Thursday
5th June, 2025

Q How to push or commit code to Github using VS code?

1. Make sure Git is installed in pc.
2. Initialize Git
   git init
3. Connect to github repo
   git remote add origin <repo-url>
4. Stage and commit code
   git add .
   git commit -m "msg"
5. Set the branch and push
   git branch -M main
   git push -u origin main

Q I have 2 Github accounts. VS code is signed in with my first (old) account, but now I want to push code to a repository in my second Github account. However, VS code is only showing the first account.

1. First check in terminal:
   git init (if not)
   git config user.name
   git config user.email
2. Remove old Github credentials:
   Credential Manager > Windows Credential
   - Find and remove any entries like:
   git:https://github.com or Github
3. (Optional) Clear Git user config: in terminal
   git config --global --unset user.name
   git config --global --unset user.email

4. Set new Github account credentials:
   git config --global user.name "username"
   git config --global user.email "your.email"
5. Connect to the repo to second github acc's repo:
   git remote remove origin
   git remote add origin <repo.url>
6. Push code
   git add .
   git commit -m "msg"
   git branch -M main (if not already)
   git push -u origin main
7. Github will now prompt for login / Token.

Q What if i don't use --global in Github config?
so, the Git setting (like username & email) will apply only to the current project (local), not system-wide.

This is safer when using multiple Github accounts,

Q What does -M main when setting a Git branch?
   git branch -M main
   "move or rename the branch"
even if the branch name already exist
- -M is a force rename
- used when you want to rename current branch

Thursday
5th June, 2025

Q What is the meaning of -u in git push? & what is origin?

-u (--set-upstream) in git push -u origin main

- It links our local branch to the remote branch, so next time we can just use git push or git pull without specifying branch and origin again.

Ex git push -u origin main

"Push my code from my local main branch to the main branch on the GitHub repo I called origin, & remember this connection."

Q What is origin?

When we want connect our local project to a GitHub repo, we run:

git remote add origin <repo-url>

Here, origin is just a nickname for that GitHub URL. We can set any name.

So instead of typing the full URL every time, we can just write:

git push origin main

Think of origin as:

"Hey Git, push my code to that GitHub repo, I called origin.

Q What does git add . do?

This ~~chain~~ stages all changed files (new, modified, deleted) in the current folder and subfolders, preparing them for the next commit.

Q How to clone a repo?

To clone a repo means to download a full copy of a project from GitHub to your local system.

1. Go to GitHub repo page
2. Click "Code"
3. Copy HTTPS URL
4. Open terminal & run
5. · git clone <url>
5. A folder with the repo name will be created with all files inside.

Q Basic Git commands

1. git init
Initializes a new GitHub repo in our current folder
Makes it a Git-Tracked project

2. git clone <repo-url>

3. git status
shows current changes, staged files, and untracked files.

4. git add.
Stages all changes (new or edited files) for commit.
Prepares files to be commited.

Thursday
5th June, 2025

5. git commit -m "message"
Saves the staged changes with a message.

6. git remote add origin <repo-url>
Connects our local repo to a GitHub repo, with the name origin.

7. git push -u origin main
Pushes our code to GitHub and sets a link between local main and remote main.

8. git push
Pushes latest commits to GitHub

9. git pull
Pulls (downloads) the latest code from GitHub to your local repo.

10. git branch
Lists all branches in our project.
The current one is highlighted.

11. git branch <name>
Creates a new branch

12. git checkout <branch-name>
Switches to another branch.

13. git merge <branch>
Merges changes from another branch into the current branch.

14. git log
Shows a list of past commits with messages, authors, and time.

15. git config --global user.name "name"
git config --global user.email "email"
To set Git username & Git email globally.

Q What is the meaning of merging branches in Git?
Mergin branches means combining the changes from one branch into another.

Explination:
Suppose you're working on a website:
- You have the main code in the main branch
- You create a new branch called contact-form to build a contact form feature:
git checkout -b contact-form

Now you add code, test it, and commit it.
Once done, you want to bring these changes into the main branch.
git checkout main # Go to main branch
git merge contact-form # merge the contact form branch into main

Now the main branch will have the contact form feature too.

Thursday
5th June, 2025

Real-life use:
Merging helps you work on features separately without disturbing the main project.
Once your work is ready and tested, you merge it back with to the main code.

Q. What is the meaning of -b in Git?
used with git checkout or git switch to create a new branch and switch to it immediately.

Ex git checkout -b new-feature
Create a new branch called new-feature and move me into it.

Q Correct Git Command sequence for new project:

1. Initialize Git in the local folder.
   git init
2. Add the GitHub repo as remote and name it origin:
   git remote add origin <repo-url>
3. Rename current branch to main
   git branch -M main
4. Stage all files for the first commit
   git add . or git add <specific-file>
5. Commit the staged changes with a msg:
   git commit -m "msg"
6. Push to GitHub and set tracking between local main and remote main:
   git push -u origin main

After this setup, next we only need to use:
git add .
git commit -m "msg"
git push