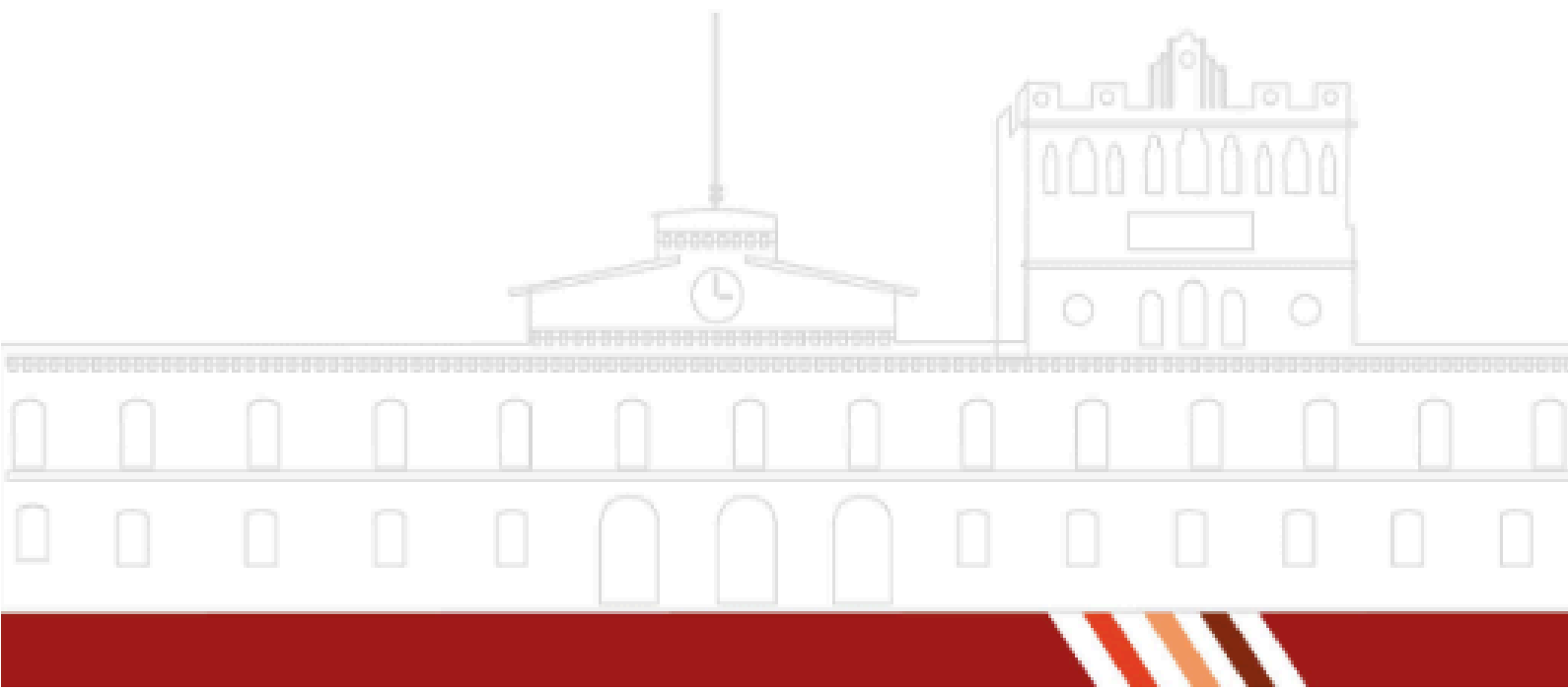


# REPORTE DE PRÁCTICA NO. 1.3

**ALUMNO:** Cesar Rodriguez Garcia  
Dr. Eduardo Cornejo-Velázquez



# 1. Introducción

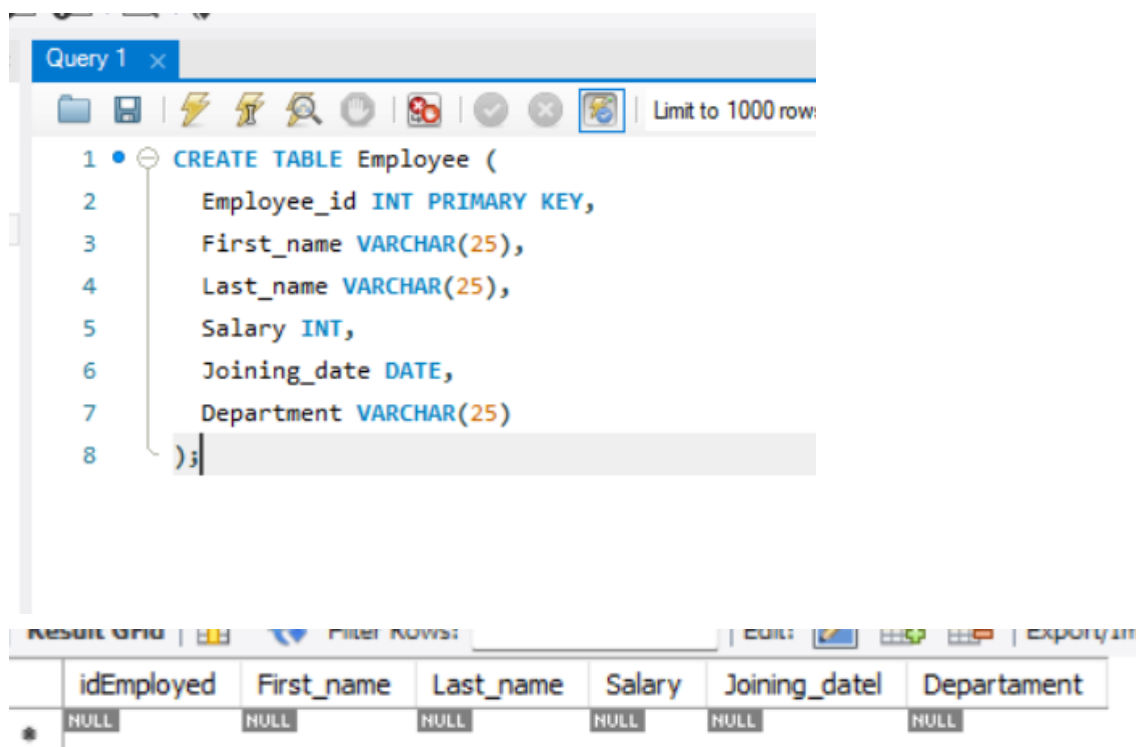
Employee y Reward y el objetivo es que el usuario escriba sentencias \textbf{SQL} para crear las tablas, insertar datos y realizar consultas para obtener información específica. Los ejercicios cubren desde la creación de esquemas y la inserción de registros hasta la manipulación y consulta de datos, como la selección de columnas, el uso de alias y la modificación de cadenas de texto. La resolución de cada ejercicio requiere una captura de pantalla que muestre la sentencia SQL y el resultado de su ejecución.

## 2. Marco Teórico

El álgebra relacional es un conjunto de operaciones que se utilizan para manipular datos en un modelo de bases de datos relacional. Es fundamental para entender cómo funcionan las bases de datos y es la base teórica del lenguaje de consulta SQL (Structured Query Language). Las operaciones del álgebra relacional toman una o más relaciones (tablas) como entrada y producen una nueva relación como resultado. Las operaciones se pueden clasificar en dos grandes grupos: operaciones de conjuntos y operaciones relacionales.

## 3. Desarrollo

- Escribe la sintaxis para crear la tabla “Employee”.



- Escribe la sintaxis para insertar 7 registros (de la imagen) a la tabla “Employee”.

```

2 • INSERT INTO Employee (Employee_id, First_name, Last_name, Salary, Joining_date, Department)
3 VALUES
4 (1, 'Bob', 'Kinto', 1000000, '2019-01-20', 'Finance'),
5 (2, 'Jerry', 'Kansxo', 6000000, '2019-01-15', 'IT'),
6 (3, 'Philip', 'Jose', 8900000, '2019-02-05', 'Banking'),
7 (4, 'John', 'Abraham', 2000000, '2019-02-25', 'Insurance'),
8 (5, 'Michael', 'Mathew', 2200000, '2019-02-28', 'Finance'),
9 (6, 'Alex', 'chreketo', 4000000, '2019-05-10', 'IT'),
10 (7, 'Yohan', 'Soso', 1230000, '2019-06-20', 'Banking');

```

Result Grid

	idEmployed	First_name	Last_name	Salary	Joining_date	Department
1	1	Bob	Kinto	1000...	2019-01-20	Finance
2	2	Jerry	Kansxo	6000...	2019-01-15	IT
3	3	Phillip	Jose	8900...	2019-02-05	Banking
4	4	John	Abraham	2000000	2019-02-25	Insurance
5	5	Michael	Mathew	2200...	2019-02-28	Finance
6	6	Alex	Chreketo	4000...	2019-05-10	IT
7	7	Yohan	Soso	1230...	2019-06-20	Banking
•	HULL	HULL	HULL	HULL	HULL	HULL

3. Escribe la sintaxis para crear la tabla “Reward”

```

CREATE TABLE Reward (
    Employee_id INT,
    Date_reward DATE,
    Amount DECIMAL(10,2),
    FOREIGN KEY (Employee_id) REFERENCES Employee(id)
);

```

	Employee_id	Date_reward	Amount
--	-------------	-------------	--------

4. Escribe la sintaxis para insertar 4 registros (en la imagen) a la tabla “Reward”.

```

10 (7, 'Yohan', 'Soso', 1230000, '2019-06-20', 'Banking');
11
12 • INSERT INTO Reward (Employee_ref_id, date_reward, amount)
13 VALUES
14 (1, '2019-05-11', 1000),
15 (2, '2019-02-15', 5000),
16 (3, '2019-04-22', 2000),
17 (1, '2019-06-20', 8000);

```

	Employee_id	Date_reward	Amount
▶	1	2019-05-11	1000.00
	3	2022-01-20	1500.00
	5	2022-03-10	1200.00
	7	2021-07-05	800.00

5. Obtener todos los empleados.

• `SELECT * FROM Employee;`

	id	First_name	Last_name	Department	Salary	Joining_date
▶	1	Jhon	Doe	HR	50000.00	2020-01-10
	2	Jane	Smith	IT	60000.00	2019-03-15
	3	Michael	Johnson	Finance	70000.00	2021-06-20
	4	Emily	Davis	IT	55000.00	2018-11-25
	5	William	Brown	HR	52000.00	2022-05-12
	6	Sophia	Wilson	Finance	75000.00	2017-08-30
	7	Daniel	Miller	IT	58000.00	2019-09-01

6. Obtener el primer nombre y apellido de todos los empleados.

21 • `SELECT First_name, Last_name FROM Employee;`

Result Grid	Filter Rows:	Export:	Write
First_name	Last_name		
▶ Jhon	Doe		
Jane	Smith		
Michael	Johnson		
Emily	Davis		
William	Brown		
Sophia	Wilson		
Daniel	Miller		

7. Obtener todos los valores de la columna "First\_name" usando el alias "Nombre de empleado".

23 • `SELECT First_name AS "Nombre de empleado" FROM Employee;`

Result Grid	Filter Rows:
Nombre de empleado	
▶ Jhon	
Jane	
Michael	
Emily	
William	
Sophia	

8. Obtener todos los valores de la columna "Last\_name" en minúsculas.

```
SELECT LOWER>Last_name) FROM Employee;
```

Result Grid | Filter Rows: | Export: | Wrap C

LOWER>Last_name)
doe
smith
johnson
davis
brown
wilson
miller

9. Obtener todos los valores de la columna "Last\_name" en mayúsculas.

```
27 • SELECT LOWER>Last_name) FROM Employee;  
28 • SELECT UPPER>Last_name) FROM Employee;
```

upper>Last_name)
DOE
SMITH
JOHNSON
DAVIS
BROWN
WILSON
MILLER

10. Obtener los nombre únicos de la columna "Departament".

```
30 • SELECT DISTINCT Departement FROM Employee;
```

Department
HR
IT
Finance

11. Obtener los primeros 4 caracteres de todos los valores de la columna "First\_name".

```
32  
33 • SELECT SUBSTRING(First_name, 1, 4) FROM Employee;
```

## 4. Conclusión

Estos ejercicios proporcionan una base práctica y sólida para cualquiera que comience con SQL. Al completarlos, has adquirido una comprensión fundamental de cómo definir, llenar y, lo más importante, consultar datos en una base de datos relacional. Esta es una excelente base para abordar tareas de bases de datos más complejas..

## 5. Bibliografía

Estado, del. (2025). *Universidad Autónoma del Estado de Hidalgo :: UAEH*.

Universidad Autónoma Del Estado de Hidalgo.

[https://lms3.uaeh.edu.mx/lms/JD31025/Licenciatura/pluginfile.php/244390/mod\\_resource/content/4/1.Algebra\\_Relacional.pdf](https://lms3.uaeh.edu.mx/lms/JD31025/Licenciatura/pluginfile.php/244390/mod_resource/content/4/1.Algebra_Relacional.pdf)