In this course, you learned key concepts in using web applications. How would you describe your ability to relate the course concepts to application in the workplace? Provide two to three examples.

Research and examine the differences between relational and non-relational databases. In 150–200 words, describe the differences between the two database technologies.

Explore the features of the MongoDB non-relational database. If you had to choose between the MySQL database and the MongoDB database, or could select either database for an application, which would you choose? In at least 150 words, explain your rationale for choosing the database or databases selected.

Professor and Class,

I’ve decided to answer both the technical and non-technical questions. I’ll start with the non-technical question.

Previous courses in this degree plan had me wondering if the short programming boot camps I’ve seen advertisements for all over the place would have been a better choice than an actual degree, but this is one of a few courses that makes me feel that the degree is worthwhile. Learning languages is important, but I’ve realized that all languages can be self-taught with free resources online. What we need to know is what to actually expect out of programming jobs and what skills employers would be searching for aside from certain languages and software. This course not only taught me about two of the most popular frameworks that are most likely to be used by companies, but it also taught me how to use gits along with how to write reports using markdown which are real-world skills needed for these jobs. I’d say that this course gave me more confidence in my chosen field of study.

I’ll answer the technical question next. This one may require a bit of research.

Relational databases are made up of tables that connect via keys such as ID’s. Companies sometimes use keys such as phone numbers since they are unique and can’t be duplicated- I’m pretty sure this is how AT&T’s databases are set up, having worked as a sales representative there in the past. Non-relational databases are made up of tables within tables- kind of like Russian nesting dolls. No-SQL databases such as MongoDB stores data in JSON format, which I could see being useful to apps such as the music app. Personally, I just feel more comfortable enough using SQL databases though. The language and desktop app are easy enough to use, but really it’s just a personal preference for me.

Dioukou,

I’ve never heard of PostgreSQL. How similar is it to MySQL? Also, I forgot that MongoDB does provide more scalability. That would indeed factor into the decision-making process. I was thinking of small projects, but if I were to be creating something more along the lines of a social media application I would definitely require scalability over my own personal preference of a MySQL database.

Jess,

That’s right, I forgot that MongoDB is good for changing the data structures after the database has already been deployed. It would require less work to maintain the database. It would be good for testing out a database connection or just starting out on a project that will have many changes in the future. MySQL would be better suited for a project that is already mostly or entirely thought out.

Dempsey,

I didn’t realize that both could be used for one project. It would be complicated but separate databases like that might be more secure too. It kind of reminds me of how hybrid cloud models allow data to be stored partially in a public model and partially in a private model. When it comes to the hybrid cloud model, I’m sure that the more critical information would be stored in a private model for more security. I wonder what data would be stored in each kind of database if a project used both.

Professor and Class,

Raw SQL frameworks involve putting full SQL queries into the code in order to access and manipulate the database from the project. I can see this allowing the programmer more control over the database itself but it could also take up much more time to develop. Object-Relational-Mapping frameworks, on the other hand, take much less time as the developer doesn’t have to write full SQL queries into the code. ORM frameworks do have limitations when manipulating the database so developers who require full control of a database may need to turn to RawSQL framework to meet their needs.