

GitHub

What is Git:

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. It manages source code history.

What is GitHub:

GitHub is a Git repository **hosting service**, but it adds many of its own features. While Git is a command line tool, GitHub provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project.

Service Providers Similar to GitHub:

- BitBucket
- SourceForge

Steps to be followed to work using GitHub:

- Install git on local machine (windows or Ubuntu).
- Link to download git: <https://git-scm.com/downloads>
- After Installing git on local machine, run it and open **Git Bash**.

Git Commands:

Command	Action Performed
git --version	Gives version of git on local machine
git help config/ git config --help	Opens config available for git in browser
mkdir test	Creates a directory with name test
cd test	Sets the working directory to test
Manually create a txt/any file here	
git init	Initializes git
git status	Gives the status of the file updated or not
git config - global user. username "sri1108"	Giving permission to access GitHub account
Manually create a repository in GitHub and copy the https address	
git remote add origin address	Creates the connection from local to GitHub
git add ./file.txt/ any file	Add the file which is needed to put on GitHub
git commit -m "committing the file"	Creates commit for the file on GitHub
git push origin master	Pushes the committed file into GitHub

- Whenever we need to **make any changes** and upload on GitHub simple make changes for the file on local and then run these commands on git bash:
- `git add ./file.txt/ any file`
- `git status`
- `git commit -m "committing this file"/ git commit -a -m "committing all files"`
- `git push origin master`

Git Pull Requests:

- when we need to pull request from certain repository of other user, we need to fork the repository into our account and then clone it to local machine by copying SSH address from GitHub forked repository and create a branch (Other than master branch) and make changes under that branch then add, commit and push the file into GitHub which creates a pull request for the owner of the original repository.
- Accordingly, if the owner wants to merge pull request, he can do it on GitHub directly.

Command	Action Performed
<code>git clone SSH Address</code>	Creates the clone on local machine
<code>git branch</code>	Gives the list of available branches
<code>git branch TBB</code>	Creates a branch with name TBB
<code>git checkout TBB</code>	Changes current working branch to TBB
<code>git checkout -b TBB</code>	Creates branch & makes current branch
<code>git status</code>	Status of file updated or not
<code>git diff 1...2</code>	Gives difference between 1 and 2 files
<code>git add file</code>	Add the file which needs to put on GitHub
<code>git commit -m "commit the file"</code>	Commits the changes to the file
While adding it as branch, we need to change the default branch to master only while pushing we need to push origin TBB (branch)	
<code>git checkout master</code>	Back to master branch
<code>git push origin TBB</code>	Pushes the branch content TBB

Git Branch Create, Merge, Delete:

- Create Branch and add it to github account.

Command	Action
<code>git branch TBB</code>	Creating new Branch in local
<code>git checkout TBB</code>	Assigns TBB as current working branch
<code>git add file</code>	Adds all files to TBB branch

git commit -a -m "commit all"	All updates
git push - - set - - upstream origin TBB	Pushes TBB branch into GitHub
Merging of 2 branches	
git checkout master	Set Master as Active Branch
git merge TBB	Merges TBB to master and combines
Deletion of Branch	
git checkout -b dummy	Creates a dummy branch and makes active
git checkout master	Makes master as active branch
git branch -d dummy	Deletes dummy branch

Security Provided by Github :

- GitHub repositories are of 2 types: Public, Private.
- While Building a Repository, GitHub asks whether you want to make it public or private.
- If we chose to make it public, it will be accessible to everyone and we can find it even by using google search (Not secure).
- If we chose to make it private, it will be accessible only by the single owner and no one can find that repository anywhere.
- If we want to make code accessible to everyone in the team then it's a paid service. we need to set up an organization and then create a team and add that repository there.
- If it is required to share the code between only 3 persons (at max) then we can add them as collaborators to our repository and it is free of cost. Only those 3 persons can access your repository if you make it private.

Cost:

- Costing of GitHub to create and manage organization where we can create teams, assign tasks, see the progress of each task, get unlimited private repositories is \$9 per user per month.
- Cost for the same service from BitBucket is 2\$ per month per user but BitBucket is giving this service free for small teams up to 5 users. But maximum file storage is 1 GB and Build minutes are 50 minutes/ month.