

REPORTE DE PRÁCTICA NO. 1.3

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

DOCENTE: Dr. Eduardo Cornejo-Velázquez

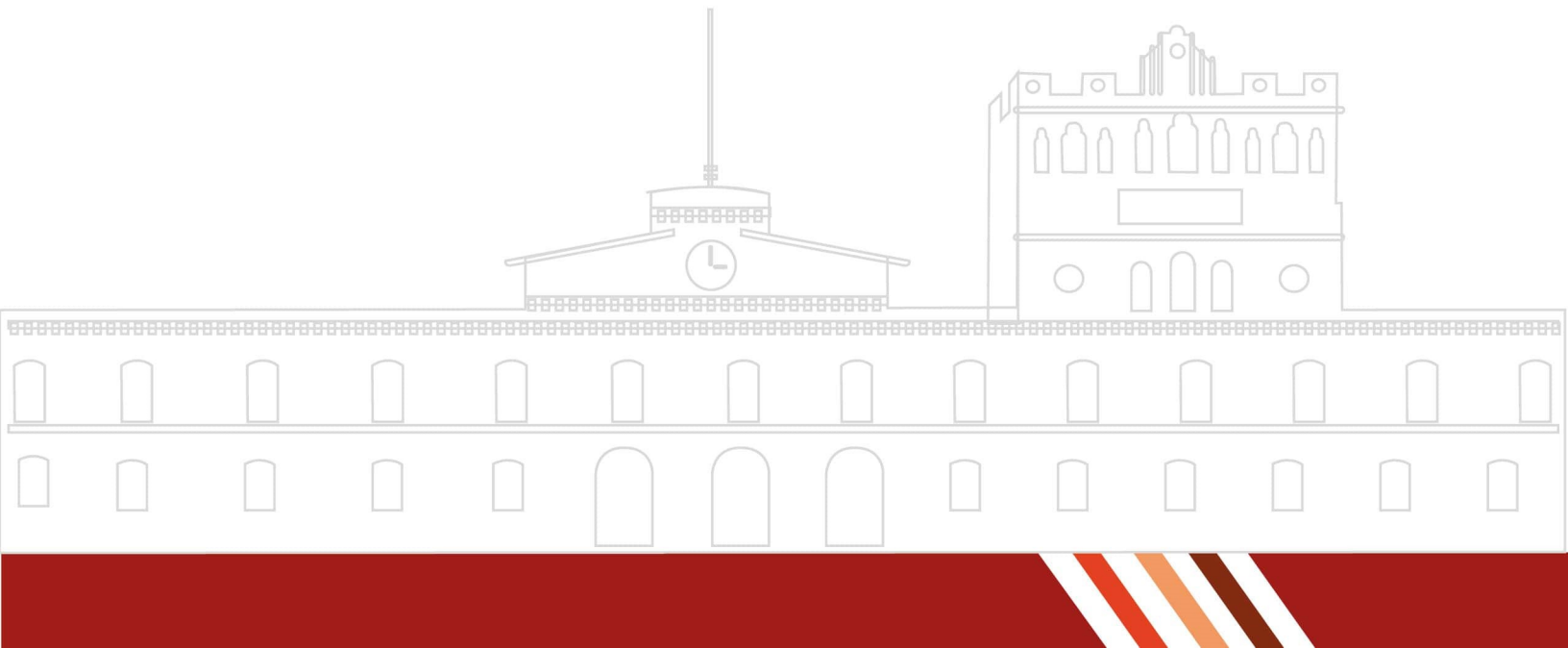
ALUMNO: Ramirez Suarez Gerardo

SEMESTRE y GRUPO: 6°-2

ASIGNATURA: Base de Datos Distribuida

No. CUENTA: 340097

LICENCIATURA EN CIENCIAS COMPUTACIONALES



1. Introducción

Esta práctica esta empleada para que nosotros los estudiantes se relaciones con las sentencias SQL, ya que este se es util para una base de datos. Para la creación, manipulación y consultas de BD.

Los ejercicios se proporciona dos tablas, para crearlas en MySQL y insertar datos a ellas y tambien ejecución de consultas que nos permiten extraer, modificar y formaterar información.

2. Herramientas empleadas

Describir qué herramientas se han utilizado...

1. ERD Plus. ERDPlus es una herramienta en línea gratuita para la creación de diagramas entidad-relación (ERD), esquemas relacionales y esquemas estrella. Se utiliza principalmente para visualizar y diseñar bases de datos, facilitando la creación de modelos de datos que representan las relaciones entre diferentes entidades.
2. MySQL Server. MySQL Command es una interfaz de línea de comandos para interactuar con bases de datos MySQL. Permite a los usuarios ejecutar comandos SQL directamente, gestionar bases de datos, realizar consultas, insertar datos, y mucho más. Es una herramienta esencial para administradores de bases de datos y desarrolladores.
3. LaTeX. LaTeX es un sistema de preparación de documentos que se utiliza para la creación de documentos de alta calidad tipográfica, especialmente en el ámbito académico y científico. Se utiliza para escribir artículos, tesis, libros y otros documentos que requieren una presentación profesional y precisa de fórmulas matemáticas, referencias bibliográficas y otros elementos complejos.

3. Desarrollo

Sentencias SQL

Presentar las sentencias para crear la base de datos y tablas. Además incluir las sentencias para insertar registros.

En el Listado 1 se presenta la sentencia SQL para crear la base de datos competencia.

Listing 1: Crear base de datos competencia.

CREATE DATABASE competencia .

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

```
CREATE TABLE Employee (  
    Employee_id INT PRIMARY KEY,  
    First_name VARCHAR(50),  
    Last_name VARCHAR(50),  
    Salary INT(8),  
    Joining_date DATE,  
    Department VARCHAR(50)  
);
```

Figure 1: Creación de Employee

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

```
mysql> INSERT INTO Employee (Employee_id, First_name, Last_name, Salary, Joining_date, Department) VALUES  
-> (1, 'Bob', 'Kinto', 1000000, '2019-01-20', 'Finance'),  
-> (2, 'Jerry', 'Kanso', 600000, '2019-01-15', 'IT'),  
-> (3, 'Philip', 'Jose', 890000, '2019-02-05', 'Banking'),  
-> (4, 'John', 'Abraham', 2000000, '2019-02-25', 'Insurance'),  
-> (5, 'Michael', 'Mathew', 2200000, '2019-02-28', 'Finance'),  
-> (6, 'Alex', 'chreketo', 400000, '2019-05-10', 'IT'),  
-> (7, 'Yohan', 'Soso', 1230000, '2019-06-20', 'Banking');  
Query OK, 7 rows affected (0.02 sec)  
Records: 7 Duplicates: 0 Warnings: 0
```

Figure 2: Registros de Employee

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

```
CREATE TABLE Reward (  
    Employee_ref_id INT,  
    date_reward DATE,  
    amount INT  
);
```

Figure 3: Creación de Reward

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

```
mysql> INSERT INTO Reward (Employee_ref_id, date_reward, amount) VALUES
-> (1, '2019-05-11', 1000),
-> (2, '2019-02-15', 5000),
-> (3, '2019-04-22', 2000),
-> (1, '2019-06-20', 8000);
Query OK, 4 rows affected (0.01 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

Figure 4: Registros de Reward

5. Obtener todos los empleados.

```
mysql> SELECT * FROM Employee;
+-----+-----+-----+-----+-----+-----+
| Employee_id | First_name | Last_name | Salary | Joining_date | Department |
+-----+-----+-----+-----+-----+-----+
| 1 | Bob | Kinto | 1000000 | 2019-01-20 | Finance |
| 2 | Jerry | Kansxo | 6000000 | 2019-01-15 | IT |
| 3 | Philip | Jose | 8900000 | 2019-02-05 | Banking |
| 4 | John | Abraham | 2000000 | 2019-02-25 | Insurance |
| 5 | Michael | Mathew | 2200000 | 2019-02-28 | Finance |
| 6 | Alex | chreketo | 4000000 | 2019-05-10 | IT |
| 7 | Yohan | Soso | 1230000 | 2019-06-20 | Banking |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

Figure 5: Mostrar tabla Employeee

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

```
mysql> SELECT First_name, Last_name FROM Employee;
+-----+-----+
| First_name | Last_name |
+-----+-----+
| Bob | Kinto |
| Jerry | Kansxo |
| Philip | Jose |
| John | Abraham |
| Michael | Mathew |
| Alex | chreketo |
| Yohan | Soso |
+-----+-----+
7 rows in set (0.00 sec)
```

Figure 6: Nombre y Apellido de Empleados

7. Obtener todos los valores de la columna “First name” usando el alias “Nombre de empleado”.

```
mysql> SELECT First_name AS 'Nombre de empleado' FROM Employee;
+-----+
| Nombre de empleado |
+-----+
| Bob                |
| Jerry              |
| Philip             |
| John               |
| Michael            |
| Alex               |
| Yohan              |
+-----+
7 rows in set (0.00 sec)
```

Figure 7: Alias Nombre

8. Obtener todos los valores de la columna “Last name” en minúsculas.

```
mysql> SELECT LOWER>Last_name) FROM Employee;
+-----+
| LOWER>Last_name) |
+-----+
| kinto             |
| kansxo            |
| jose              |
| abraham           |
| mathew            |
| chreketo          |
| soso              |
+-----+
7 rows in set (0.01 sec)
```

Figure 8: Apellido en minúsculas

9. Obtener todos los valores de la columna “Last name” en mayúsculas.

```
mysql> SELECT UPPER>Last_name) FROM Employee;
+-----+
| UPPER>Last_name) |
+-----+
| KINTO             |
| KANSXO            |
| JOSE              |
| ABRAHAM           |
| MATHEW            |
| CHREKETO          |
| SOSO              |
+-----+
7 rows in set (0.00 sec)
```

Figure 9: Apellido en mayúsculas

10. Obtener los nombre únicos de la columna “Departament”.

```
mysql> SELECT DISTINCT Department FROM Employee;
+-----+
| Department |
+-----+
| Finance    |
| IT         |
| Banking    |
| Insurance  |
+-----+
4 rows in set (0.01 sec)
```

Figure 10: Nombre de Departament

11. . Obtener los primeros 4 caracteres de todos los valores de la columna “First name”.

```
mysql> SELECT LEFT(First_name, 4) FROM Employee;
+-----+
| LEFT(First_name, 4) |
+-----+
| Bob                 |
| Jerr                 |
| Phil                 |
| John                 |
| Mich                 |
| Alex                 |
| Yoha                 |
+-----+
7 rows in set (0.00 sec)
```

Figure 11: 4 Valores de First name

12. Obtener la posición de la letra “h” en el nombre del empleado con First-name=“John”.

```
mysql> SELECT LOCATE('h', First_name) AS position_of_h
-> FROM Employee
-> WHERE First_name = 'John';
+-----+
| position_of_h |
+-----+
| 3             |
+-----+
1 row in set (0.00 sec)
```

Figure 12: posición de ”h”

13. Obtener todos los valores de la columna “First-name” después de remover los espacios en blanco de la derecha.

```
mysql> SELECT RTRIM(First_name) FROM Employee;
+-----+
| RTRIM(First_name) |
+-----+
| Bob                |
| Jerry              |
| Philip             |
| John               |
| Michael            |
| Alex               |
| Yohan              |
+-----+
7 rows in set (0.01 sec)
```

Figure 13: Espacio en blanco derecha

14. Obtener todos los valores de la columna “First- name” después de remover los espacios en blanco de la izquierda.

```
mysql> SELECT LTRIM(First_name) AS trimmed_first_name
-> FROM Employee;
+-----+
| trimmed_first_name |
+-----+
| Bob                |
| Jerry              |
| Philip             |
| John               |
| Michael            |
| Alex               |
| Yohan              |
+-----+
7 rows in set (0.00 sec)
```

Figure 14: Espacio en blanco Izquierda

4. Conclusiones

En esta práctica, se abordaron conceptos y operaciones esenciales para la gestión de bases de datos distribuidas utilizando MySQL. A través de la creación y manipulación de las tablas Employee y Reward, se reforzaron habilidades clave en SQL, tales como la creación de tablas, inserción de registros, y ejecución de consultas para extraer y transformar datos. Además, se exploraron funciones básicas de SQL como la conversión de texto a mayúsculas y minúsculas, la eliminación de espacios en blanco, y el uso de alias.

Estas actividades proporcionan una base sólida para la comprensión de cómo gestionar y manipular datos en un entorno de bases de datos relacionales, lo cual es fundamental en el desarrollo de aplicaciones y en la administración de información en sistemas distribuidos. El dominio de estas técnicas no solo permite realizar consultas eficientes, sino que también garantiza la integridad y consistencia de los datos.

Referencias Bibliográficas

References

- [1] Grabowska, S.; Saniuk, S. (2022). Business models in the industry 4.0 environment—results of web of science bibliometric analysis. *J. Open Innov. Technol. Mark. Complex*, 8(1), 19.
- [2] Connolly, T., & Begg, C. (2015). *Database systems: A practical approach to design, implementation, and management* (6th ed.). Pearson.
- [3] Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Pearson.
- [4] MySQL. (2024). *MySQL 8.0 Reference Manual*. Oracle Corporation. Retrieved from <https://dev.mysql.com/doc/refman/8.0/en/>