**Practice Sheet-GIT and GITHUB**

**Repository Setup**

* Create a new Git repository named sample-project on your local machine.
* Initialize the repository using the git init command.
* Add a README.md file to the repository with the content:  
  # Sample Project: Git Assignment

 **Version Control Basics**

* Stage and commit the README.md file with the message:  
  "Initial commit: Added README.md"

 **GitHub Repository Creation**

* Create a public GitHub repository named sample-project.
* Link your local repository to the remote GitHub repository using:  
  git remote add origin <repository-url>
* Push the local repository to GitHub using:  
  git push -u origin main

 **Branching and Feature Development**

* Create a new branch named feature-1 using:  
  git checkout -b feature-1
* Add a new file named feature.txt to the branch with the content:  
  "This is a new feature."
* Stage and commit the new file with the message:  
  "Added feature.txt to feature-1"
* Push the feature-1 branch to GitHub.

 **Merging Changes**

* On GitHub, create a pull request to merge the feature-1 branch into main.
* Merge the pull request and verify that feature.txt is now part of the main branch.

 **Collaboration Simulation**

* Collaborate with a peer or create a second GitHub account.
* Fork the original sample-project repository and clone it.
* Make a change to README.md (e.g., add a new section).
* Commit the change and submit a pull request to the original repository.

Question 2

 **Repository Setup**

* Create a Git repository named conflict-practice.
* Add a file named story.txt with the content:  
  "Once upon a time, there was a developer."
* Stage and commit the file with the message:  
  "Added initial story"

 **Branching and Modifying the Same File:**

* Create a branch named branch-A.
* On branch-A, modify story.txt to:  
  "Once upon a time, there was a curious developer learning Git."
* Commit the change with the message:  
  "Updated story on branch-A"
* Switch to the main branch and modify story.txt to:  
  "Once upon a time, there was a skilled developer working on Git projects."
* Commit the change with the message:  
  "Updated story on main"

 **Merge and Handle Conflict:**

* Attempt to merge branch-A into main.

git merge branch-A

* A conflict will be detected. Open story.txt and resolve the conflict by keeping both changes (combine the sentences).

Once upon a time, there was a curious developer learning Git and working on Git projects.

* Stage the resolved file and commit the merge with the message:  
  "Resolved merge conflict in story.txt"

 **History Exploration:**

* Use the following Git commands to inspect the repository:
  + Show all commits: git log --oneline
  + Show changes in story.txt: git log -p story.txt
  + Show file changes since the initial commit: git diff HEAD~2

 **Reverting a Commit (Optional):**

* Create a new branch named undo-branch.
* Revert the commit that updated the story on the main before the merge.

git revert <commit-hash>

* Push the changes to GitHub.