the Git |
| k. | Check the add file in the local repo |
|  |  |
| l. | Convert the add file for tracking by using add command and check the status |
|  | Note: The file is not yet commited |
| m | Save the file to the local repo |
|  |  |
|  | Check the status of the file in the loca repo |
|  |  |
|  | Push the add file from local repo(Git) to remote repo(Git Hub) |
|  |  |
|  | Now check it in Git Hub |
|  |  |
|  | Consider this Testproj in Git Hub is accessed by your project team members and they edit the file as shown below in the Git Hub. |
|  | Open the add.ipynb from the colab through Git Hub repo    Note : Choose add.ipynb to add to the colab |
|  | Modify the code as a=1 to a=10 |
|  |  |
|  | Save the file in the GitHub |
|  | In the colab, file—save a copy in Git Hub |
|  | Check the add file in the local repo |
|  | Since it is a Jupiter file and it will come in different format. Just check the value of a in source    You can see still, a=1 |
|  | Pull the updated files from the remote repo(Git Hub) to local repo(Git) |
|  |  |
|  | Now check the local repo whether files are updated from remote repo |
|  | So, you see the local repo is updated since a=10 |
|  | Suppose you have change the a value to15 and some other person has change the a value to 22 in the remote repo. In that case, if you perform git pull, then merge conflict occurs. |
|  | **Change the a=15 in the loca repo**   1. Open the file      1. Change the a to 15 and save it      1. Commit it to the local repo     Note: use – am options  **Modify a=22 in Git Hub**   1. Open add.ipynb in colab using Git Hub and modify a=22      1. Save it in the Git Hub      1. Check it in the Git Hub |
|  | **Now pull the file from remote repo to local repo** |
|  | **Merge conflicts occurs because local repo contains a=15 but remote repo contains a=22** |
|  | Open add file from the loca repo |
|  |  |
|  | Solve the merge conflict |
|  | You can keep a=15 or a=22 or assign any new value to a. So, I’m making a=20.  Remove the other tags and update as shown in the figure    Save the file |
|  | Save to the local repo |
|  |  |
|  | Push the local repo to remote repo |
|  |  |
|  | Confirm whether a=20 in the remote repo(Git Hub) |
|  |  |
|  | Perform Branching |
|  | Create a branch to modify a+b to a\*b in add.ipynb |
|  | Check whether any branch exists |
|  |  |
|  | Create a new branch called “NewBranch” and find the status |
|  | * Indicates Newbranch is in active stage |
|  | Modify the content a+b to a\*b |
|  | Save the file |
|  | Commit the file to loca repo |
|  |  |