Laboratory Activity 2 – Using Git in Visual Studio Code

Objectives:

By the end of this lab, students should be able to:
1. Initialize a Git repository inside VS Code.
2. Stage and commit changes using the Source Control panel.
3. Connect a local repository to GitHub.
4. Push and pull changes between local and remote repositories.
5. Create and switch Git branches inside VS Code.

Background:

Version control is essential in modern software development. Git is the most widely used distributed version control system, and GitHub is a cloud-based platform that allows collaboration, remote repositories, and project management. Visual Studio Code (VS Code) integrates Git, so students can manage version control without leaving the editor.

Requirements:

- Installed Git (https://git-scm.com)
- Installed Visual Studio Code (https://code.visualstudio.com)
- Active GitHub account
- Internet connection

Procedure (Step-by-Step):

Step 0: Access VS Code Terminal

- Open your project folder in VS Code.
- Open terminal:
  - Shortcut: Ctrl + ` (backtick)
  - Or Menu: View → Terminal

Step 1: Initialize Git Repository

1. Open the Source Control panel (branch icon on left sidebar).
2. Click Initialize Repository. (This runs git init in the background).
3. Observe that all files are now listed under Changes.

Step 2: Stage and Commit Changes

1. Modify or create a file, e.g. index.js:
   console.log("Hello Git from VS Code!");
2. In Source Control panel, hover over the file → click ✚ Stage Changes.
3. In the message box, type: Initial commit
4. Press ✓ Commit (or Ctrl + Enter).

Step 3: Connect to GitHub

1. Log in to GitHub.
2. Create a new empty repository (no README, no .gitignore).
3. Copy the HTTPS URL.

4. Back in VS Code Terminal:

```
git remote add origin https://github.com/username/my-first-repo.git
git branch -M main
git push -u origin main
```

Step 4: Push and Pull Changes

- To Push changes: After committing, click the ↑ (Push) icon in the bottom-left of VS Code.
- To Pull updates from GitHub: Click the ↓ (Pull) icon in the bottom-left.

Step 5: Create and Switch Branches

1. Look at the bottom-left corner → click the branch name (default: main).
2. Select Create new branch → name it feature-login.
3. Make changes, commit, and push branch:
   git push -u origin feature-login
4. Switch back to main branch from the same menu.

Step 6: Verify on GitHub

- Refresh your GitHub repo page → confirm that your files and commits are uploaded.
- Check branches tab to see both main and feature-login.

Expected Output:

- Local repository created inside VS Code.
- Initial commit pushed to GitHub repository.
- Students can successfully push, pull, and branch within VS Code.
- **Students should provide screenshots of:**
    1. Initializing repository in VS Code
    2. Staging and committing changes
    3. GitHub repository showing uploaded files
    4. Branch creation and switching process

Questions (for Lab Report):

1. What is the importance of version control in collaborative projects?
2. What is the difference between staging and committing?
3. What command (or button in VS Code) is used to send local commits to GitHub?
4. How does branching help in managing multiple features in a project?
5. What common errors did you encounter (e.g., refspec error) and how did you solve them?