During the assignment, the first task I tackled was creating the data class. This class contained all the getters and setters, as well as the functions used for each method in my grade calculator. I designed the class to take four arrays: the received grades for levels 5 and 6, as well as the credits for both levels. I chose this approach because arrays are easy to work with and made calculating the total credits and final weighted grades straightforward, as I could simply loop through the arrays. Additionally, I included the number of modules as an input. However, I ultimately did not use this input for anything.

The next class I created was the GUI/Main class, which inherited from the data class. I decided to separate the data and GUI into different classes because combining everything into a single class made the code difficult to manage and debug, especially given the complexity and size of some of the GUI functions. The GUI class also handled interactions with the database. I chose to implement database actions as event handlers which are triggered by button clicks. This approach felt more intuitive and made it easier to manage database actions compared to creating separate functions and calling them independently.

Overall, I feel that my approach to the calculator itself was effective, and I am fairly happy with what I created. However, reflecting on my approach to aspects outside of the calculator, there are several things I would approach differently if given the chance to redo the assignment.

The primary adjustment would be to complete the post tasks as they were assigned, rather than waiting to do them until I started the main assignment. Waiting to do these tasks significantly reduced the time I had available to focus on developing the actual calculator. This had a noticeable impact on my assignment, particularly on the UI, which ended up being very simple. If I had more time, I would have liked to create a more polished and visually appealing user interface.

Additionally, while creating unit tests for my assignment, I encountered issues with JUnit 5 in NetBeans. I spent a lot of time troubleshooting these problems, only to eventually downgrade to JUnit 4 to proceed with the assignment. In hindsight, I wish I had done more research before starting development. This would have saved me significant time and allowed me to focus more on refining other aspects of the assignment.

Another aspect I would change is the unused "number of modules" variable in my code. This variable was a leftover from an earlier idea for the calculator, where I intended to use it instead of calculating the length of the arrays I was looping over. I initially thought this approach would be more efficient since the number of modules would always match the length of the arrays. However, I ultimately decided to calculate the array length during each loop. Unfortunately, I ran out of time to clean up this unused

variable. If I could go back, I would focus on managing my time more efficiently to address issues like this and improve the overall quality of my code.

In conclusion, while I am generally happy with the overall functionality and design of my grade calculator, reflecting on the approach has highlighted several areas where I could have managed my time and resources more efficiently. Completing the post-tasks as they were assigned would have given me more time to work on the calculator. Additionally, early research into the issues with JUnit 5 would have saved valuable time and allowed me to focus more on enhancing the quality of my code. Finally, ensuring that unused variables and redundant code were removed in time would have contributed to a more efficient project.