Module #3 Report | CSE 310 – Applied Programming

Name	Date	Teacher
Jacob Emhoff	3/26/2025	Brother Porter McGary

Project Repository Link

Magic Database Github Repository Python + DB Demonstration Link

Module

Mark an **X** next to the module you completed

Module		Language
Cloud Databases		Java
Data Analysis		Kotlin
Game Framework		R
GIS Mapping		Erlang
Mobile App		JavaScript
Networking		C#
Web Apps		TypeScript
Language – C++		Rust
SQL Relational Databases	X	Choose Your Own Adventure

Fill Out the Checklist

Complete the following checklist to make sure you completed all parts of the module. Mark your response with **Yes** or **No**. If the answer is **No** then additionally describe what was preventing you from completing this step.

Question	Your Response	Comments
Did you implement the entire set of unique requirements as described in the Module Description document in I-Learn?	Yes	
Did you write at least 100 lines of code in your software and include useful comments?	Yes	
Did you use the correct README.md template from the Module Description document in I-Learn?	Yes	
Did you completely populate the README.md template?	Yes	

Question	Your Response	Comments
Did you create the video, publish it on YouTube, and reference it in the README.md file?	Yes	
Did you publish the code with the README.md (in the top-level folder) into a public GitHub repository?	Yes	

Did you complete a Stretch Challenge

I had two stretch challenges for this project. The first was to create a join between two tables in the database. This is done when you select option 5 from the Main Menu. The other stretch I assigned was to create a color to symbol convertor, and I completed it.

Record your time

How many hours did you spend on this module and the team project this Sprint?
Include all time including planning, researching, implementation, troubleshooting, documentation, video production, and publishing.

	Hours
Individual Module	16
Team Project	5

Retrospective

- **What learning strategies worked well in this module?** I knew these languages fairly well previous to the completion of this module, but I still accepted there was more to learn with them. A side effect from this previous expierence manifested itself in the form of knowledge and debugging/syntax fixing. I knew where to go to answer my questions, I've used W3 Schools numerous times to fix or learn SQL syntax, and I'm familar with Python documentation. The strategy here was knowing where to look for answers, and expanding my knowledge of well-known subject.
- **What strategies (or lack of strategy) did not work well?** I didn't really plan out the specific ways I would execute certain code and inserts/deletions. I had a general idea of implementing all four parts of CRUD, but I never broke it down into a statement by statement basis. Basically I alloted the same amount of time to each part, when creating a entry was significantly easier then updating one (i.e. ~1 hour to 4+ hours).
- **How can you improve in the next module?** The set of unique requirements we're supposed to list out at the start of every module is something I really need to improve. I've been listing mostly bullet points, and not specific requirements. I also recognize it's rather difficult to list specific requirements for a language or process you're trying to learn. To combat this, I'll try to research my project a little more, while still understanding that perfect requirements aren't practical.