# a. Create a Git repository, initialize it, and add a Python project

**Step 1: Create a Project Directory** *mkdir git-prac3 cd git-prac3*

**Step 2: Create a Sample Python Project** *echo "print('Hello from main branch')" > app.py*

**Step 3: Initialize Git**

*git init*

**Step 4: Add Files and Commit**

*git add .*

*git commit -m "Initial commit with basic app.py"*

**Step 5: Create a Remote Repo on GitHub**

* Go to [https://github.com](https://github.com/)
* Click New Repository → Name: git-prac3
* **DO NOT** initialize with README

**Step 6: Add Remote and Push(in VSCode)** *git remote add origin https://github.com/YOUR\_USERNAME/git-prac3.git git branch -M main git push -u origin main*

|  |
| --- |
|  |
|  |

**b. Demonstrate branching and pull requests**

**Step 1: Create a Feature Branch** *git checkout -b feature/greet-user*  **Step 2: Add New Code**

Edit app.py:

*name = input("Enter your name: ")*

*print(f"Hello, {name}!")*

*Then in Powershell:*

*git add app.py git commit -m "Add user greeting feature" git push origin feature/greet-user*

**Step 3: Create a Pull Request**

1. Go to your repo on GitHub
2. You’ll see a prompt: **“Compare & pull request”**
3. Click it → Add a description → Click **“Create pull request”**
4. Review and click **“Merge pull request”**
5. Confirm merge and delete branch (optional)

**c. Simulate a conflict and resolve it**

**Step 1: Create Two Feature Branches from Main** *git checkout main git checkout -b feature/conflict-A*

Change app.py:

*print("This is branch A")*

Then:

*git add app.py git commit -m "Change message in branch A" git push origin feature/conflict-A*

**Optional:**

Now create another conflicting branch:

*git checkout main git checkout -b feature/conflict-B*

Edit the **same line** in app.py:

*print("This is branch B")*

Then:

*git add app.py git commit -m "Change message in branch B" git push origin feature/conflict-B*

*OpOptional Close\_\_\_\_\_\_\_\_\_\_\_\_\_*

**Step 2: Merge One Branch First (e.g., conflict-A)**

* On GitHub → Create PR from feature/conflict-A → Merge into main

**Step 3: Create PR from conflict-B**

* On GitHub → Create PR from feature/conflict-B
* GitHub will **detect a conflict**
* Click **“Resolve conflict”**
* Manually fix the conflict. Example:

*<<<<<<< HEAD*

*print("This is branch A")*

*=======*

*print("This is branch B")*

*>>>>>>> feature/conflict-B*

Edit it to:

*print("This is the resolved version")*

Click **“Mark as resolved”** → Commit → Merge the pull request