

# Peer Code Review Instructions

Ellen Bledsoe

2025-12-02

## Assignment Overview

For this assignment, you will conduct a formal **peer code review** of another student's final project hosted on GitHub. You will assess code for clarity, functionality, organization, and reproducibility, and provide **constructive, actionable feedback**.

The majority of your code review will occur on GitHub (follow the instructions below). In addition, you will submit a knitted PDF file with a completed checklist and reflection to D2L.

**Total Points:** 100

**Due Date:** Tuesday, Dec 9 at midnight MST

---

## Assignment Instructions

### 1. Access Your Assigned Peer's GitHub Repository (5 points)

- Go to your peer's GitHub repository.
- Fork and clone the repository to your computer (see detailed instructions below).
- Review the README and folder structure before diving into code.

### 2. Review Code and Submit Comments via GitHub (75 points)

Provide **specific, respectful, and helpful** comments via GitHub. See the "Checklist" document for more detailed information.

- Leave at least **8 inline comments, suggestions, or edits** across the project.
- Be specific. For example, "*Consider renaming this function to clarify its purpose*" is more helpful than "*This is confusing.*"

### 3. Submit the Checklist and Reflection to D2L (20 points)

In the `PeerCodeReview_Checklist_Reflection.qmd` file, complete the checklist and write 7-10 sentences summarizing your review experience:

- What was done well?
- What was challenging about this assignment?
- What did you learn for your own projects?
- How did you approach constructive feedback?

Upload the knitted .pdf file to the "Peer Code Review" Assignment on D2L.

---

## Grading Rubric

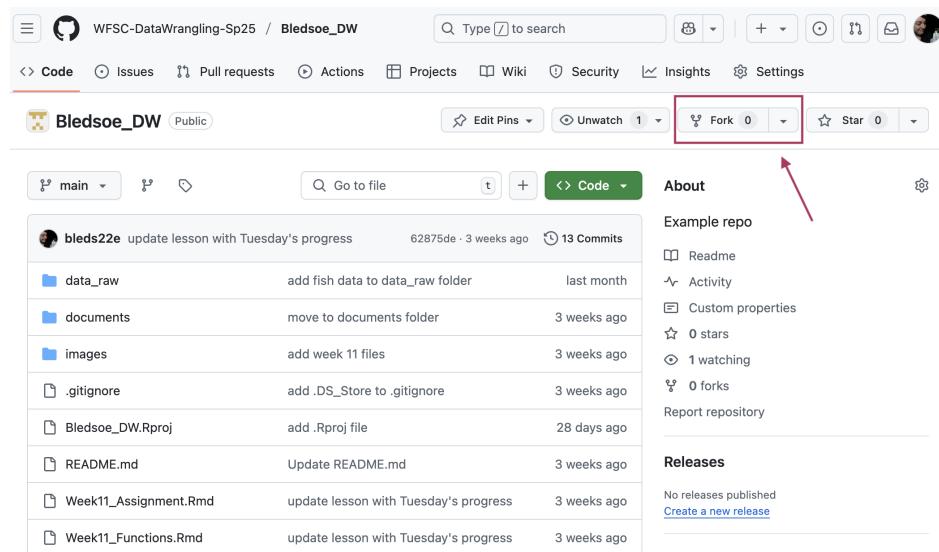
Category	Points	Criteria
<i>Repository Access &amp; Setup</i>	5	Repository is successfully forked and cloned. Reviewer created a new branch.
<i>Clarity &amp; Readability</i>	15	Comments address variable names, structure, and ease of understanding.
<i>Comments</i>		
<i>Functionality Comments</i>	15	Reviewer tested code and provided feedback on functionality and outputs.
<i>Code Organization</i>	15	Feedback includes directory structure and logic of .Rmd layout.
<i>Comments</i>		
<i>Documentation Comments</i>	15	Reviewer comments on README quality and .qmd documentation.
<i>Suggestions for Improvement</i>	15	At least 3 specific, actionable, and constructive suggestions are given.
<i>Checklist and Reflection</i>	20	Checklist completed and a thoughtful, specific, and complete reflection (7-10 sentences).
<b>Total</b>	<b>100</b>	

## Step-by-Step Instructions

Follow these steps to complete your peer code review using GitHub. This process uses a **fork-based workflow**. You will fork your peer's repository, add review comments in your fork, and submit a draft pull request with your feedback.

### Step 1: Fork and Clone the Repository

- Go to your assigned peer's GitHub repository via the provided link. You should have access (if not, email me!)
- Click the **Fork** button in the top-right corner to create your own copy of their repository under your GitHub account.



- Make sure you create the “forked” repo in your own personal account, not in the class organization!

The screenshot shows the GitHub interface for creating a new fork. At the top, the repository path 'WFSC-DataWrangling-Sp25 / Bledsoe\_DW' is visible. Below it, a search bar and various navigation icons are present. A red box highlights the 'Code' tab. The main area is titled 'Create a new fork'. It includes fields for 'Owner' (set to 'bleds22e') and 'Repository name' (set to 'Bledsoe\_DW'). A note says 'Bledsoe\_DW is available.' Below these, a message states that forks are named like their upstream repository, with an option to customize the name. There's a 'Description (optional)' field containing 'Example repo'. A checked checkbox 'Copy the main branch only' has a note below it: 'Contribute back to WFSC-DataWrangling-Sp25/Bledsoe\_DW by adding your own branch.' A note also says 'You are creating a fork in your personal account.' At the bottom right is a green 'Create fork' button.

### Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.

*Required fields are marked with an asterisk (\*)*.

Owner *	Repository name *
bleds22e	<input style="width: 100%;" type="text" value="Bledsoe_DW"/> <div style="font-size: small; margin-top: 2px;">Bledsoe_DW is available.</div>

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Example repo

Copy the main branch only

Contribute back to WFSC-DataWrangling-Sp25/Bledsoe\_DW by adding your own branch. [Learn more](#).

ⓘ You are creating a fork in your personal account.

Create fork

- Once you have successfully forked the repo, you should have a copy of your peer's repo in your personal GitHub account.

The screenshot shows the GitHub repository page for 'Bledsoe\_DW'. At the top, the owner is 'bleds22e' and the repository is 'Bledsoe\_DW'. It's a 'Public' repository 'forked from WFSC-DataWrangling-Sp25/Bledsoe\_DW'. The main branch is 'main'. The repository has 0 stars, 0 forks, and 0 watching. On the left, there's a file tree showing recent commits:

- data\_raw: add fish data to data\_raw folder (last month)
- documents: move to documents folder (3 weeks ago)
- images: add week 11 files (3 weeks ago)
- .gitignore: add .DS\_Store to .gitignore (3 weeks ago)
- Bledsoe\_DW.Rproj: add .Rproj file (28 days ago)
- README.md: Update README.md (3 weeks ago)
- Week11\_Assignment.Rmd: update lesson with Tuesday's progress (3 weeks ago)

The right sidebar contains sections for 'About', 'Example repo', 'Releases', and 'Packages'.

- On your forked copy, click the green **Code** button and copy the URL.

This branch is up to date with WFSC-DataWrangling-Sp25

**bleds22e** update lesson with Tuesday's progress

- data\_raw**
- documents**
- images**
- .gitignore** add .DS\_Store to .gitignore 3 weeks ago
- Bledsoe\_DW.Rproj** add .Rproj file 28 days ago
- README.md** Update README.md 3 weeks ago
- Week11\_Assignment.Rmd** update lesson with Tuesday's progress 3 weeks ago

**About**

Example repo

- Readme
- Activity
- 0 stars
- 0 watching
- 0 forks

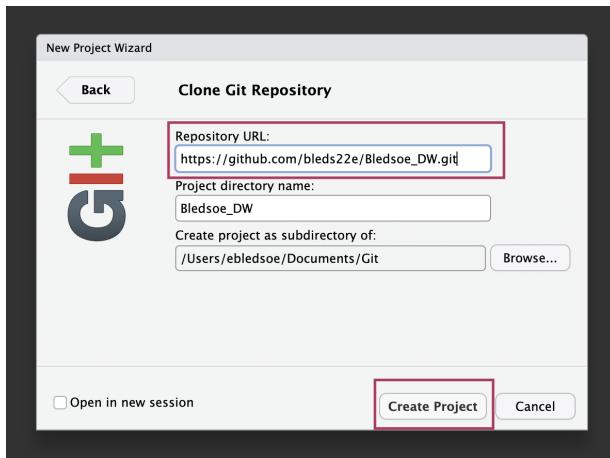
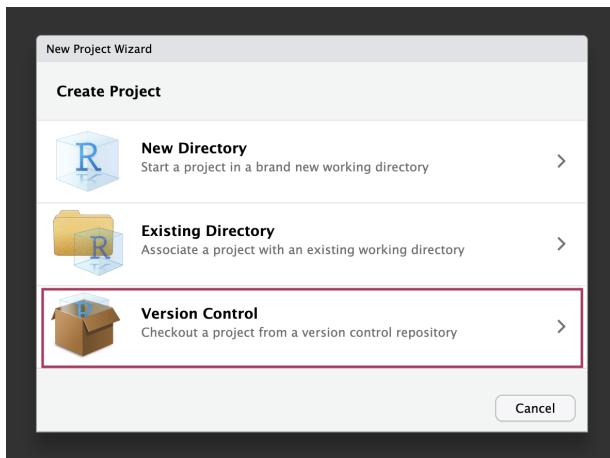
**Releases**

No releases published [Create a new release](#)

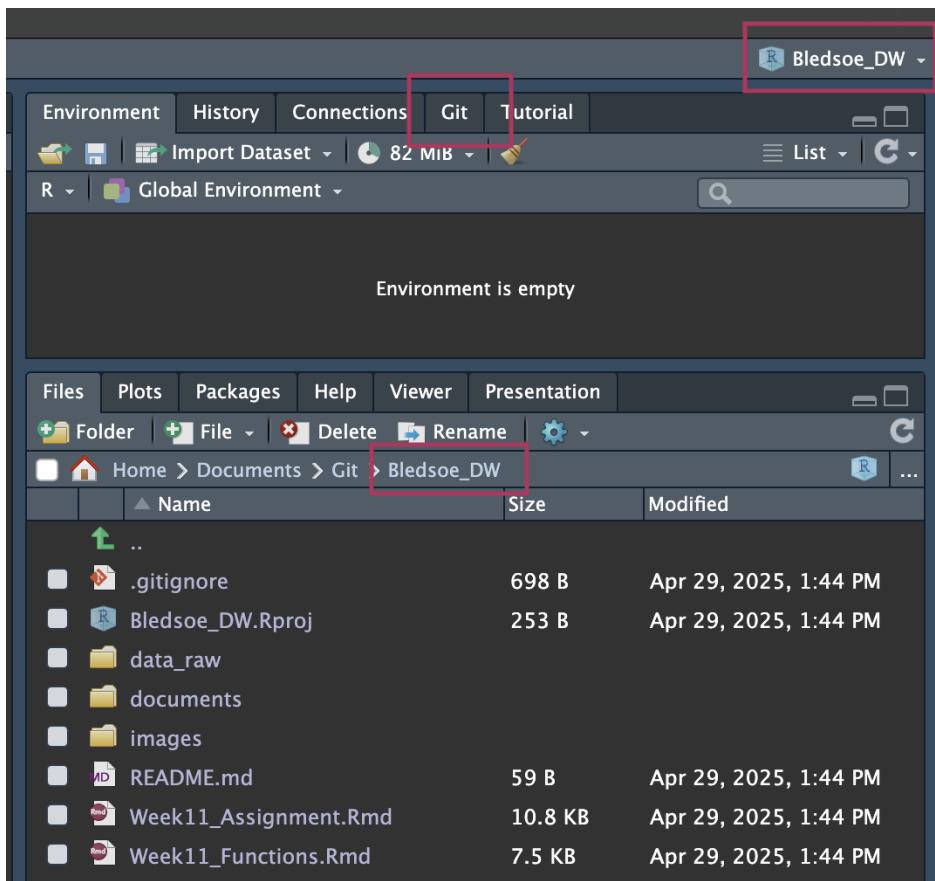
**Packages**

No packages published [Publish your first package](#)

- You can now proceed to create an RStudio Project linked to your forked version of the repository as you would with any Github repository.
- In RStudio: Go to *File > New Project > Version Control > Git*, paste the URL, and choose a folder.



- You should now have your own local copy of your peer's final project repo.



## Step 2: Add Comments to the Code

You should leave at least **eight comments** during your review.

1. Open the cloned project.
2. Leave review notes using comments (for example, lines that begin with “REVIEW:”).
3. Save the files after adding your comments.

```
Let's write out our first function. We want to calculate shrub volumes.

```{r}
# Add comments to code to ask questions or offer suggestions
# For example, I might suggest putting the functions that are used in the
# code at the top of the document or in a separate file, as is best practice!

# You can also make edits to code directly (adding a comment about why you
# made the change should go along with that).
# rename function to "calc_volume" (from "calc_shrub_volume") to make more
# general
calc_volume <- function(length, width, height){
  volume <- length * width * height
  return(volume)
}
```

```

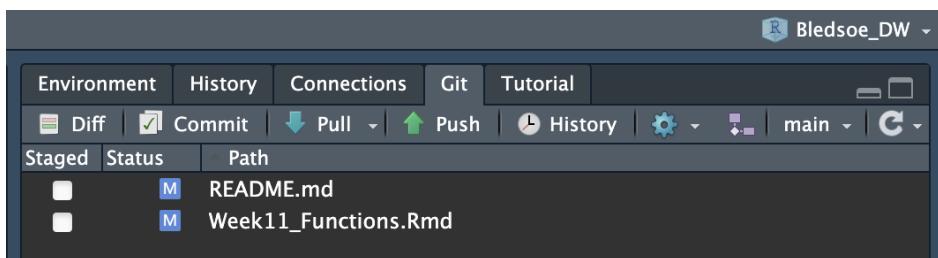
```

1 # Bledsoe_DW
2
3 Working repo for Data Wrangling--Spring 2025
4
5 I am making changes to this file.

```

### Step 3: Commit and Push Your Changes

- Save the comments or edits to the files.
- Commit and push your comments and edits
  - Use the Git pane to stage and commit the changes.
  - Write a brief commit message like “Peer review comments.”
  - Push the changes to your forked GitHub repository.
- Repeat this process as many times as needed.



### Step 4: Create a Pull Request to Submit Your Review

Once you have completed your comments:

- Go to your **forked repository** on GitHub and click on the “Contribute” button.

This branch is 1 commit ahead of WFSC-DataWrangling-Sp25/Bledsoe\_DW:main.

[Contribute](#) [Sync fork](#)

**bleds22e** useful commit message · e606b65 · 1 minute ago · 14 Commits

- Select “Open pull request.”

The screenshot shows a GitHub repository page for 'bleds22e / Bledsoe\_DW'. The repository is public and was forked from 'WFSC-DataWrangling-Sp25/Bledsoe\_DW'. The 'Code' tab is selected. A message at the top indicates that the branch is 1 commit ahead of the upstream 'main' branch. Below this, there are buttons for 'Contribute' and 'Sync fork'. A callout box highlights the 'Contribute' button. To the right, a list of 14 commits is shown, with the most recent being a merge commit from 'WFSC-DataWrangling-Sp25/Bledsoe\_DW:main'.

| Commit                           | Date        |
|----------------------------------|-------------|
| e606b65 · 1 minute ago           | 14 Commits  |
| add fish data to data_raw folder | last month  |
| move to documents folder         | 3 weeks ago |
| add week 11 files                | 3 weeks ago |
| add .DS_Store to .gitignore      | 3 weeks ago |
| add .Rproj file                  | 28 days ago |

- Confirm that:
  - The *base* repository is your peer's/the class organization's (e.g., `DataWrangling-Fall2025/project`).
  - The *head* repository is your fork (e.g., `yourusername/project`).
- Write a short summary of your review in the pull request description.
- Include the “3 actionable suggestions” from your review (and any other notes not included in the code).
- Click on the downward arrow on the right side of the “Create pull request” button and select “Create draft pull request.”

The screenshot shows the GitHub interface for creating a pull request. At the top, it says "Comparing changes" and "Able to merge". Below that, there are fields for "Add a title" (Peer Code Review) and "Add a description" (Opening a pull request (PR) from my fork of the repo to the original repo. This will show any/all changes I made to the files in my fork of the repo below.). On the right, there are sections for "Reviewers" (No reviews), "Assignees" (No one—assign yourself), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), and "Development" (Use Closing keywords in the description to automatically close issues). Under "Helpful resources", there is a link to GitHub Community Guidelines. At the bottom, there is a "Draft pull request" button with a dropdown menu showing "Create pull request" and "Create draft pull request" (which is highlighted with a red box). Other buttons include "Allow edits by maintainers" and "Allow edits by maintainers".

**Important Note!** You are submitting comments and suggestions only. Do NOT merge the pull request.

If you and/or you peer want to have further discussion about any of the comments or suggestions, you can use the “Conversation” and “Files Changed” tabs.

The screenshot shows the GitHub Peer Code Review interface for pull request #2. At the top, it says "Peer Code Review #2" and "Draft". Below that, there are tabs for "Conversation" (0), "Commits" (1), "Checks" (0), and "Files changed" (2). The "Files changed" tab is active, showing two files: README.md and Week11\_Functions.Rmd. The README.md file has a single commit with a message: "Working repo for Data Wrangling--Spring 2025". The Week11\_Functions.Rmd file has several commits, with the last one being "I am making changes to this file.". In the "Conversation" tab, there is a comment from "bleds2e" saying "Good idea!". The "Commits" tab shows a single commit from "bleds2e" with the message "Working repo for Data Wrangling--Spring 2025". The "Checks" tab shows no checks. The "Files changed" tab shows the code changes in the Week11\_Functions.Rmd file, with some lines highlighted in green and red. The "Conversation" tab also shows a comment from "bleds2e" with the message "Good idea!".