Reflective Note - Git Merge Conflict (Simple Calculator project)

Cause of Conflict:
A merge conflict occurred because the same line in calculator.py (the add function) was modified differently in two branches (main and feature-update). Git cannot automatically decide which change is correct when the exact same region of text is changed.

How it was Resolved:
1. Ran 'git merge feature-update' which produced conflict markers in calculator.py.
2. Opened the file, manually edited the conflict region to choose the correct implementation.
3. Staged the file with 'git add calculator.py' and completed the merge with 'git commit'.
4. Pushed the resolved main branch to remote with 'git push'.

What I learned:
- Branching and merging allow parallel work, but conflicting edits require human review.
- Use small commits and communicate with teammates to avoid conflicts.
- Tools like 'git status' and 'git diff' help locate and resolve conflicts.

Exact commands used during resolution (examples):
git checkout -b feature-update
git commit -am "Updated add function in feature-update"
git checkout main
git commit -am "Changed add function in main"
git merge feature-update
git add calculator.py
git commit -m "Resolved merge conflict in add function"
git push origin main

End of reflective note.