NAME: ADEBIMPE ADEROJU M.

COURSE: CSE210- PROGRAMMING WITH FUNCTIONS

INSTRUCTURE'S NAME: JOHN READING

WHAT IS VERSION CONTROL AND WHY IS IT IMPORTANT?

Version control is basically a smart way to save every change you make to your code, so you can go back, see who changed what and when, and keep everyone on the same page as a team. It's a system that records changes to files (codes) over time. It lets you look at earlier versions, see exactly who changed what and why, and undo mistakes if it is needed. Tools like 'Git' manage a repository that holds a code and its entire history, which makes collaboration, branching, and exploring the history of the code easy for every team member.

A key benefit of Version control is collaboration without chaos. It simply means that lots of coders can work on different features at the same time without stomping on each other's work or stopping others in their work. Changes get merged in a controlled way, conflicts are sorted out, and you always have a history you can rely on if something goes wrong.

An example of how a dev team might use it is having a team sets up a central repository on GitHub. Each person on the team creates a feature branch from main, writes her code, and saves changes with meaningful commit messages. They push the branch to GitHub and open a pull request when the feature is ready. Teammates review, run tests, and when everything looks good, the feature gets merged into main. If a bug crops up later, they can revert to a past commit or pick a fix from branches to merge into the main. The history shows who did what and why.

A common command in Version control is **"git commit -m "Add user authentication feature""—** this saves the staged changes with a descriptive message.

Version control matters because it's a reliable record of how a project evolves, which helps us track changes, debug, and learn from mistakes. Branches let us try new ideas without breaking the main code, merging makes it smooth to bring those ideas back in, and tags/releases help us mark shipped versions. Overall, it reduces risk, helps teams work together, and keeps a clear trail of every tweak, mistakes and improvement.