REPORTE DE PRÁCTICA NO. 3

NOMBRE DE LA PRÁCTICA: Álgebra relacional y SQL (1)

ALUMNO:

José Eduardo Valles Aguilera

1. Introducción

Diseñar y hacer consultas en una base de datos a través de sentencias derivadas del álgebra relacional

2. Marco teórico

Álgebra relacional

En la teoría de bases de datos, el álgebra relacional es una teoría que utiliza estructuras algebraicas con una semántica bien fundamentada para modelar datos y definir consultas sobre ellos. La teoría fue presentada por Edgar F. Codd. El propósito principal del álgebra relacional es definir operadores que transformen una o más relaciones de entrada en una relación de salida. Dado que estos operadores aceptan relaciones como entrada y producen relaciones como salida, pueden combinarse y usarse para expresar consultas potencialmente complejas que transforman potencialmente muchas relaciones de entrada (cuyos datos se almacenan en la base de datos) en una sola relación de salida (los resultados de la consulta) .
colaboradores de Wikipedia. (2024, 7 octubre). Álgebra relacional. Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/

SQL

SQL (por sus siglas en inglés Structured Query Language; en español lenguaje de consulta estructurada) es un lenguaje específico de dominio, diseñado para administrar, y recuperar información de sistemas de gestión de bases de datos relacionales. Una de sus principales características es el manejo del álgebra y el cálculo relacional para efectuar consultas con el fin de recuperar, de forma sencilla, información de bases de datos, así como realizar cambios en ellas.
colaboradores de Wikipedia. (2024b, noviembre 11). SQL. Wikipedia, la Enciclopedia Libre. https://es.wikipedia.org/wiki/SQL

MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: licencia pública general/licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo, y una de las más populares en general junto a Oracle y Microsoft SQL Server, todo para entornos de desarrollo web.
Para esta práctica, se han utilizado las siguientes sentencias para crear y consultar una base de datos que almacena información sobre los empleados de una empresa y los bonos que obtienen en determinada fecha

CREATE DATABASE Algebrarelacional;
USE Algebrarelacional;
CREATE TABLE Employee (
$Employee_IDINTPRIMARYKEY$,
$First_nameVARCHAR(50)$,
$Last_nameVARCHAR(50)$,
$SalaryDECIMAL(10,2)$,
$Joining_dateDATE$,
$DepartmentVARCHAR(50)$
);

$CREATETABLEReward($
$Employee_reef_IDINTPRIMARYKEY$,
$Reward_DateDATE$,
$Reward_AmountDECIMAL(10,2)$
);

$INSERTINTOEmployeeVALUES(1,'Bob','Kinto',1000000.00,'2019-01-20','Finance');$
$INSERTINTOEmployeeVALUES(2,'Jerry','Kansxo',6000000.00,'2019-01-15','IT');$
$INSERTINTOEmployeeVALUES(3,'Philip','Jose',8900000.00,'2019-02-05','Banking');$
$INSERTINTOEmployeeVALUES(4,'John','Abraham',2000000.00,'2019-02-25','Insurance');$

$INSERTINTOEmployeeVALUES(5,'Michael','Mathew',2200000.00,'2019-02-28','Finance');$
$INSERTINTOEmployeeVALUES(6,'Alex','Chreketo',4000000.00,'2019-05-10','IT');$
$INSERTINTOEmployeeVALUES(7,'Yohan','Soso',1230000.00,'2019-06-10','Banking');$

$INSERTINTORewardVALUES(1,'2019-05-11',1000.00);$
$INSERTINTORewardVALUES(2,'2019-02-15',5000.00);$
$INSERTINTORewardVALUES(3,'2019-04-22',2000.00);$
$INSERTINTORewardVALUES(4,'2019-06-20',8000.00);$

$--5.Obtenertodoslosempleados$
$SELECT*FROMEmployee;$

$--6.Obtenerelprimernombreyapellidodetodoslosempleados$
$SELECTFirst_name,Last_nameFROMEmployee;$

$--7.ObtenerlacolumnaFirst_nameconalias$
$SELECTFirst_nameAS"Nombredeempleado"FROMEmployee;$

$--8.ObtenerLast_nameenminúsculas$
$SELECTLOWER(Last_name)FROMEmployee;$

$--9.ObtenerLast_nameenmayúsculas$
$SELECTUPPER(Last_name)FROMEmployee;$

$--10.Obtenernombresúnicosdelacolumna Department$
$SELECTDISTINCTDepartmentFROMEmployee;$

$--11.Obtenerlosprimeros4caracteresdeFirst_name$
$SELECTSUBSTRING(First_name,1,4)FROMEmployee;$

$--12.Posiciónde"h"enFirst_name="Jhon"$
$SELECTLOCATE('h',First_name)FROMEmployeeWHEREFirst_name='Jhon';$

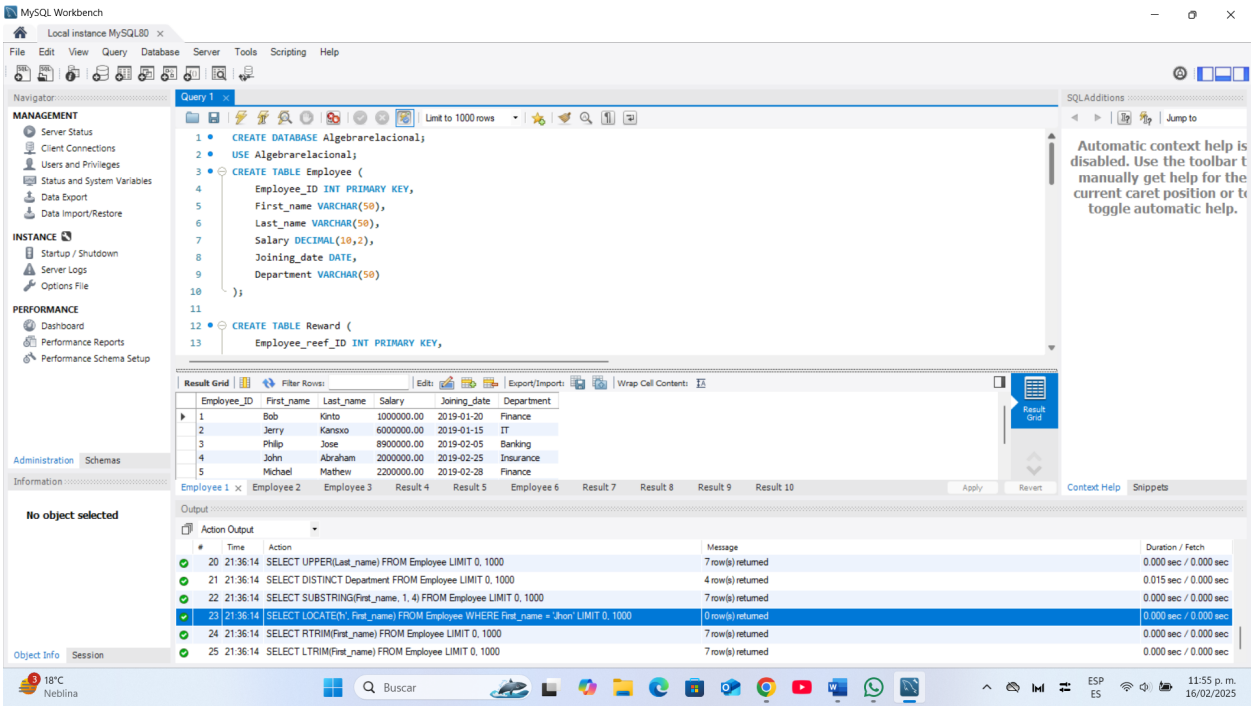
$--13.Eliminarespaciosenblancodeladerechade First_name$
$SELECTRTRIM(First_name)FROMEmployee;$

$--14.Eliminarespaciosenblancodelaizquierda deFirst_name$
$SELECTLTRIM(First_name)FROMEmployee;$

4. Desarrollo

Análisis de requisitos

EJERCICIOS.
1. Escribe la sintaxis para crear la tabla “Employee”.
2. Escribe la sintaxis para insertar 7 registros (de la imagen) a la tabla “Employee”.
3. Escribe la sintaxis para crear la tabla “Reward”.
4. Escribe la sintaxis para insertar 4 registros (en la imagen) a la tabla “Reward”.
5. Obtener todos los empleados.
6. Obtener el primer nombre y apellido de todos los empleados.
7. Obtener todos los valores de la columna $\backslash Firstname$” usando el alias “Nombre de empleado”.
8. Obtener todos los valores de la columna $'Last_name'$ in minsculas.
9. Obtener todos los valores de la columna $\backslash Last_name$” en mayúsculas.
10. Obtener los nombre únicos de la columna $\backslash Departament$”.
11. Obtener los primeros 4 caracteres de todos los valors de la columna $\backslash First_name$”.
12. Obtener la posición de la letra “h” en el nombre del empleado con $First_name = \backslash Jhon$”.
13. Obtener todos los valores de la columna $\backslash First_name$” después de remover los espacios en blanco de la derecha.
14. Obtener todos los valores de la columna $\backslash First_name$” después de remover los espacios en blanco de la izquierda.



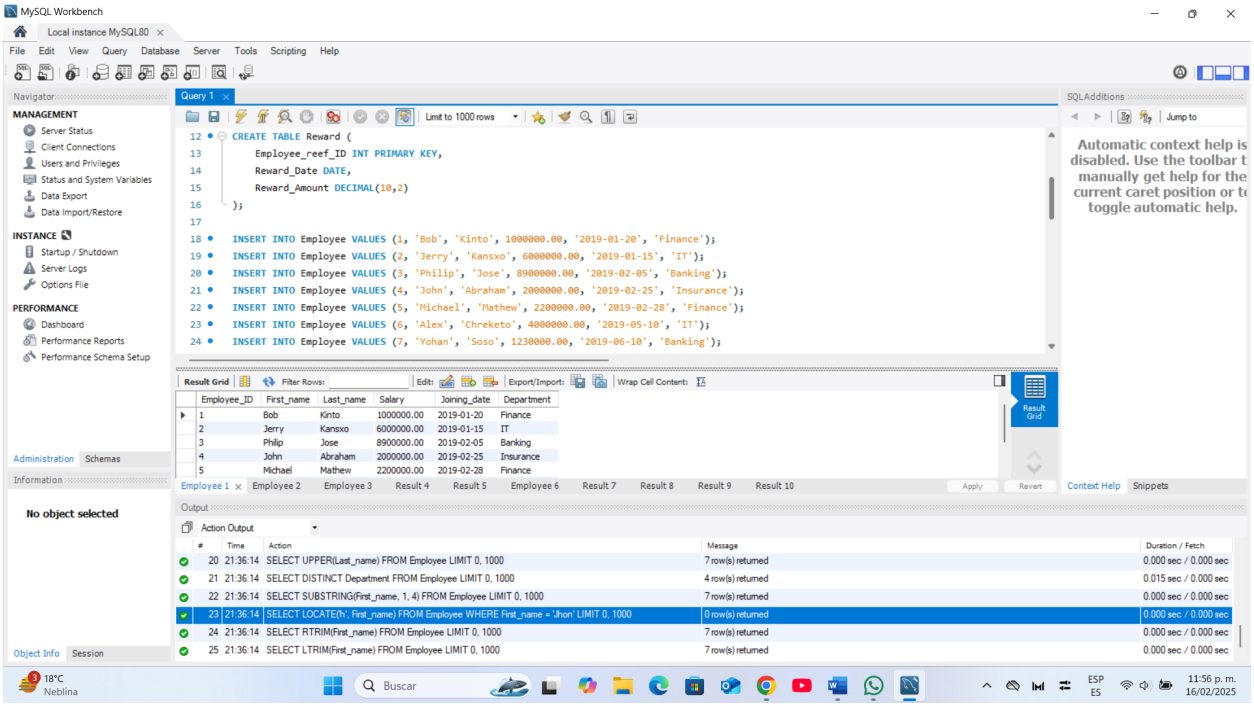


Figure 1: Creación de la base de datos y las tablas

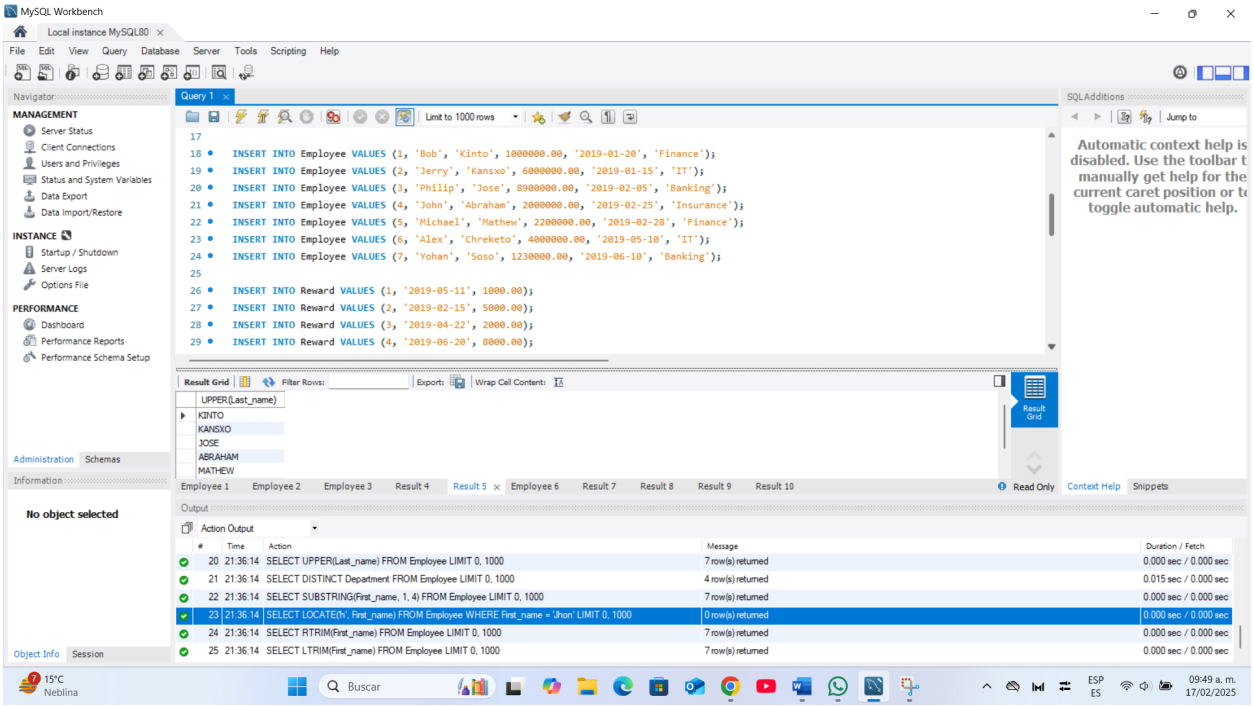


Figure 2: Inserción de registros

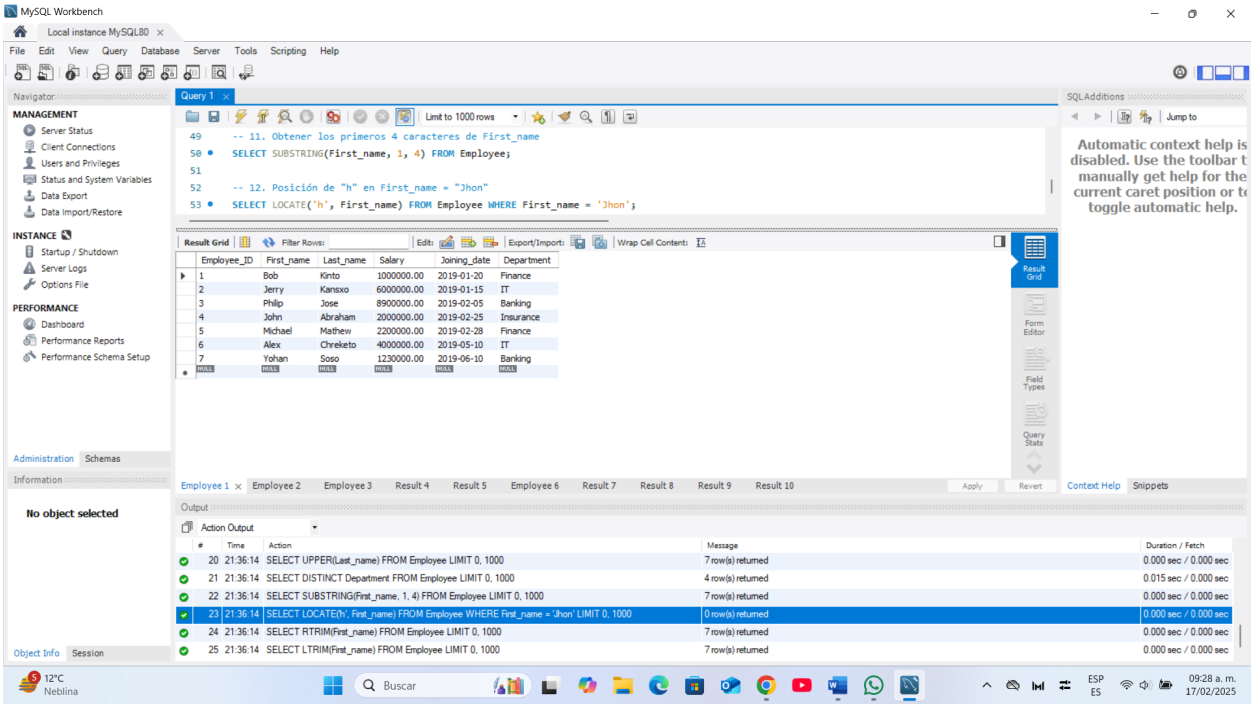


Figure 3: Obtener todos los empleados

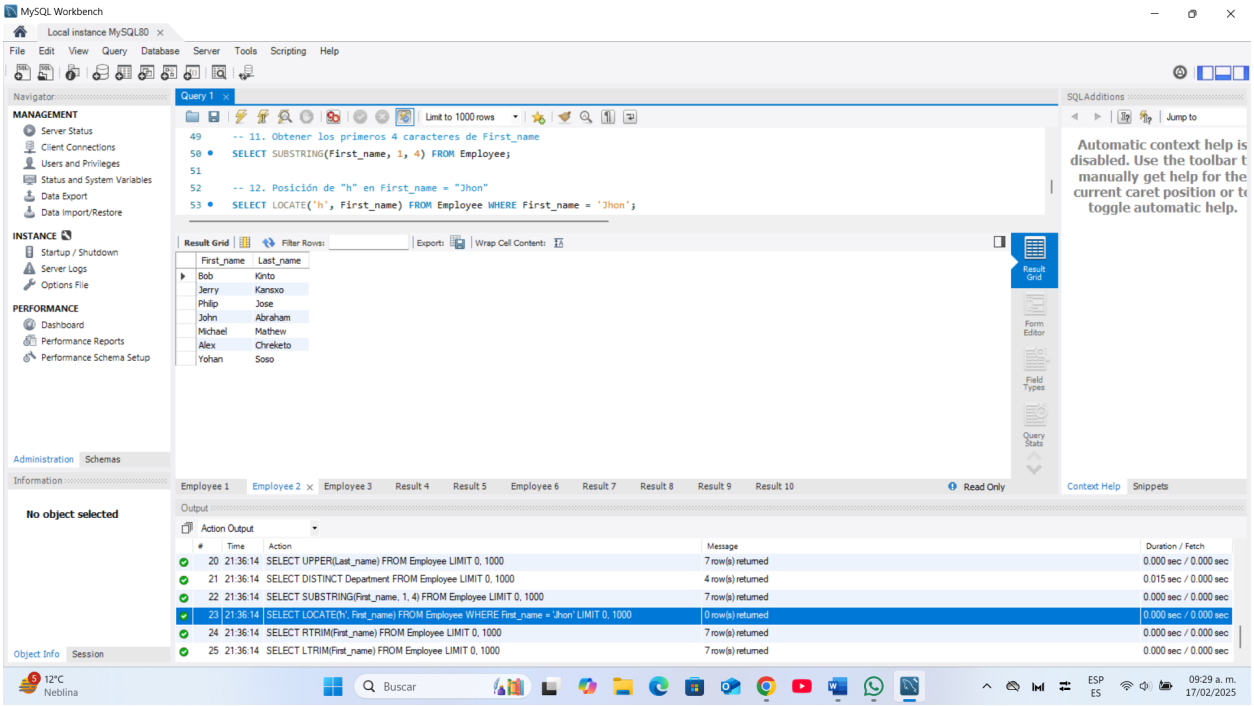


Figure 4: Obtener el primer nombre y apellido de todos los empleados

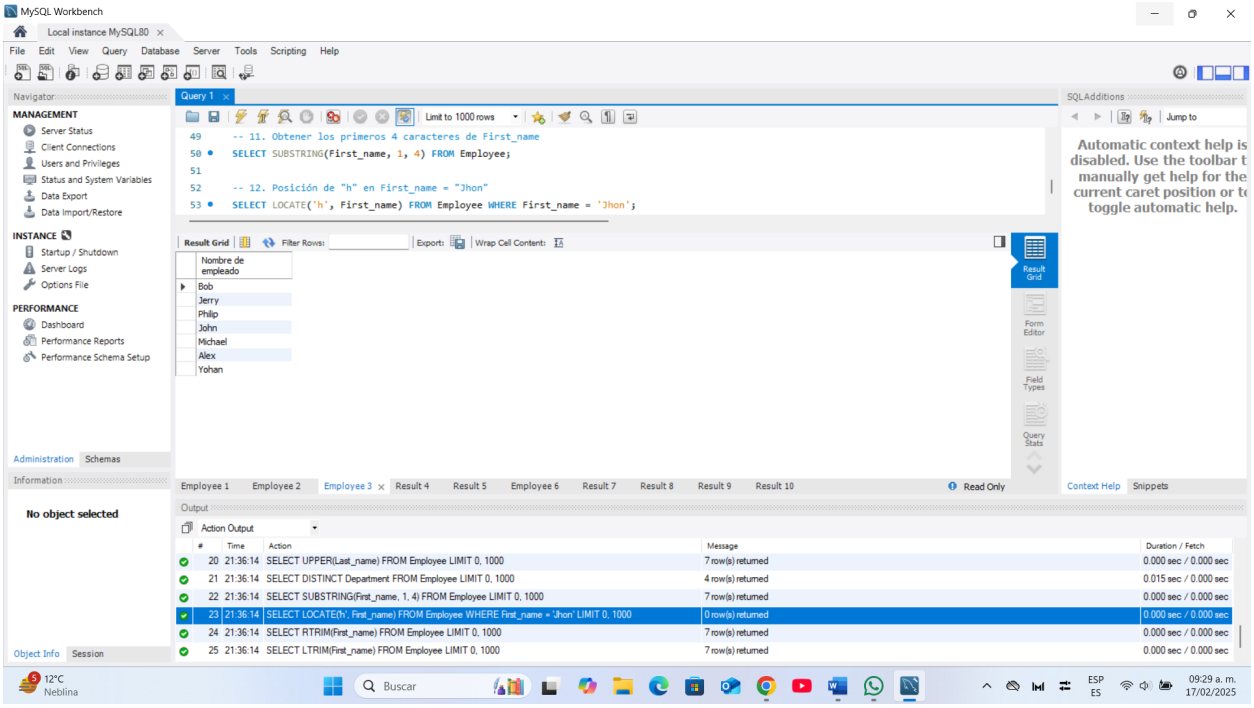


Figure 5: Obtener la columna Firstname con alias

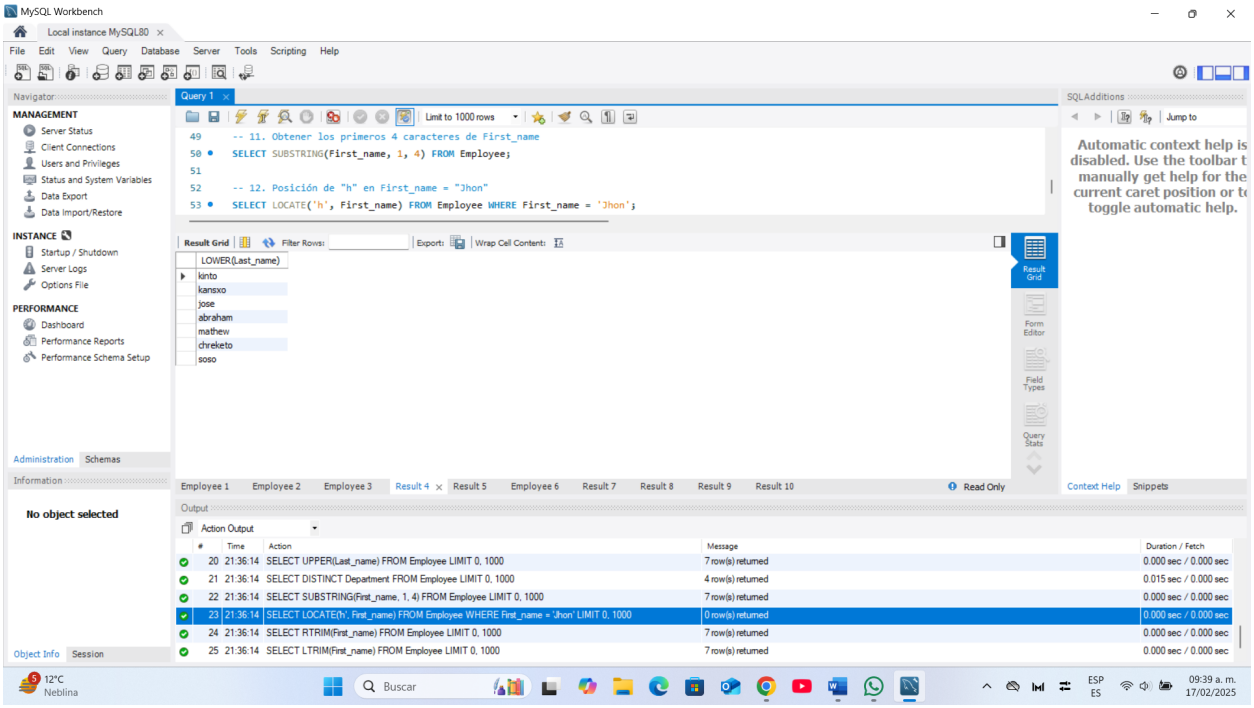


Figure 6: Obtener Lastname en minúsculas

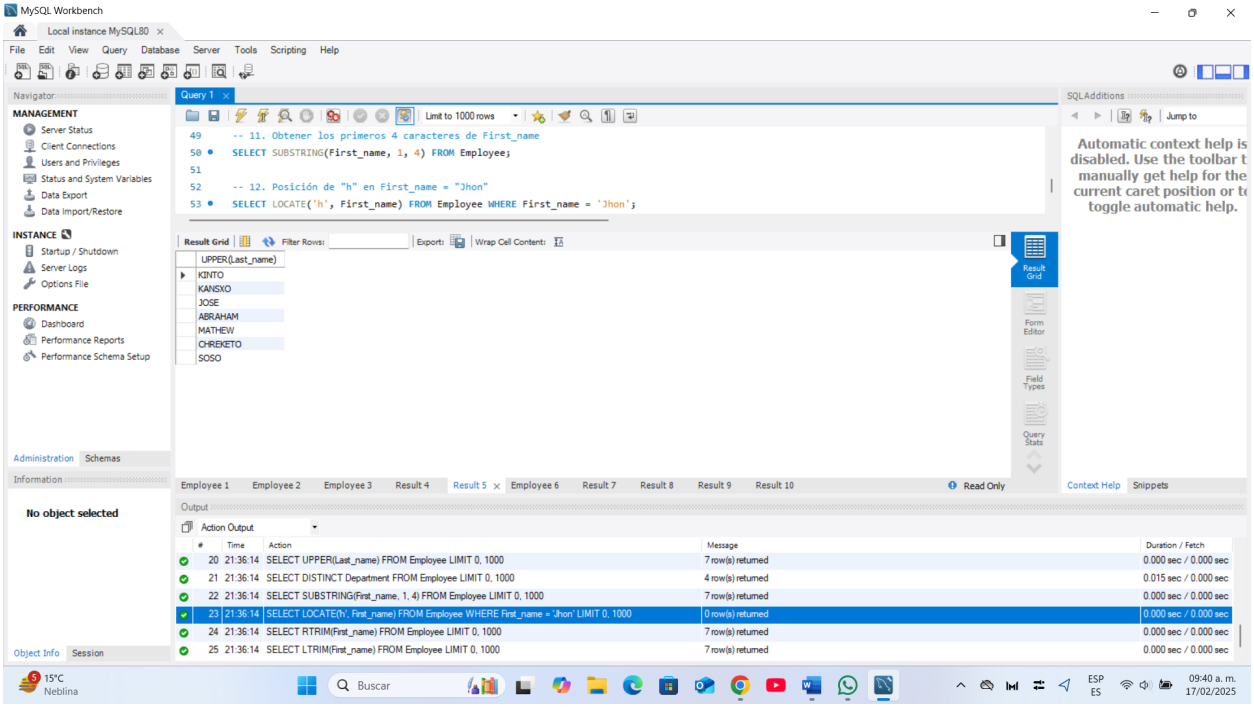


Figure 7: Obtener Lastname en mayúsculas

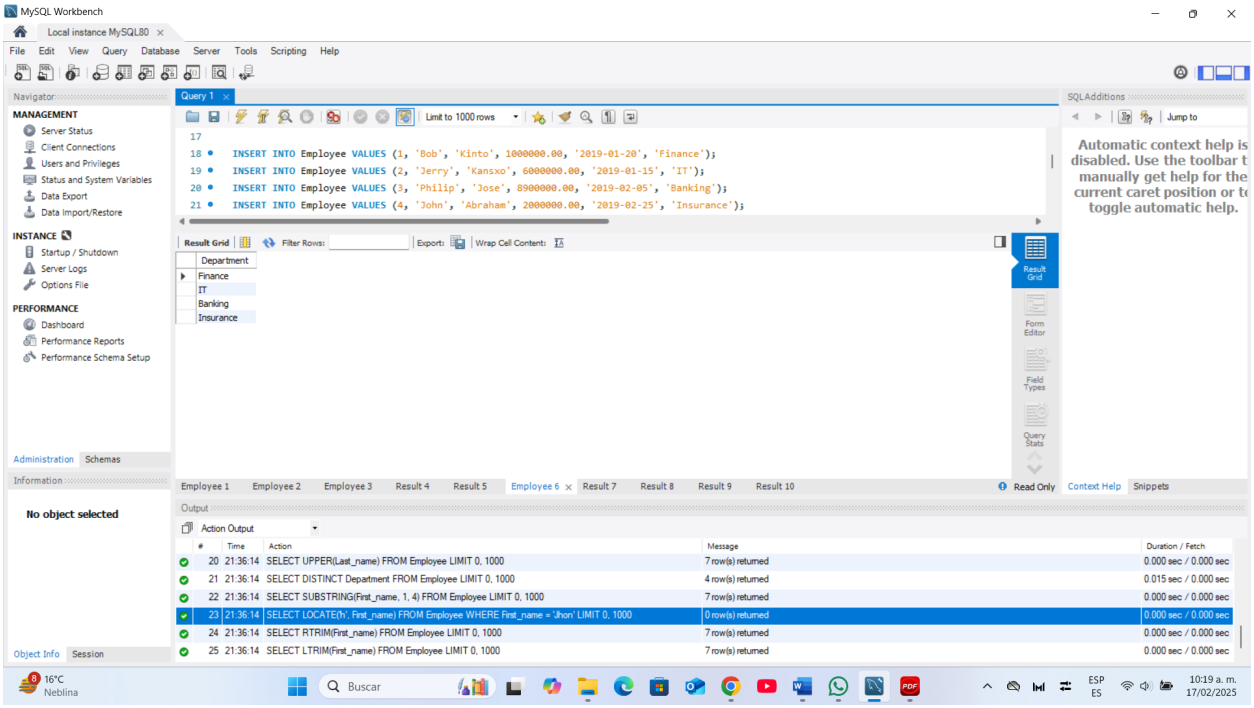


Figure 8: Obtener los nombres únicos de la columna “Departament”

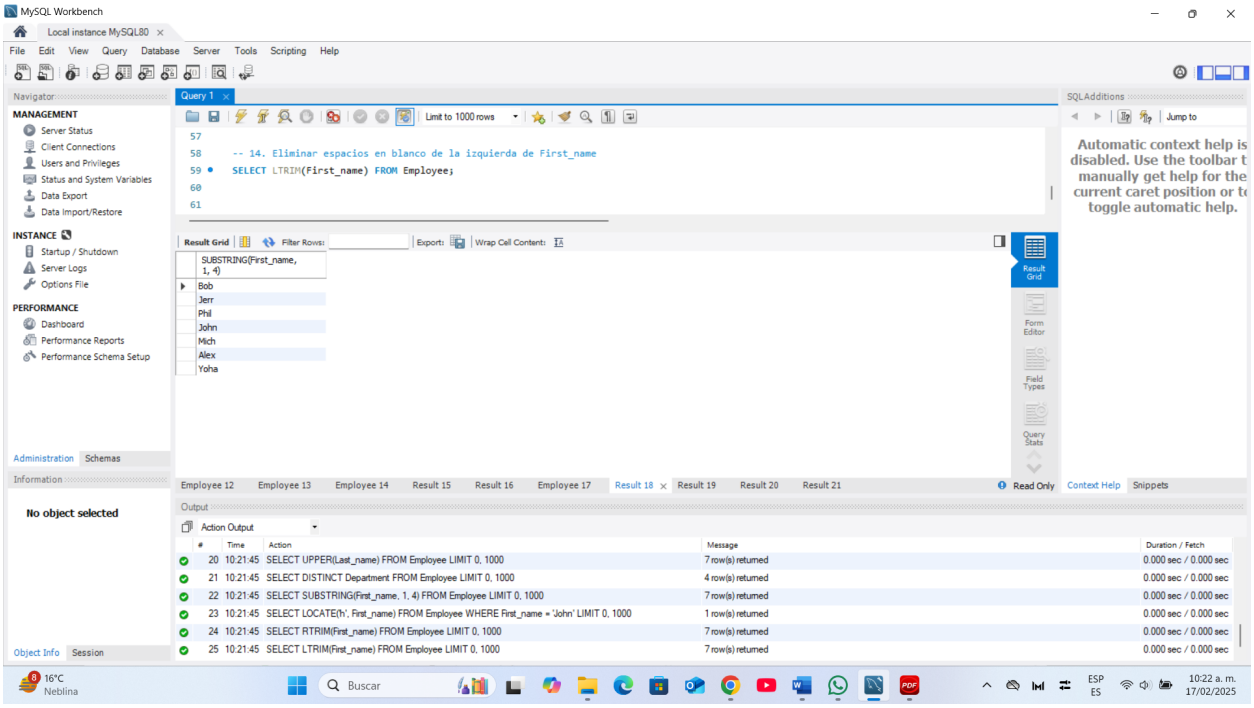


Figure 9: Obtener los primeros 4 caracteres de todos los valores de la columna “Firstname”

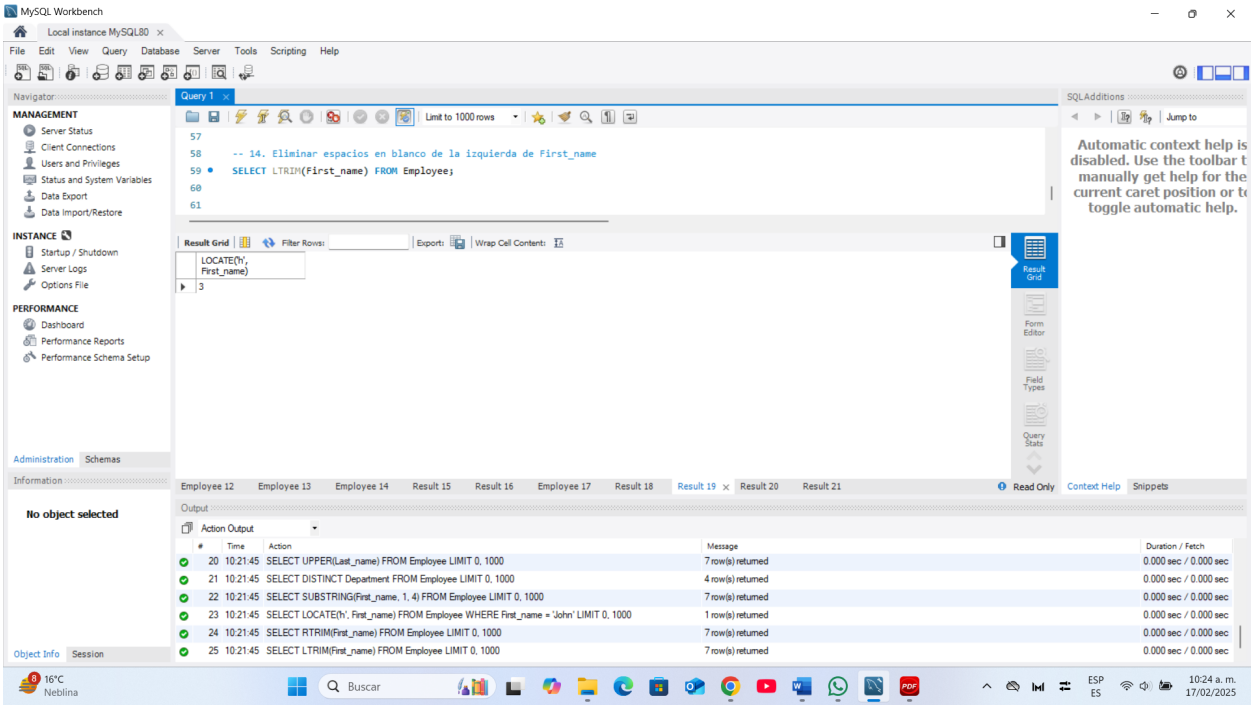


Figure 10: Obtener la posición de la letra “h” en el nombre del empleado con Firstname=“John”

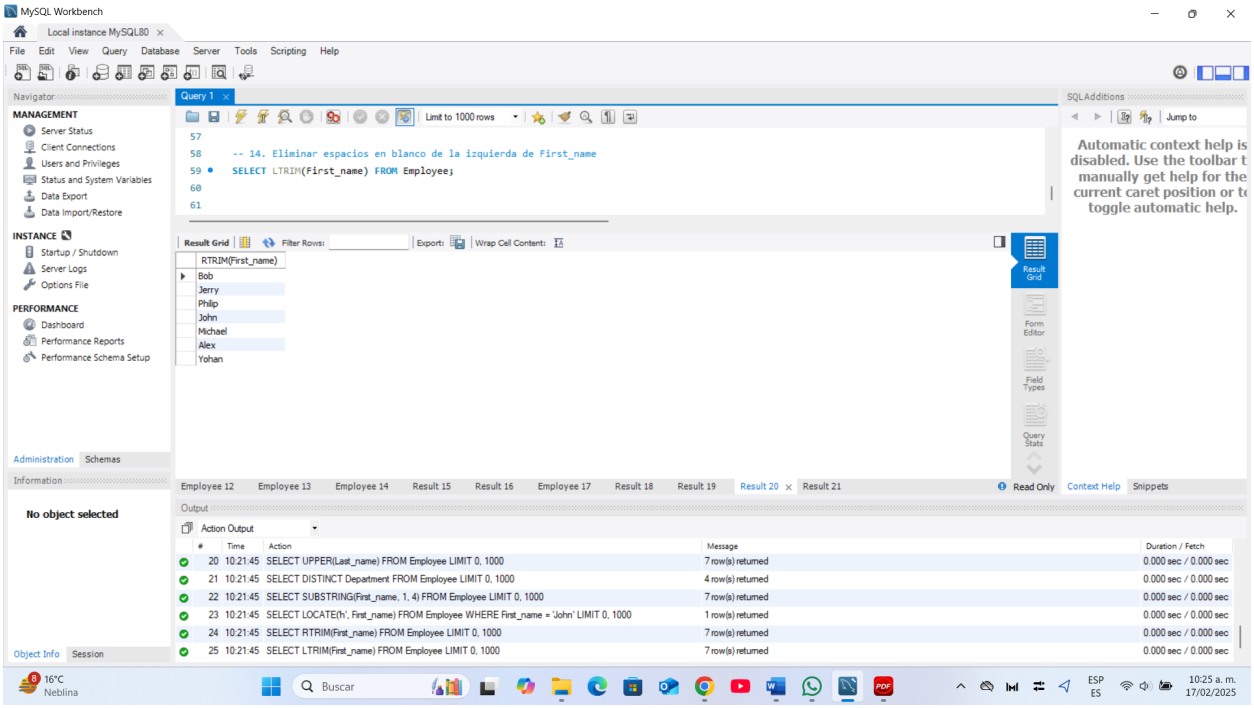


Figure 11: . Obtener todos los valores de la columna “Firstname” después de remover los espacios en blanco de la derecha.

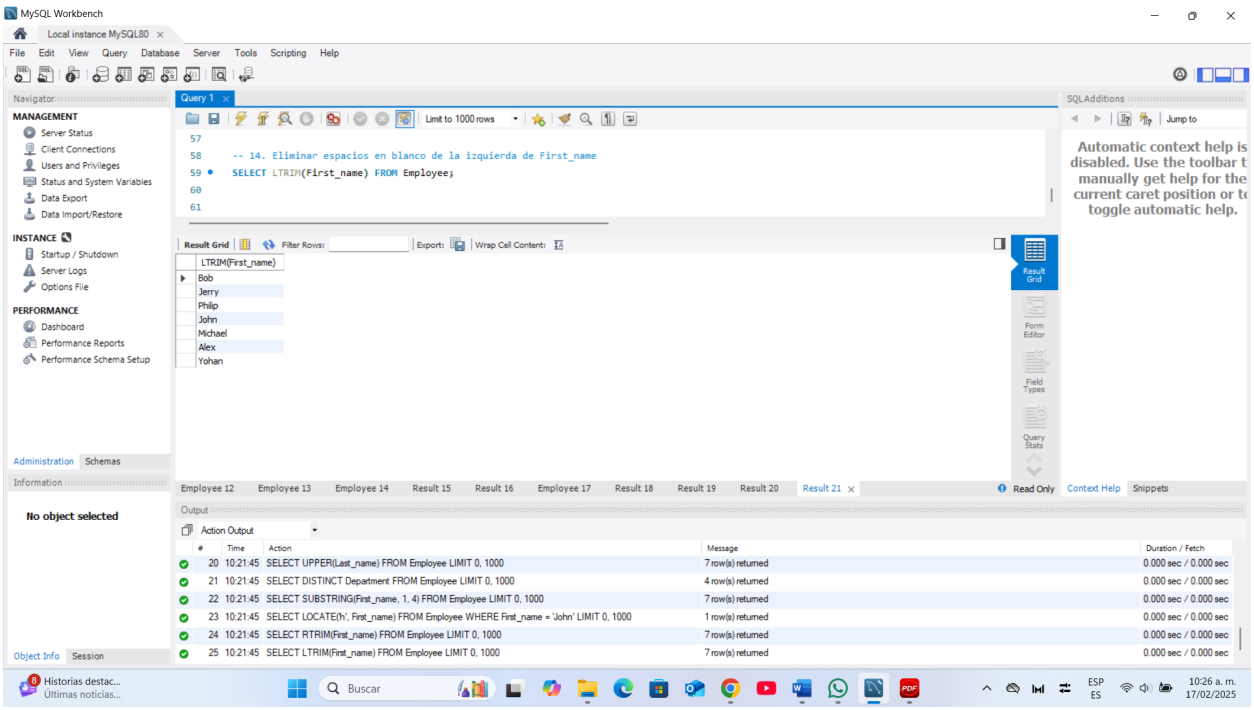


Figure 12: Obtener todos los valores de la columna “Firstname” después de remover los espacios en blanco de la izquierda

Inserción en Álgebra relacional

- Employee←Employee(1,'Bob','Kinto',1000000.00,'2019-01-20','Finance')
- Employee←Employee(2,'Jerry','Kansxo',6000000.00,'2019-01-15','IT')
- Employee←Employee(3,'Philip','Jose',8900000.00,'2019-02-05','Banking')
- Employee←Employee(4,'John','Abraham',2000000.00,'2019-02-25','Insurance')
- Employee←Employee(5,'Michael','Mathew',2200000.00,'2019-02-28','Finance')
- Employee←Employee(6,'Alex','Chreketo',4000000.00,'2019-05-10','IT')
- Employee←Employee(7,'Yohan','Soso',1230000.00,'2019-06-10','Banking')

Consultas en Álgebra relacional

- $\pi * (Employee)$

- $\pi Firstname, Lastname(Employee)$
- $\rho"Nombredeempleado"/First_name(First_name(Employee))$
- $\pi lower(Last_name)(Employee)$
- $\pi upper(Last_name)(Employee)$
- $\delta Department(Department(Employee))$
- $\pi substring(First_name, 1, 4)(Employee)$
- $\pi locate('h', First_name)(First_name =' John'(Employee))$
- $\pi rtrim(First_name)(Employee)$
- $\pi ltrim(First_name)(Employee)$

5. Conclusiones

Aunque ya dominaba lo básico de crear una base de datos, tablas y hacer consultas, no tenía muy claro el Álgebra Relacional. Todavía no siento que lo domine al 100 por ciento, pero ahora entiendo un poco más de los conceptos básicos. Seguiré buscando información relacionada al tema para entender mejor.

Referencias Bibliográficas

References

[1] Bases de Datos. (2020, 25 julio). 01 Álgebra Relacional — Conceptos y operaciones [Vídeo]. YouTube. https://www.youtube.com/watch?v=vh0KMMsTGQ0