

✅ QUESTION 1 — Git Mistakes & Recovery Challenge

(Fixing Real-World Git Problems)

Simulate several common Git mistakes and fix them using correct Git commands.

Task Requirements

1. Create a new Git repository.
 2. Create a file named notes.txt and make an initial commit.
 3. Make a **wrong commit** (any incorrect change).
 4. Fix that commit using:
 5. `git commit --amend`
 6. Create **two more commits** with any changes you like.
 7. Now assume the **second commit** contains incorrect work.
Undo it using:
 8. `git revert <commit-hash>`
 9. Create another mistake:
 - Delete a file
 - Stage the deletion
 - Commit it
 10. Realize it was incorrect. Restore the file using one of the following:
 11. `git reset --soft HEAD~1`
- or
- `git restore <filename>`
12. Push the repository to GitHub.
 13. Add your teammates as collaborators.

✅ QUESTION 2 — Git Versioning & Release Tags

Simulate the versioning workflow for a small project with multiple files.

Task Requirements

1. Start a new Git repository.
 2. Add **five files** to the project and make the first commit (Version 1).
 3. Create **Version 2**:
 - Modify two files
 - Add one new file
 - Remove one file
 - Commit these changes
 4. Create **Version 3**:
 - Rename one file
 - Make additional changes to other files
 - Commit again
 5. Tag each version:
 6. v1.0
 7. v2.0
 8. v3.0
 9. Push everything to GitHub.
 10. Add your teammates as collaborators.
-

✅ **QUESTION 3 — Git Workflow Simulation (Git Flow Model)**

(Feature Branching + Release Flow)

Follow a simplified version of the Git Flow branching model.

Task Requirements

1. Create a GitHub repository and clone it locally.
2. Create a branch named:
3. develop
4. Create **three feature branches**:
 - feature/login

- feature/register
 - feature/profile
5. In each feature branch:
 - Add a unique file
 - Make at least one commit
 6. Merge each feature branch into develop.
 7. Create a release branch:
 8. release/v1.0
 9. Inside the release branch:
 - Add a file named VERSION.txt
 - Write "Version 1.0" inside it
 - Commit the change
 10. Merge the release branch into main.
 11. Tag the release commit as:
 12. v1.0
 13. Push all branches and tags to GitHub.
 14. Add your teammates as collaborators.