HOMEWORK 2A -T-SQL: SINGLE TABLE QUERIES

**1.  VERSIONS of Server.  You have founded a new company with two friends. Your new application (app) uses a SQL Server database to store information. You are unsure whether your app will be successful but if it is, you will need both high performance and space for large volumes of data. However, you have not yet launched, so are unsure how many people will use your app. Which edition of SQL Server 2016 should you use for this system? Why?**

In order to address this question of which edition of SQL server 2019 should you use, it is important to understand the requirements of your project and your target audience (potential customer/consumer base). But if you want to account for future customer usage, it is important to include a feature that is scalable at a moment’s notice. Scalability allows for the system to increase or decrease in size based off demand in order to reduce the chances of system outages or bottlenecks. I would recommend using the “Stretch Database” in SQL Server 2016. The stretch database feature lets programmers securely migrate and archive data from a local server to an Azure server in the cloud environment.

**2.  What is predicate logic? How is it a different way of thinking about data and the processing that you want to do on data? What does it mean?  What is a "predicate?"  Describe each of the below T-SQL elements as it relates to predicates:**

Predicate logic is the concept SQL uses to select and retrieve data. Predicate logic states that an expression (SQL Query) is either true or false. The RDBMS relies on predicates to maintain the logical integrity of the data and defines the databases’ structure. Predicates are conditions or clauses that are part of the query to help retrieve specific values if the query is evaluates to be true. If the conditions evaluate to be false, information will not be retrieved or displayed.

* **WHERE –** Used to filter records to specifically retrieve information if the specified condition is met.
* **JOIN conditions-** Used to combine rows from 2 or more tables based on the related column between them.
* **HAVING clauses –** This is used in conjunction with a WHERE clause. Having clauses are used to address aggregate functions.
* **WHILE statements -**Sets a condition for the repeated execution of a SQL query. The statements are executed repeatedly as long as the specified condition is true. (Equivalent to C# DO-WHILE Loops). If the condition returns false, the code breaks out of the loop and continues with the default code block.

**3. Going wild with wildcards.  Using wildcards as arguments (such as Select \*) is not recommended. Why? What if you really need to do such a "wild" search, select, or filtering? What trade-offs should you consider? Alternatives?**

 Using wildcard (\*) as part of your select statement is not good practice as an emerging data/software engineer. What the wildcard does in a SQL query is that it selects and displays ALL columns and its associated data. This is bad practice because in a constrained environment and depending on the size of the Database, it may cause network issues for you and your team because you are retrieving massive amounts of data.

It is better practice to understand the schema of the database, understand what each column represents, and determine from there what information you specifically care about. This is a better alternative if you need to conduct a wildcard search because most of the time, a wildcard search is used to understand what information is present in the database. With this mindset, you are able to draft more efficient and focused queries that results with more precise information.

**4.  JOINS.  Inner and Outer joins are talked about a lot as we use tables and database systems.  What do these mean, and why you'd need to use them. What if you're operating in a non-SQL kind of environment, where you don't have JOIN operators? How might you accomplish the same results?**

A Join clause is used to combine rows from 2x or more tables that are based on a related column between them. This is a way to create a table that shows related information. Outer Joins (left and right) are commands used to return all records from the respective table that matches the other. This is needed to maintain data and referential integrity in your datasets.

If you are operating in a non-sql environment where one does not have join operators, one method to address the issue while achieving the same results is to create a spreadsheet in excel from scratch and create the table with only the columns and values you want.

**5. Procedural, Object, or Data: Perspectives?  As a programmer, you have to focus your thinking either on the step-by-step of what you want to do, or the object-by-object, data-item-by-data-item that you need to do those steps to.  How does your viewpoint or thought process change as you "think in SQL?"**

As a programmer, I believe it is important to have all of these mental models and be aware of when to be using those methodologies. Procedural, Object and Data perspectives is needed based on what the programmer is doing at the moment.

When it comes to SQL thought processes, I believe it is important for a programmer to view think more abstractly and try to understand how the data is related to one another. Thinking in a sequential manner when working in a SQL environment may not be the most efficient way to accomplish tasks(retrieve and display information).

**References**

1. Ben-Gan, Itzik (2019). *T-SQL Fundamentals, Third Edition.*Microsoft Press. Chapter 1
2. W3Schools.com. *“SQL Tutorial” – While, Having, Joins.* <https://www.w3schools.com/sql/default.asp>
3. Microsoft Docs. (2017). *“WHILE (Transact-SQL)”.* [*https://docs.microsoft.com/en-us/sql/t-sql/language-elements/while-transact-sql?view=sql-server-ver15*](https://docs.microsoft.com/en-us/sql/t-sql/language-elements/while-transact-sql?view=sql-server-ver15)
4. Microsoft Docs. (2017). *“What’s new in SQL Server 2016”.* [*https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-2016?view=sql-server-ver15*](https://docs.microsoft.com/en-us/sql/sql-server/what-s-new-in-sql-server-2016?view=sql-server-ver15)