

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT114-004-S2024/generic-git-readings-via-local/grade/grg>

IT114-004-S2024 - [Generic] Git Readings via Local

Submissions:

Submission Selection

1 Submission [active] 1/29/2024 8:59:17 PM

Instructions

^ COLLAPSE ^

Preliminary Setup:

- 1 .Go to w3schools.com
- 2 .Create an account (preferably with your college account)
- 3 .Visit my-learning.w3schools.com/tutorial/git
- 4 .Complete the following readings:
 - 1 .Essentials 1.1, 1.2, 1.3
 - 2 .Essential Commands 2.1, 2.2, 2.3, 2.4
 - 3 .Branch Management 3.1, 3.2
 - 4 .Remote Collaboration 4.1-4.9
 - 5 .Security Practices 6.1-6.3
 - 6 .Attempt the Git Quiz (aim for <=70%)
- 1 .Verify you're in the main branch via ``git status`` or ``git branch``
 - 1 .If not, ``git checkout main``
- 2 .Create a branch for this assignment ``git checkout -b M1-Git-Readings``
- 3 .**Note:** In this assignment, we'll make the pull request later. In future assignments, we'll likely open it earlier so we can use the URL for assignments
- 4 .Fill in the items in the worksheet below (save as often as necessary)
- 5 .Once finished, export the worksheet
- 6 .Take the exported file and add it anywhere in your repository (a Module1 folder is best, but not required)
- 7 .Make sure git detects it by checking with ``git status``
- 8 .If everything is good, continue to submit
 - 1 .Track the file either with ``git add path/to/file`` or ``git add .``
 - 2 .Commit changes via ``git commit -m "some relevant message"``
 - 3 .Push the changes via ``git push origin M1-Git-Readings``
- 9 .Go to GitHub and use the dropdown in the top left to find the M1-Git-Readings branch and ensure the file is present
- 10 If the file is there, either use the pull request popup or go to the pull request tab and open a request where main is base and M1-Git-Readings is compare
- 11 Open and complete the merge of the pull request (it should turn purple)
- 12 Go to your M1-Git-Readings branch on GitHub and navigate to the submission file
- 13 Paste the direct link to that file on Canvas

Branch name: M1-Git-Readings

Tasks: 7 Points: 10.00



Github Readings (10 pts.)

^ COLLAPSE ^



^ COLLAPSE ^

Task #1 - Points: 1

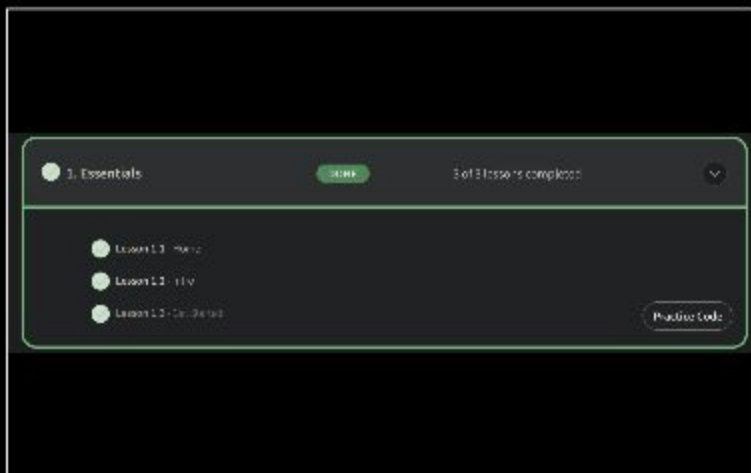
Text: Complete Essentials

Details:

Lessons 1.1-1.3

Task Screenshots:

☐ Large Gallery



lessons 1.1 - 1.3



^ COLLAPSE ^

Task #2 - Points: 1

Text: Complete Essential Commands

Details:

Lessons 2.1-2.4

Task Screenshots:

☐ Large Gallery




lessons 2.1 - 2.4



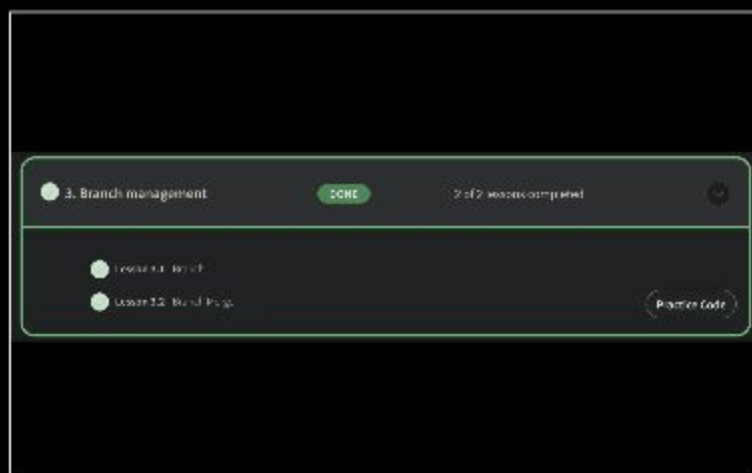
Task #3 - Points: 1

Text: Complete Branch Management

 Details:
Lessons 3.1-3.2

Task Screenshots:

☐ Large Gallery




lessons 3.1 - 3.2



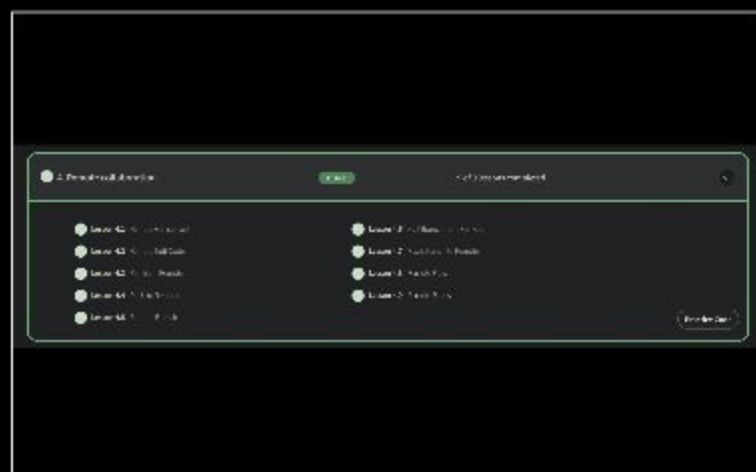
Task #4 - Points: 1

Text: Complete Remote Collaboration

 Details:
Lessons 4.1-4.9

Task Screenshots:

☐ Large Gallery



lessons 4.1 - 4.9



^ COLLAPSE ^

Task #5 - Points: 1

Text: Complete Security Practices

Details:
Lessons 6.1-6.3

Task Screenshots:

☐ Large Gallery



lessons 6.1 - 6.3



^ COLLAPSE ^

Task #6 - Points: 1

Text: Attempt Quiz ($\geq 70\%$)

Details:

Aim for 70% or higher for full marks

Task Screenshots:

☐ Large Gallery



git quiz



COLLAPSE

Task #7 - Points: 1

Text: Reflection

Details:

Summarize your understanding of the readings and how we'll be using git in this class.

Mention any issues you might have encountered and how you resolved them.

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Mentions issues or no issues
<input type="checkbox"/> #2	1	Covers core concepts related to the readings
<input type="checkbox"/> #3	1	At least a few reasonable sentences

Response:

I have been using git -- a combination of git, github, gitkracken, and DevOps -- for several years now, and use it for all of my personal projects, school projects, and professional work, as well as using it to keep track of documents (usually latex documents to keep track of blame), as well as several other things. I expect that I will have no issues. This course will primarily be using rather simple aspects of git -- such as:

1. initializing git in a repo and connecting it to a remote origin (a github repo)
2. staging changes and committing those staged changes
3. creating branches to isolate working on different features / fixes
4. merging branches, handling pull requests & code reviews

5. creating forks of repos to 'make them our own' and then make pull requests back to the original cloned repo

End of Assignment