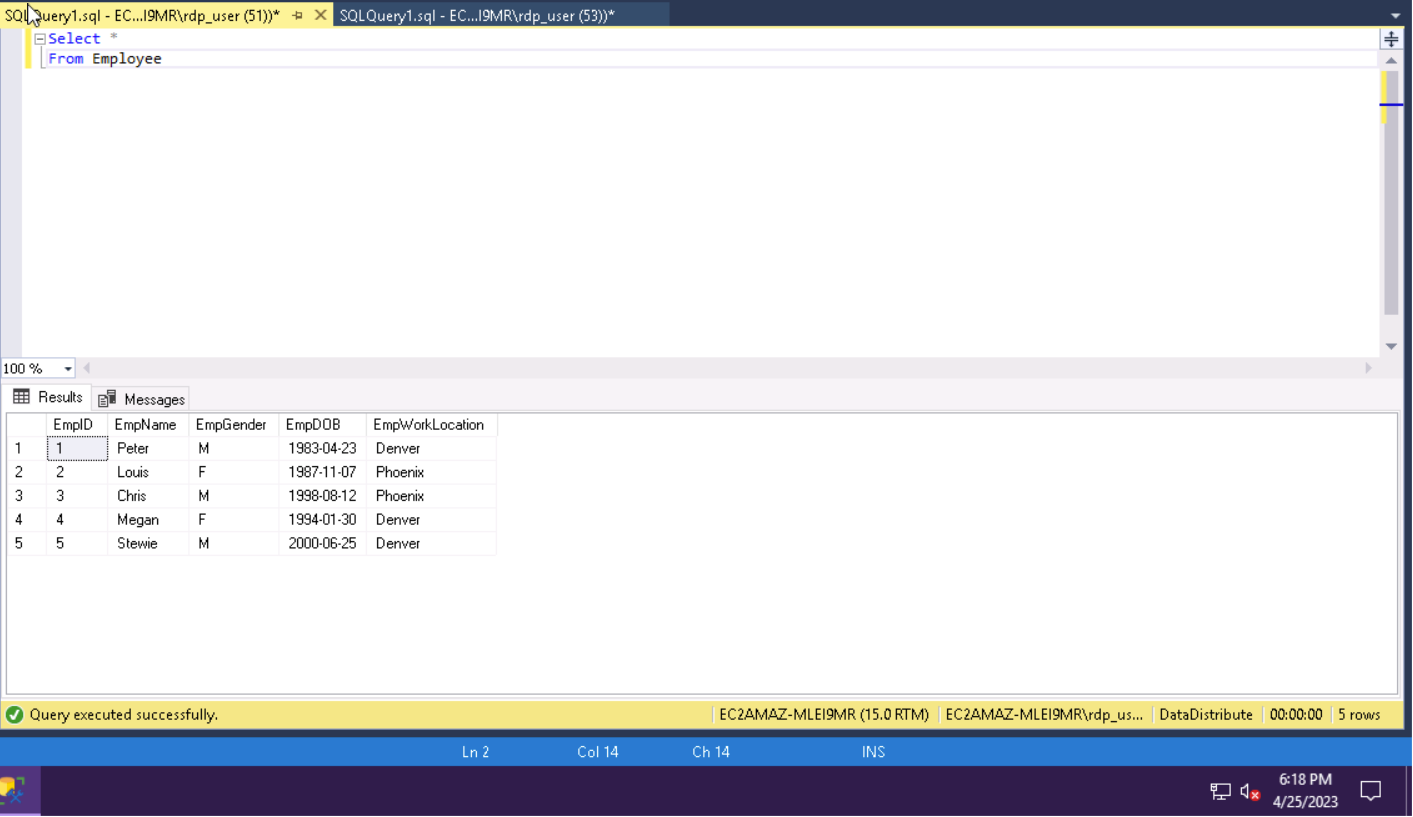
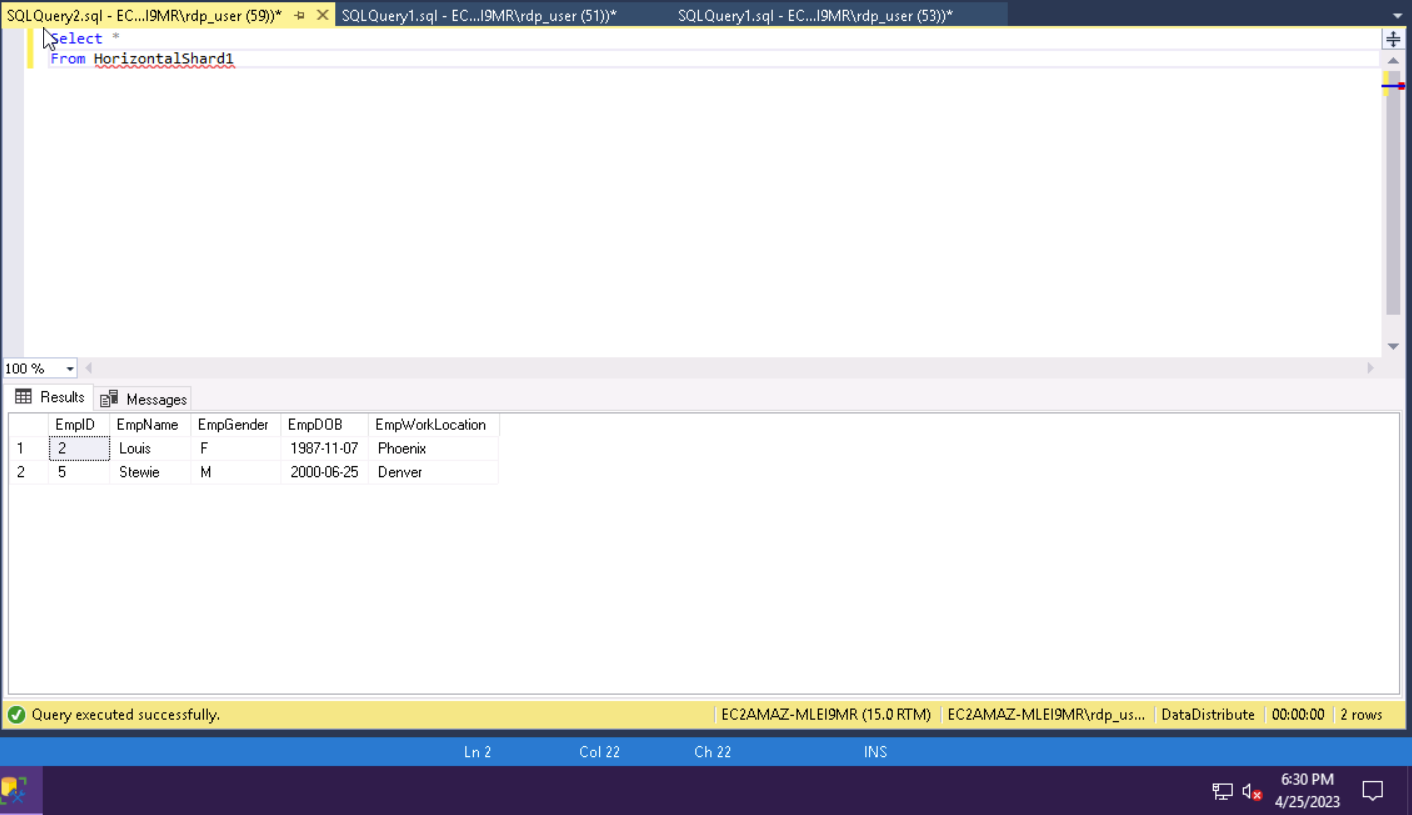
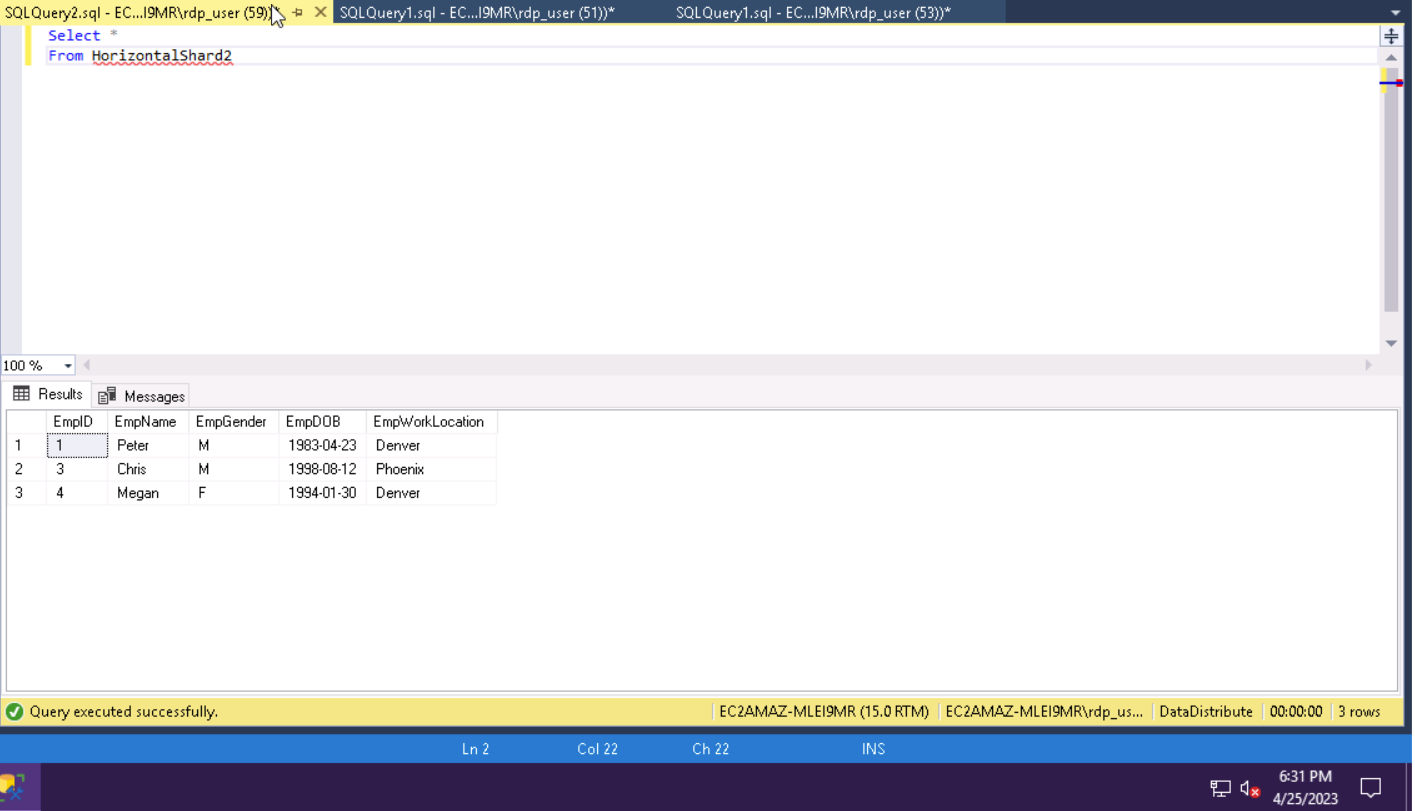
Michael Lee, Robert Minson, Zaid Flores, Luke Mercurio, Tai Kauhaa-Po

Data Distribution Strategy

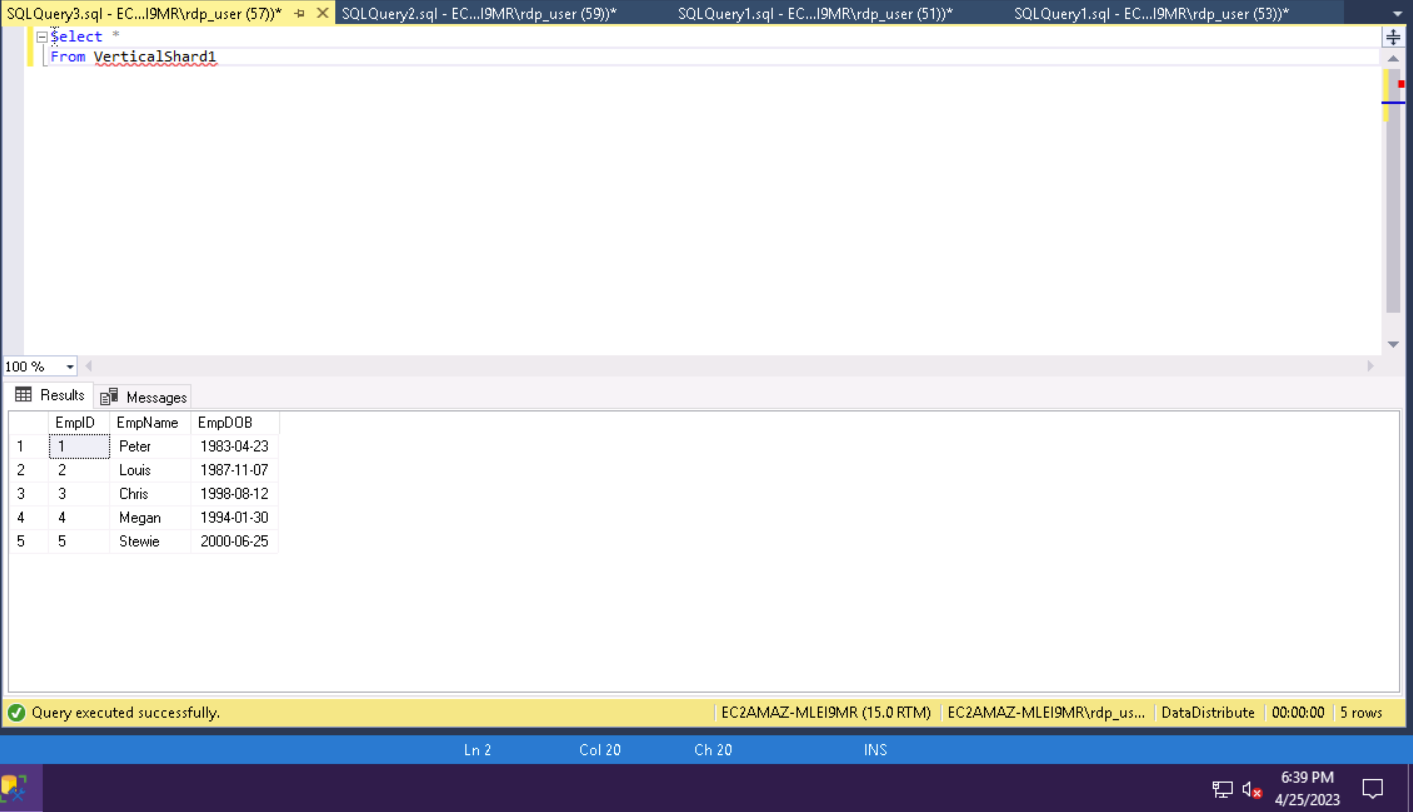
April 30, 2023

Full Employee TableHorizontal Tables





Vertical Tables



Explain and apply horizontal partitioning to this table and the one above.

Graphical user interface, application

Description automatically generated

**Explain Horizontal Partitioning**

Horizontal partitioning involves taking data that already exist within a table and dividing the data into multiple tables with the same number of columns but fewer rows. According to sqlshack, “ Horizontal partitioning divides a table into multiple tables that contain the same number of columns, but fewer rows. For example, if a table contains a large number of rows that represent monthly reports it could be partitioned horizontally into tables by years, with each table representing all monthly reports for a specific year. This way queries requiring data for a specific year will only reference the appropriate table. Tables should be partitioned in a way that queries reference as few tables as possible” (Medic). The full table shows EmpID, EmpName, EmpGender, EmpDOB, and EMPWorkLocation. After a Horziontal prtitioning is executed, the new table only shows EmpID, EmpGender, and EmpWorkLocation. This table only shows the columns for ID, Gender, and work location and can help a company determine how many female and male employees work in different locations. This can be helpful for creating statistics for the company to show whether they are a diverse or not a diverse company.

**Explain Vertical Partitioning**

Vertical partitioning consists of breaking a table vertically into columns, with distinguished columns stowed on different partitions (Custer, 2023). One may want to partition a table vertically when they want to simplify another table to focus on specific columns. In this assignment, the main Employee table consists of EmpID, EmpName, EmpGender, EmpDOB, and EMPWorkLocation. With vertical partitioning, the Employee table can be split up into two separate tables, VerticalShard1 and VerticalShard2. In VerticalShard1, only the EmpID, EmpName, and EmpDOB columns are displayed. This is beneficial because it can help the business focus on each employee’s birthday rather than their gender or work location. On the other hand, the VerticalShard2 table only consists of EmpID, EmpGender, and EmpWorkLocation columns. This can be helpful if the business wants to focus on gender and work location. Furthermore, the vertical partitioning technique gives businesses the opportunity to improve their database system and performance by reducing the number of columns. Sometimes, businesses may have to protect data stored in a specific column, so businesses might vertically partition a table to limit a user’s access.

**Explain how and why data replication should be addressed in this environment. What replication strategies are available and how do they work? How do data replication/partitioning impact reliability, expandability, communications overhead cost, manageability, and data consistency? - Zaid**

Data replication is the process of creating and storing multiple copies of the same data in different locations. In a distributed environment, data replication is essential to ensure data availability, reliability, and scalability. Data replication should be addressed in this environment because a) it ensures that data is available even if one or more nodes fail or become unavailable b) it can improve performance by distributing the load across multiple nodes, and c) Data replication can be used to create backups that can be used in the event of a disaster (Pandey, 2023, pp. 1-6).

There are several replication strategies available, including:

1. Master-slave replication: In this strategy, one node (the master) is responsible for all write operations, and the other nodes (the slaves) replicate the data from the master. This strategy is easy to implement but can create a single point of failure.
2. Master-master replication: In this strategy, multiple nodes act as masters and can perform both read and write operations. This strategy can improve availability and scalability but can be complex to implement.
3. Multi-master replication: This strategy is similar to master-master replication, but nodes can be added or removed dynamically. This strategy is suitable for environments where the number of nodes can change frequently (Goel & Buyya, 2007, p.211).

Data replication can improve reliability by ensuring that data is available even if one or more nodes fail. Additionally, it can improve expandability by distributing the load across multiple nodes, allowing for more data to be stored and processed. Data replication can increase communications overhead cost by requiring additional network bandwidth to synchronize data between nodes. Manageability complexity can also be increased through data replication because multiple copies of data must be managed, and data consistency must be maintained. Data consistency: Data replication can impact data consistency because changes made to one copy of data must be propagated to all other copies. This can result in data conflicts if changes are made concurrently on different nodes (*Introducing*…, n.d., pp. 1-6)

**Explain how and why data partitioning and replication impact the creation of IT and technology infrastructures, creation of plans and schedules, measuring performance, and ensuring quality? LUKE**

The impact that data partitioning and replication have on IT and the corresponding infrastructures is vital to the success of organizations. It has become more known that “partitioning a system can provide several benefits. First of all, the system becomes fault-tolerant to failures. Failure of a single machine means only the affected partitions are impacted and users accessing data from the other partitions can continue using the system” (Partitioning and Replication, n.d.). This makes things much easier for IT personnel because less work would be required if only one machine was down as opposed to the entire system. Creating and planning schedules would also become a lot simpler after partitioning the data since the dates can be used as a means for separating the data accordingly. Partitioning is also a much cheaper option as opposed to keeping everything on one machine, which would take more time and energy to analyze. In the end, a partitioned system is the most efficient way to store and analyze data if there are multiple partitions with smaller data sets.

**References**

Custer, C. (2023, February 7). *What is data partitioning, and how to do it right*. Cockroach Labs. Retrieved April 30, 2023, from <https://www.cockroachlabs.com/blog/what-is-data-partitioning-and-how-to-do-it-right/#vertical-partitioning>

Goel, S., & Buyya, R. (2007). Data replication strategies in wide-area distributed systems. In Enterprise service computing: from concept to deployment (pp. 211-241). IGI Global

*Introducing chatgpt*. Introducing ChatGPT. (n.d.). Retrieved from <https://openai.com/blog/chatgpt>

Medic, M. (2020). Database table partitioning in SQL Server. SQL Shack - Articles About Database Auditing, Server Performance, Data Recovery, and More.   
 https://www.sqlshack.com/database-table-partitioning-sql-server/ ,

Pandey, H. (2023, January 26). Data Replication in DBMS. GeeksforGeeks. Retrieved from https://www.geeksforgeeks.org/data-replication-in-dbms/

*Partitioning and Replication: Benefits & Challenges*. A curious mind. (n.d.). Retrieved April 26, 2023, from <https://dimosr.github.io/partitioning-and-replication/>