

submission

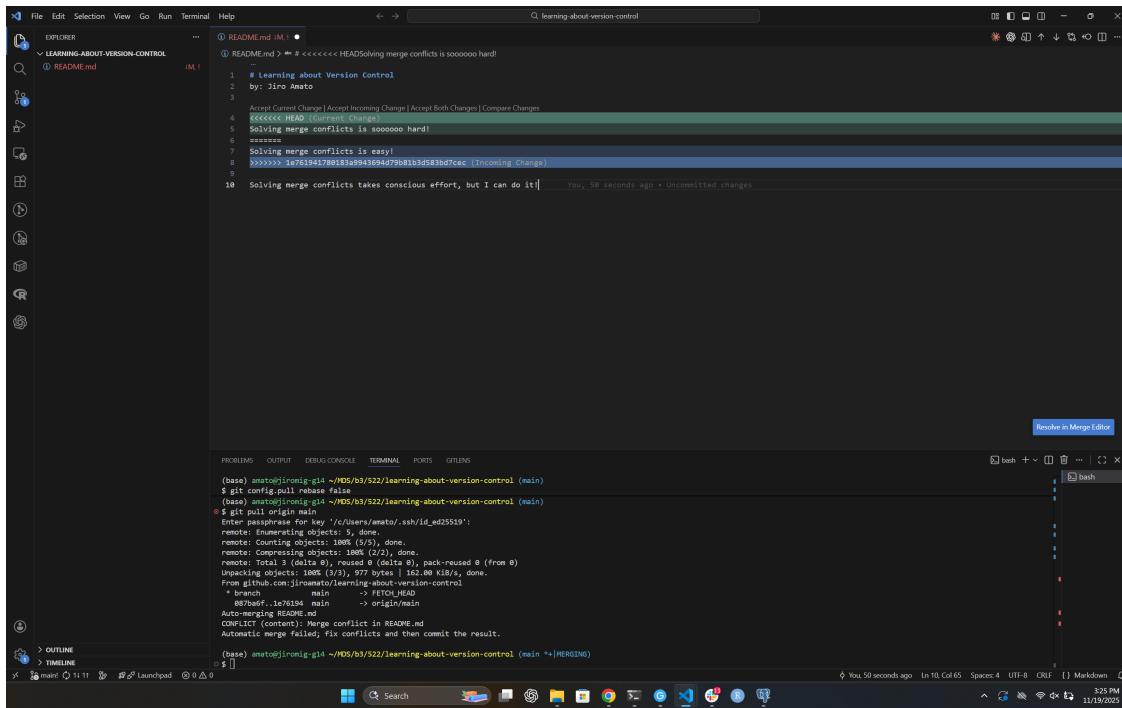
November 19, 2025

1 Individual Assingment 1 Submission

Commit History Link:

<https://github.com/jiroamato/learning-about-version-control/commits/main>

Resolving Merge Conflict:



The screenshot shows a terminal window with a code editor overlay. The code editor displays a file named README.md with the following content:

```
1 # Learning about Version Control
2 by: Jiro Amato
3
4 Accept Current Change | Accept Incoming Change | Accept Both Changes | Compare Changes
5 =====
6 Solving merge conflicts is sooooo hard!
7 =====
8 Solving merge conflicts is easy!
9 >>>> aef7a1b17170101a9a94394d79b81b3d833d7cec (Incoming Change)
10 Solving merge conflicts takes conscious effort, but I can do it!
```

The terminal below shows the command history and the execution of a git pull command, which triggered a merge conflict:

```
(base) amato@ironiq-g14 ~/WDS/b3/522/learning-about-version-control (main)
$ git config.pull.rebase false
(base) amato@ironiq-g14 ~/WDS/b3/522/learning-about-version-control (main)
$ git pull origin main
Enter passphrase for key '/C/Users/amato/.ssh/id_ed25519':
remote: Enumerating objects: 5, done.
remote: Counting commits: 1, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 977 bytes | 162.00 kB/s, done.
From https://github.com/jiroamato/learning-about-version-control
 * branch      main       -> FETCH_HEAD
 887b0eff.1e91519a main      -> origin/main
Automerging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.

(base) amato@ironiq-g14 ~/WDS/b3/522/learning-about-version-control (main +|MERGING)
```

Working on a branch-pull-request workflow allows collaborators to separate changes to the repo while keeping the main branch thus preventing breaking anything on main. Also, by creating pull-requests, changes have to be reviewed so that the entire team can collectively look at potential changes. Directly committing creates an unstructured repository where changes are not coordinated within a team.