

Nombre de la práctica	Funciones en MySQL			No.	5
Asignatura:	Taller de Bases de Datos	Carrera:	Ingeniería en Sistemas Computacionales	Duración de la práctica (Hrs)	2

I. Nombre del alumno: Francisco David Colin Lira

II. Competencia(s) específica(s):

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro): Aula

III. Material empleado: Laptop, gestor de bases de datos

IV. Desarrollo de la práctica:

Creación de la base de datos:

```
CREATE DATABASE funciones;
USE funciones;

CREATE TABLE empleados (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    department VARCHAR(50),
    salary DECIMAL(10, 2) CHECK(salary > 0),
    fecha_ingreso DATE DEFAULT(NOW())
);
```

E insertamos 94 datos...

```
INSERT INTO empleados (nombre, department, salary, fecha_ingreso)
VALUES
('Juan Pérez', 'Ventas', 3500.00, '2020-05-12'),
('Ana Gómez', 'Marketing', 4500.50, '2018-11-22'),
('Carlos Martínez', 'Ventas', 2800.00, '2019-03-15'),
('Lucía Fernández', 'Recursos Humanos', 3900.75, '2021-09-01'),
```

Ejercicios:

1.- Calcular el salario promedio de los empleados: Queremos saber cuál es el salario promedio de los empleados en la empresa. Calcula este valor utilizando una función agregada adecuada.

```
SELECT AVG(salary) AS promedio_sueldo FROM empleados;
```

123 promedio_sueldo ▼
4,339.779787

2.- Contar el número de empleados en cada departamento: Deseamos saber cuántos empleados trabajan en cada departamento. Para ello, necesitas agrupar a los empleados por departamento y contar cuántos hay en cada uno.

```
SELECT department, COUNT(department) AS departement_emple FROM empleados GROUP BY department;
```

A-Z department ▼	123 departement_emple ▼
Ventas	21
Marketing	20
Recursos Humanos	18
IT	19
Finanzas	16

3.- Encontrar el salario más alto y más bajo: La gerencia quiere conocer el salario más alto y el salario más bajo entre todos los empleados.

```
-- Minimo  
SELECT MIN(salary) AS salario_minimo FROM empleados;  
-- Maximo  
SELECT MAX(salary) AS salario_maximo FROM empleados;
```

123 salario_minimo ▼
2,800

123 salario_maximo ▼
5,450.25

4.- Convertir los nombres de los empleados a mayúsculas: Se necesita una lista de todos los nombres de empleados convertidos a mayúsculas.

```
SELECT UCASE(nombre) AS nombre_mayusculas FROM empleados;
```

A-Z nombre_mayusculas ▼
JUAN PÉREZ
ANA GÓMEZ
CARLOS MARTÍNEZ
LUCÍA FERNÁNDEZ
ROBERTO SÁNCHEZ
SOFÍA HERRERA
PEDRO CASTILLO
LAURA MORALES
MIGUEL TORRES

5.- Obtener la longitud de los nombres de los empleados: Queremos saber la longitud (número de caracteres) de los nombres de todos los empleados.

```
SELECT nombre, LENGTH(nombre) as longitud_nombre FROM empleados;
```

A-Z nombre ▼	123 longitud_nombre ▼
Juan Pérez	11
Ana Gómez	10
Carlos Martínez	16
Lucía Fernández	17
Roberto Sánchez	16
Sofía Herrera	14
Pedro Castillo	14
Laura Morales	13
Miguel Torres	13

6.- Extraer las primeras tres letras de cada nombre: Para un análisis de iniciales, necesitamos extraer las primeras tres letras del nombre de cada empleado.

```
SELECT MID(nombre, 1, 3) AS iniciales FROM empleados;
```

A-Z iniciales ▼
Jua
Ana
Car
Luc
Rob
Sof
Ped
Lau
Mig

7.- Formatear los salarios a dos decimales: Queremos mostrar los salarios de los empleados formateados con dos decimales de manera clara y legible.

```
SELECT FORMAT(salary, 2) AS salario_formateado FROM empleados;
```

A-Z salario_formateado ▼
3,500.00
4,500.50
2,800.00
3,900.75
5,000.00
4,700.60
3,300.45
3,700.00
4,150.25

8.- Obtener la fecha actual y calcular el tiempo desde el ingreso: Queremos saber cuánto tiempo ha pasado (en días) desde que cada empleado ingresó a la empresa hasta la fecha actual.

```
SELECT DATE(NOW()) AS fecha_actual;
```

```
SELECT DATEDIFF(NOW(), fecha_ingreso) AS  
tiempo_dias_desde_ingreso FROM empleados;
```

fecha_actual	123 tiempo_dias_desde_ingreso
2024-10-16	1,618
	2,155
	2,042
	1,141
	1,737
	2,768
	1,406
	1,899
	1,286

9. – Contar cuántos empleados tienen su nombre más largo de 10 caracteres: Necesitamos contar cuántos empleados tienen un nombre con más de 10 caracteres.

```
SELECT COUNT(nombre) AS nombre_mas_10_caracteres FROM  
empleados WHERE LENGTH(nombre) > 10;
```

123 nombre_mas_10_caracteres
89



10.- Filtrar empleados que ingresaron hace más de 5 años: Queremos obtener una lista de empleados que hayan ingresado hace más de 5 años.

```
SELECT nombre AS empleado_con_mas_5_anios FROM empleados WHERE  
DATEDIFF(NOW(), fecha_ingreso) > (365 * 5);
```

A-Z empleado_con_mas_5_anios	
Ana Gómez	
Carlos Martínez	
Sofía Herrera	
Laura Morales	
Sara López	
Jorge Díaz	
Manuel Vargas	
Adriana Silva	
José Hernández	

V. Conclusiones:

En esta practica conocido funciones que no conocía pero veo que ayudan a agilizar mucho mas el trabajo e incluso la búsqueda de información y pues en caso que se requiera buscar información de un poco mas complejo se pueden unir tablas e indicar que operaciones se usaran para que la consulta resulte como esperamos y pues que esto se vea de cierta manera bien, la verdad me gusto conocer estas funciones ya que son de mucha ayuda para mi.