I have to say that being part of Team Bravo was a great experience. We were all honest about our skill set, along with providing feedback respectfully. Everyone on Team Bravo truly brought forth something valuable, making everyone an asset. I also have to say how laid back everyone was, which was very refreshing.

While working on our case study, Willson Financial, I learned many things. For starters we should have utilized Git a bit more, so we did not have to keep downloading and reuploading our scripts nonstop. Had we just utilized a GitHub repository, editing our scripts would have been much easier for everyone to access in real-time without the added hassle of Google Drive. There were a couple times I had almost uploaded the wrong file which would have broken everyone’s script.

Another lesson I learned was that staying organized is key through development, such as the one for Willson Financial. If something that is small scale needed so much thought and documentation, I cannot even imagine what would be needed for something much larger scale with a larger group. Team Bravo honestly did great with our shared Google Drive. Everyone remembered to document what was needed from them where it was needed. This helps the initial script to make our database much simpler.

In hindsight, we all should have agreed upon a database name before starting. We all ended up with different variations of the same database name. So, when we went to run our SQL script to create the database, we all had to remember to update that database name and our configuration portion. I think things like this come with time and experience. Furthermore, I learned that not being on the same network, as one would at your job, makes sharing the same database a bit tougher. Which goes back to the lesson of we should have utilized GitHub.

The most important lesson I learned throughout this experience was learning to be flexible and that things will change on the fly. We created our initial ERD before scripting any of the queries needed. Once we began scripting those queries, we noticed that our initial ERD did not make sense. Plus, we had not correctly configured our “Assets Table.” Once we started talking things through over our queries that we wanted to run but also the expected outcome we needed, we realized that we needed to go back to the drawing board. There was no panic or complaining; we worked to make it happen.

The things we could have improved on was maybe challenging each other a bit more. I am not sure if everyone was just polite, or maybe it was because we do not have a lot of experience with SQL, but I can’t help but wonder if someone had caught some mistakes but did not want to come off as rude. I think in the future, I will maybe try to find a way to speak up so that it does not give the team more work later, like in this case. This is a lesson I can apply to both my school setting but especially at work.