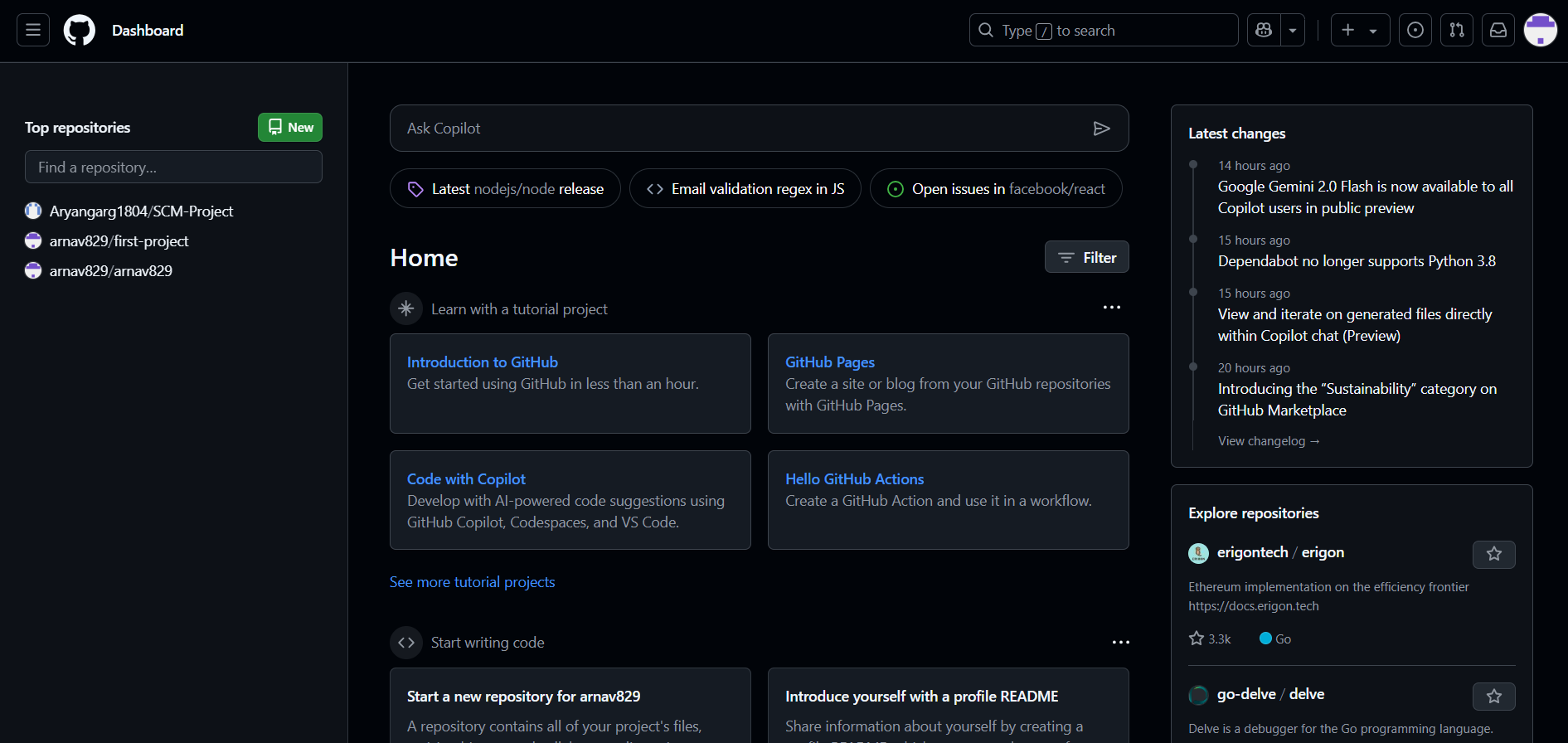
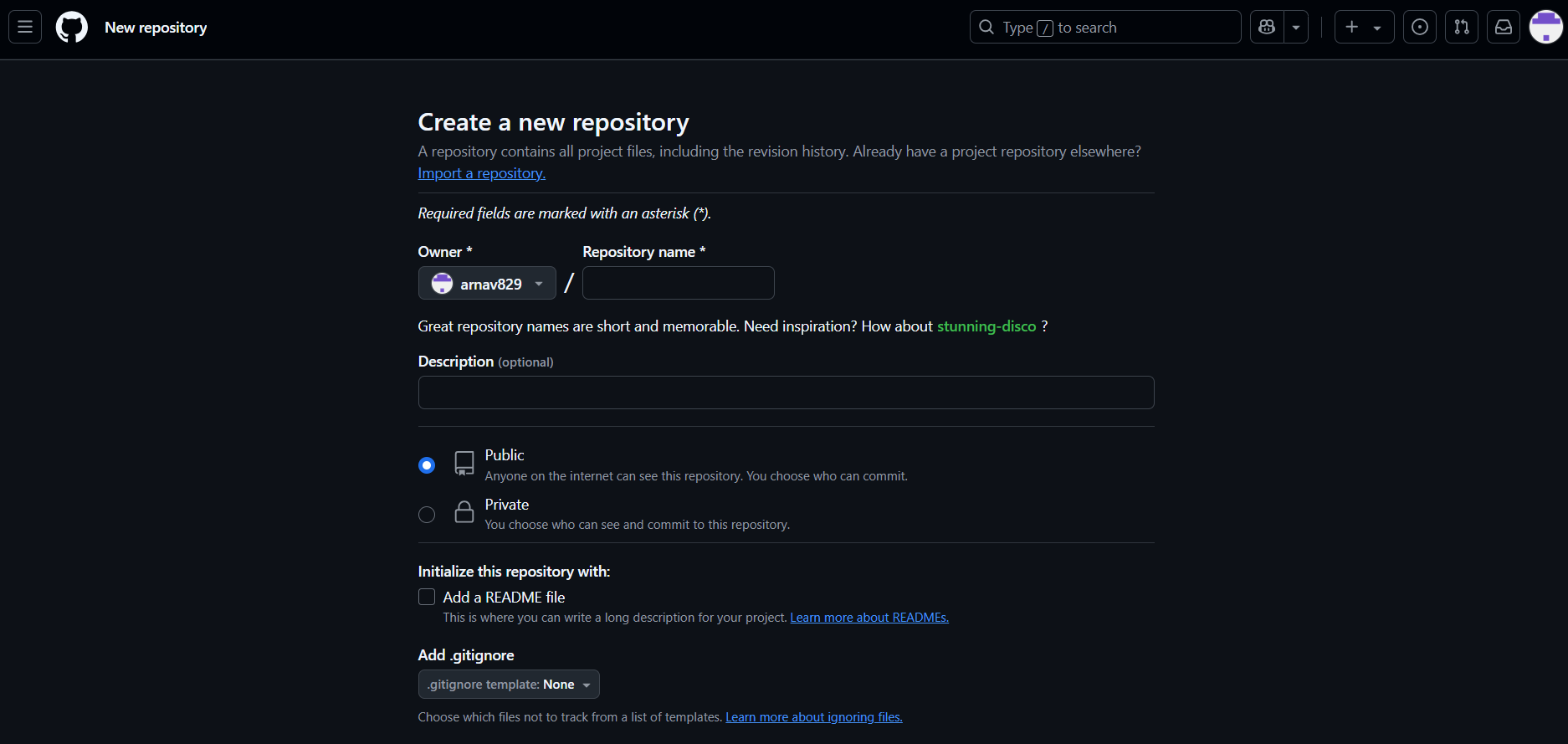
# SCM Sample Assignment: Implementing Git Basics

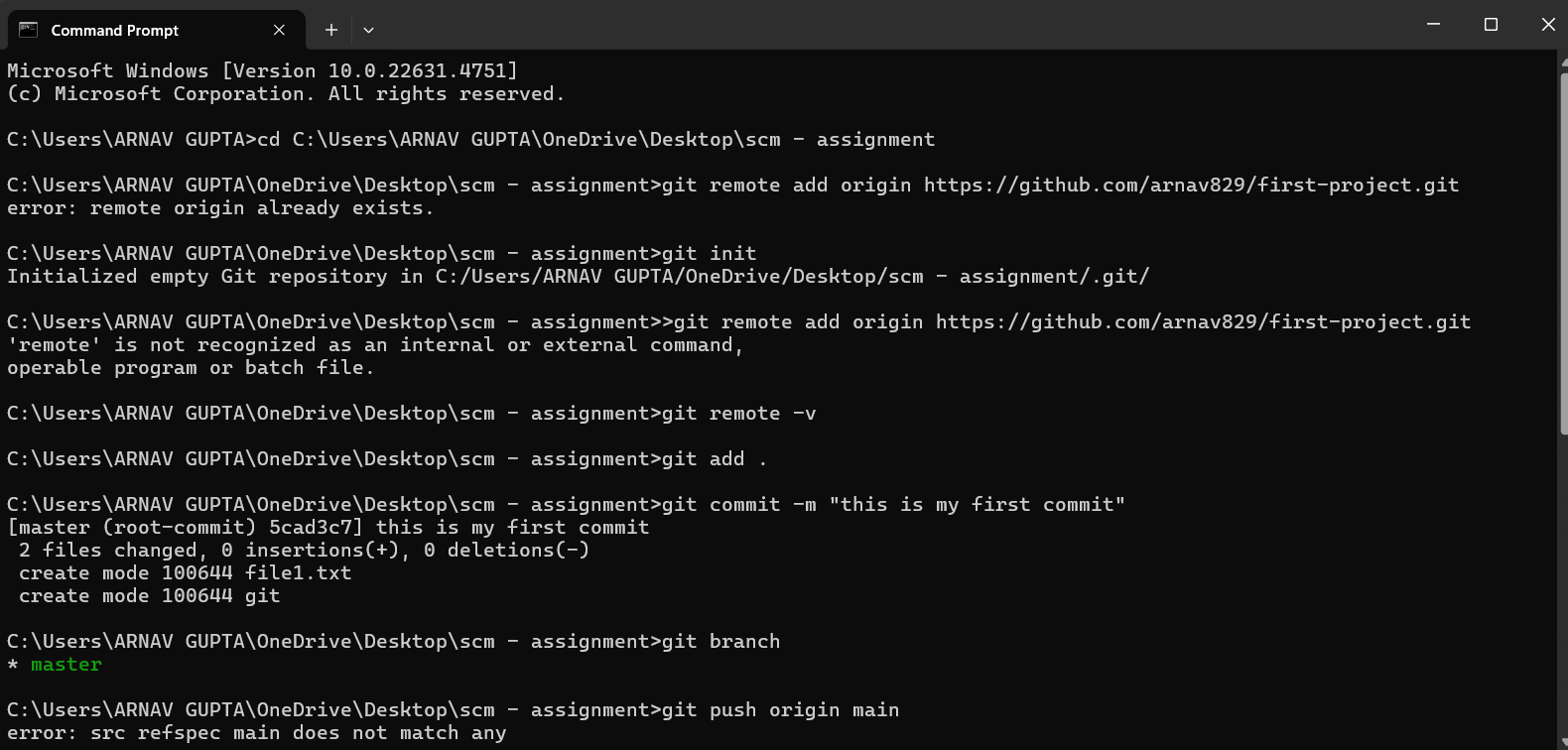
# 1. Repository Setup and Initial Commit

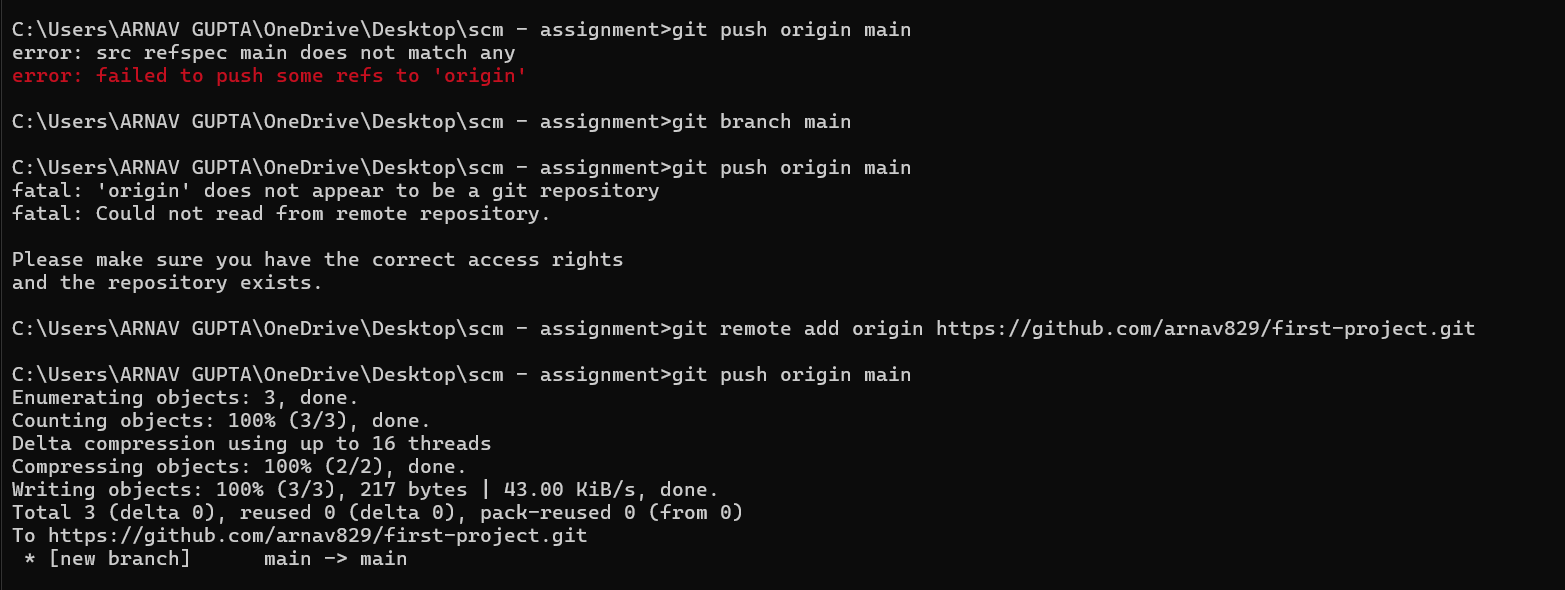
Steps: 1. Create a GitHub Repository:  
 - Navigate to GitHub and log in.  
 - Click on the "+" icon and select "New Repository."  
 - Name the repository, add a description, and create it as a public or private repository.





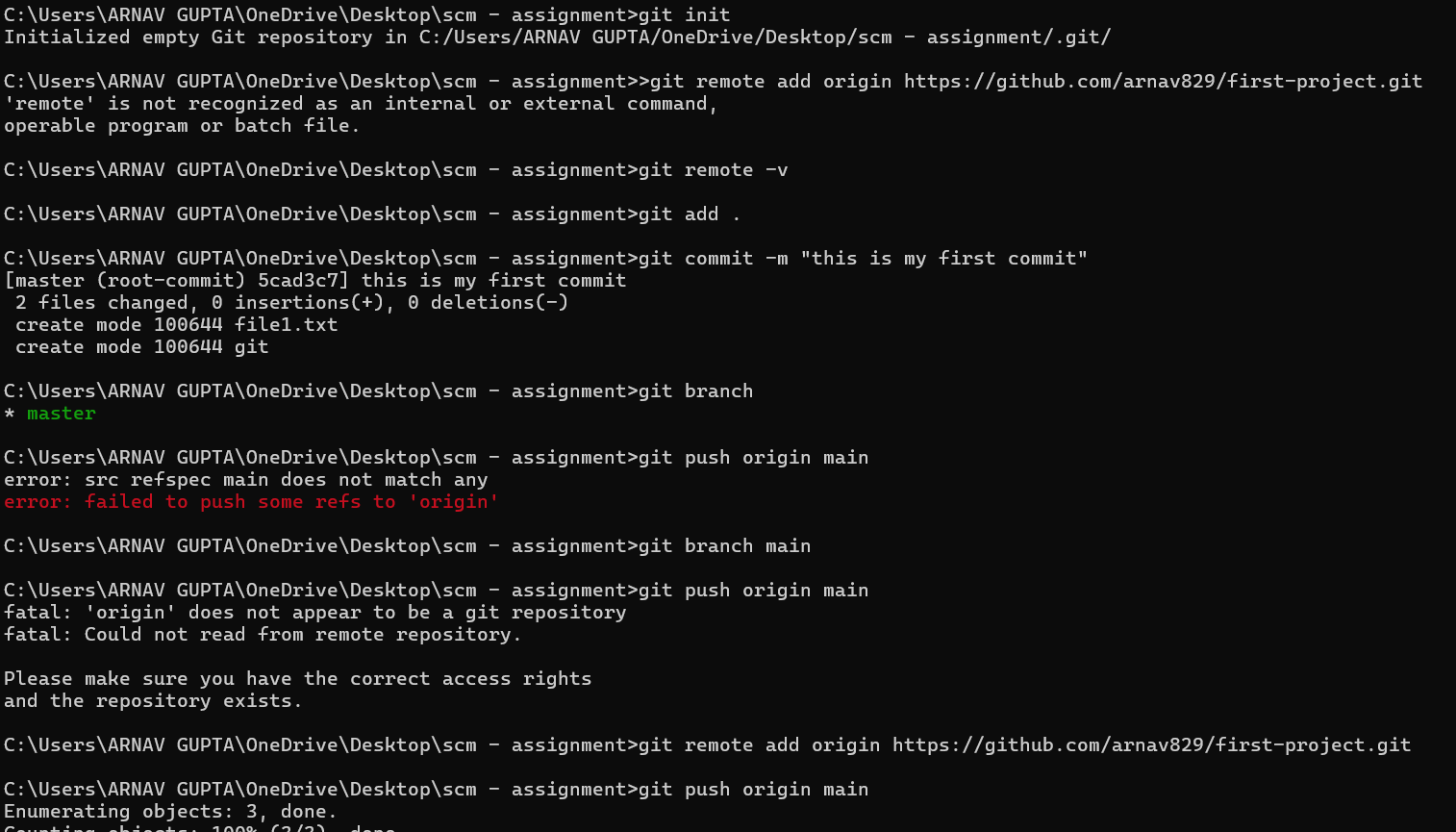
2. Clone the Repository Locally:





3. Initialize and Create an Initial Commit:

echo "text file" > project\_overview.txt  
 git add project\_overview.txt  
 git commit -m "Initial commit "  
 git push origin main



## 2. Branch Creation and Updates

Steps:

1. Create a New Branch:

git checkout -b feature-update

2. Add a Feature Details File:

echo "Details of the new feature." > feature\_details.txt  
 git add feature\_details.txt  
 git commit -m "Added feature\_details.txt in feature-update branch"  
 git push origin feature-update

Screenshot: \*(Attach an image showing the feature-update branch and commit log.)\*

## 3. Merge and Conflict Resolution

Steps:

1. Modify the `feature\_details.txt` file in the `main` branch:

git checkout main  
 echo "Modified content in main branch." > feature\_details.txt  
 git commit -am "Updated feature\_details.txt in main branch"  
 git push origin main

2. Attempt to Merge feature-update into Main:

git checkout main  
 git merge feature-update

3. Resolve the Conflict:

- Open `feature\_details.txt`, manually edit the conflicting lines.  
 - Save the file, then run:  
 git add feature\_details.txt  
 git commit -m "Resolved merge conflict in feature\_details.txt"  
 git push origin main

Screenshot: \*(Attach an image showing the merge conflict resolution.)\*

## 4. Repository Audit

Commands and Outputs:

1. Check Repository Status:

git status

\*(Attach a screenshot of the status output.)\*

2. Review Commit History:

git log --oneline --graph

\*(Attach a screenshot showing the commit history, including branch creation, commits, merges, and conflict resolutions.) \*

This document outlines the key Git operations performed, including repository setup, branch management, merging, conflict resolution, and repository auditing. The included screenshots provide visual evidence of the steps executed.

2. Branch Creation and Updates

Steps:

1. Create a New Branch:

git branch feature-update

git checkout feature-update

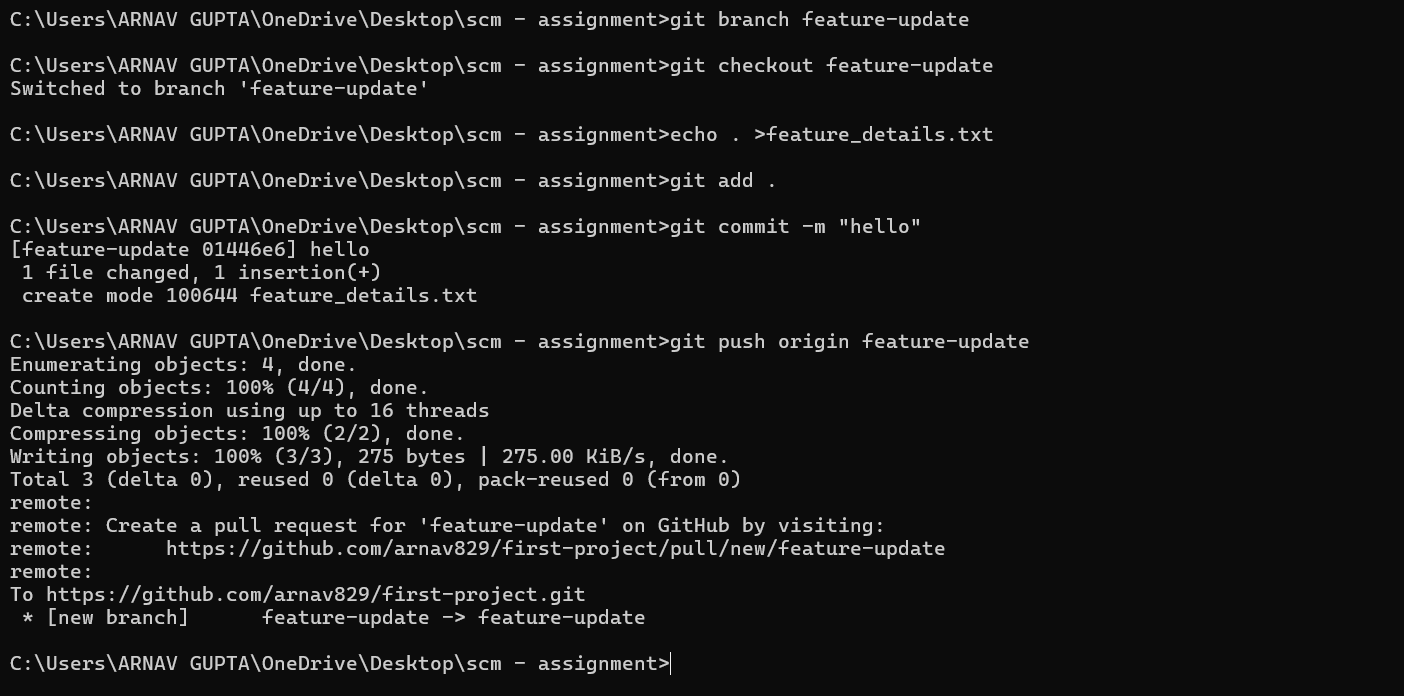
2. Add a Feature Details File:

echo. > feature\_details.txt

git add.

git commit -m "new features"

git push origin feature-update



3. Merge and Conflict Resolution

Steps:

1. Modify the feature\_details.txt file in the main branch:

git checkout main

echo "Modified content in main branch." > feature\_details.txt

git commit -am "Updated feature\_details.txt in main branch"

git push origin main

2. Attempt to Merge feature-update into Main:

git checkout main

git merge feature-update

3. Resolve the Conflict:

- Open feature\_details.txt, manually edit the conflicting lines.

- Save the file, then run:

git add feature\_details.txt

git commit -m "Resolved merge conflict in feature\_details.txt"

git push origin main

4. Repository Audit

Commands and Outputs:

1. Check Repository Status:

git status

(Attach a screenshot of the status output.)

2. Review Commit History:

git log --online –graph

