

Assignment 2

Vishwamber Reddy

2025-05-12

Objectives

This guide demonstrates the use of Git, GitHub, and the command-line interface for collaborative and reproducible practices, using a basic RStudio project setup.

1. Create a New RStudio Project

I created a new RStudio Project named 5513_Assignment2.

Inside the project folder, I created a file named `example.qmd`

Knit to HTML

I used the **Render** button in RStudio to knit `example.qmd` into HTML. The output file `example.html` was successfully generated and previewed.

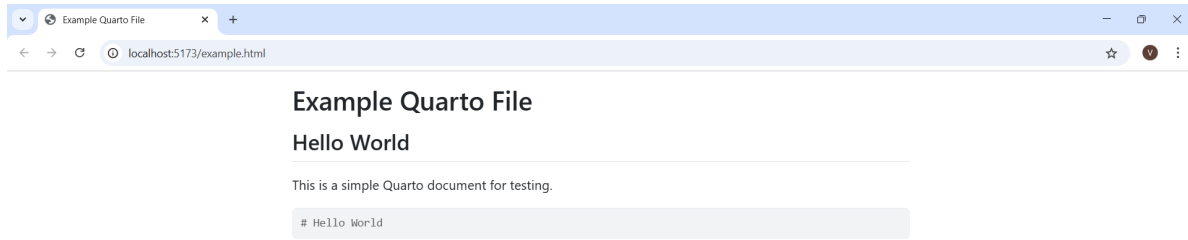


Figure 1: Knitted HTML Output

2. Initialise Git Repository and Push to GitHub

In the terminal:

```
cd 5513_Assignment2
git init
git add .
git commit -m "Initial commit with Quarto file"
```

Create GitHub Repository

I created a new repository on GitHub named assignment2-5513, then added the remote:

```
git remote add origin https://github.com/vishwamber7/assignment2-5513.git
git branch -M main
git push -u origin main
```

3. Create a Branch and Modify File

```
git checkout -b testbranch
```

I modified example.qmd by adding this line under the existing text:

“This line was added in testbranch.”

Then:

```
git add example.qmd
git commit -m "Modified example.qmd in testbranch"
git push -u origin testbranch
```

4. Add Data Folder and Amend Commit

I created a folder called data and added the CSV file from Assignment 1.

```
mkdir data
cp ../etc5513-assignment-1-vishwamber7Assignment1/modern-renewable-prod.csv data/
git add data/modern-renewable-prod.csv
git commit --amend --no-edit
git push --force
```

5. Conflict Setup in main Branch

Switching back to main:

```
git checkout main
```

I edited example.qmd differently by changing the same section modified in testbranch:

“This line was added in main branch and will conflict.”

Then:

```
git add example.qmd
git commit -m "Conflicting change in example.qmd in main"
git push
```

6. Merge Branch and Fix Conflict

```
git merge testbranch
```

This created a merge conflict in example.qmd. Git output:

Auto-merging example.qmd

CONFLICT (content): Merge conflict in example.qmd

Automatic merge failed; fix conflicts and then commit the result.

I opened example.qmd and resolved the conflict manually:

“This line includes content from both branches.”

Then:

```
git add example.qmd
git commit -m "Resolved merge conflict between main and testbranch"
git push
```

Conflict resolved and pushed to GitHub.

7. Tag Commit

```
git tag -a v1.0 -m "Version 1.0 release after merge"
git push origin v1.0
```

8. Delete Branch

```
git branch -d testbranch
git push origin --delete testbranch
```

testbranch deleted locally and remotely.

9. Condensed Git Log

```
git log --oneline
```

Output:

```
a70ec5e (HEAD -> main, tag: v1.0, origin/main) Resolved merge conflict between main and testbranch
a70d25b Conflicting change in example.qmd in main
f3dee0b Modified example.qmd in testbranch
5caaed0 Initial commit with Quarto file
```

10. Add Plot and Undo Commit

I added the following to example.qmd:



Figure 2: Example Plot Code

Then:

```
git add example.qmd
git commit -m "Added simple plot to example.qmd"
```

To undo the commit but keep changes:

```
git reset --soft HEAD~1
```

The commit was undone, but changes remained in the working directory.

Summary of Git Skills Demonstrated

- Git init, add, commit
- Pushing to GitHub
- Branching and merging
- Conflict resolution
- Amending commits
- Using tags
- Deleting branches
- Condensed git logs
- Undoing commits with reset

Commit Messages

Commit messages were concise and descriptive:

- Initial commit with Quarto file
- Modified example.qmd in testbranch
- Conflicting change in example.qmd in main
- Resolved merge conflict between main and testbranch
- Added simple plot to example.qmd

Conclusion

This guide demonstrates essential collaborative Git practices, including conflict resolution, branching, tagging, and commit management, ensuring reproducibility and clean version history in data science projects.

AI acknowledgement

I have used AI to correct grammatical errors made in the file.