**Fast-Forward Merge**

A fast-forward merge happens when one of the branches being merged fails to diverge from the other branches it is being merged into. Since there aren’t any conflicts that arise from this process, Git moves the branch pointer forward to the latest commit of the merged branch. Since there is no divergence occurring, the process remained linear and Git was able to simply understand it. This avoids any merge redundancies that arise and makes the commit history easier to follow since each branch is linearly aligned.

**Conflicts**

B. A merge conflict occurs when two branches update/modify the same line in the same file to give different information. For example, if I am collaborating on one branch and another person is doing the same on a different branch, if we were both working on the same file and put different code on the same line then a conflict will arise from merging the branch since Git does not know which information is correct.

C. If both branches modify different parts of the same file, they can be merged without conflict. For example, if one person is aggregated to working on the header of the page and another only edits the main content on their branch, then a conflict shouldn’t arise because they shouldn’t be modifying the same lines of code.

D. Git Graph Screenshot

A screenshot of a computer

AI-generated content may be incorrect.