How did viewing a diff between two versions of a file help you see the bug that

was introduced?

It helps us in recognising the difference points in the file which helps us to recognise the possible locations of the appearance of a bug.

How could having easy access to the entire history of a file make you a more

efficient programmer in the long term?

It could help us in realising the different checkpoints in time where we made some changes which will prevent us from going through the same process again.

What do you think are the pros and cons of manually choosing when to create a

commit, like you do in Git, vs having versions automatically saved, like Google

docs does?

By manually choosing when to do a commit we can depict a logical structure in our version control without letting it get cluttered.

Why do you think some version control systems, like Git, allow saving multiple

files in one commit, while others, like Google Docs, treat each file separately?

By allowing to save multiple files in one commit we can depict a logical change in all three files when they are related with each other and prevents discomfort.

How can you use the commands git log and git diff to view the history of files?

By using git log we can view that where a change was made with a particular message and by using git diff we can see the changes mae.

How might using version control make you more confident to make changes that

could break something?

As in version control we have the previous version of the file saved before making a commit in case we broke something we can restore it to its older version.

Now that you have your workspace set up, what do you want to try using Git for?

As a simple command line interface and for future version control.