

Funciones

El lenguaje SQL tiene funciones incorporadas para hacer cálculos sobre los datos. Las funciones se pueden dividir en dos grupos principales (aunque existen muchas más, dependiendo del sistema de bases de datos que se utilice):

Crear una tabla de ejemplo

```
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())  
);
```

Insertar datos en la tabla

```
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'),
('Roberto Sánchez', 'IT', 5000.00, '2020-01-14'),
('Sofía Herrera', 'Finanzas', 4700.60, '2017-03-19'),
('Pedro Castillo', 'Marketing', 3300.45, '2020-12-10'),
('Laura Morales', 'Ventas', 3700.00, '2019-08-05'),
('Miguel Torres', 'Recursos Humanos', 4150.25, '2021-04-09'),
('Sara López', 'IT', 5300.80, '2019-07-12'),
('Luis Rodríguez', 'Ventas', 3400.00, '2021-02-18'),
('Patricia Gómez', 'Marketing', 4550.75, '2020-10-03'),
('Jorge Díaz', 'Finanzas', 4900.50, '2018-06-21'),
('Natalia Jiménez', 'Recursos Humanos', 3800.30, '2022-05-16'),
('Alberto Ríos', 'IT', 5200.40, '2020-03-25'),
('Isabel Ortega', 'Ventas', 3600.90, '2019-11-29'),
('Fernando Pérez', 'Marketing', 4000.00, '2021-09-08'),
('Daniela Romero', 'Recursos Humanos', 4250.35, '2020-08-14'),
('Manuel Vargas', 'Finanzas', 4600.15, '2017-04-24'),
('Adriana Silva', 'IT', 5100.70, '2018-12-19'),
('José Hernández', 'Ventas', 3450.00, '2019-05-26'),
('Andrea Ruiz', 'Marketing', 4400.50, '2020-07-07'),
('Ricardo Flores', 'Finanzas', 4700.80, '2018-09-30'),
('Gabriela Navarro', 'IT', 5150.25, '2019-06-11'),
('Juan González', 'Ventas', 3500.00, '2020-01-22'),
('Alejandra Morales', 'Marketing', 4300.45, '2018-04-17'),
('Enrique Sánchez', 'Recursos Humanos', 3900.75, '2021-10-13'),
('Carmen Castro', 'IT', 5250.60, '2019-01-06'),
('Martín Gómez', 'Finanzas', 4600.10, '2020-02-20'),
('Diana Paredes', 'Ventas', 3350.80, '2021-08-14'),
('Samuel Castro', 'Marketing', 4200.35, '2019-09-24'),
('Paola Soto', 'Recursos Humanos', 3800.55, '2018-05-09'),
('Carlos Romero', 'Finanzas', 4900.00, '2021-03-30'),
('Vanessa Ortiz', 'IT', 5350.90, '2020-06-05'),
('Julio Fernández', 'Ventas', 3650.20, '2017-07-21'),
('Mónica Lara', 'Marketing', 4150.50, '2019-11-08'),
('Mario Rivas', 'Recursos Humanos', 4000.00, '2018-02-18'),
('Lorena González', 'IT', 5400.45, '2020-09-22'),
('Pablo Ruiz', 'Ventas', 3500.60, '2021-05-14'),
('María García', 'Marketing', 4400.00, '2021-01-26'),
('Cristian Ramos', 'Finanzas', 4700.90, '2018-11-16'),
('Jessica Medina', 'IT', 5200.30, '2022-04-12'),
('Hugo Mendoza', 'Recursos Humanos', 4150.75, '2019-10-02'),
('Elena Salinas', 'Ventas', 3450.00, '2021-12-17'),
('Gustavo Torres', 'Marketing', 4600.50, '2018-03-04'),
('Carolina Ramírez', 'Finanzas', 4850.25, '2019-07-23'),
('Ángel Sánchez', 'IT', 5050.70, '2020-05-11'),
('Laura Vargas', 'Ventas', 3400.90, '2019-08-29'),
```

('Rodrigo Herrera', 'Marketing', 4500.00, '2020-03-06'),
('Verónica Delgado', 'Recursos Humanos', 4200.80, '2021-07-25'),
('Javier Martínez', 'Finanzas', 4700.45, '2017-09-28'),
('Isabel Rodríguez', 'IT', 5350.10, '2019-04-03'),
('Felipe Ortega', 'Ventas', 3550.35, '2021-11-22'),
('Sandra Ruiz', 'Marketing', 4250.00, '2020-08-10'),
('Óscar Castro', 'Recursos Humanos', 4100.75, '2019-05-15'),
('Sergio Pérez', 'Finanzas', 4800.90, '2020-12-18'),
('Liliana Moreno', 'IT', 5300.00, '2018-06-27'),
('Tomás Navarro', 'Ventas', 3450.45, '2020-09-05'),
('Álvaro Silva', 'Marketing', 4600.80, '2021-10-12'),
('Patricia Paredes', 'Recursos Humanos', 4300.90, '2018-07-02'),
('David Jiménez', 'Finanzas', 4700.30, '2019-02-08'),
('Raquel Ortiz', 'IT', 5450.25, '2021-04-26'),
('Sebastián Hernández', 'Ventas', 3550.00, '2020-01-14'),
('Rosa Gómez', 'Marketing', 4450.50, '2019-06-18'),
('Claudia Pérez', 'Recursos Humanos', 3950.35, '2021-05-27'),
('Ángela Morales', 'Finanzas', 4900.20, '2017-12-21'),
('Eduardo Castillo', 'IT', 5200.75, '2020-11-04'),
('Gabriel Martínez', 'Ventas', 3500.15, '2019-03-30'),
('Mariana Ríos', 'Marketing', 4350.00, '2018-01-16'),
('Carlos Ruiz', 'Recursos Humanos', 4200.60, '2021-09-14'),
('Lucía Fernández', 'IT', 5300.80, '2019-04-25'),
('Antonio Díaz', 'Finanzas', 4600.70, '2020-02-28'),
('Valeria Ramos', 'Ventas', 3450.10, '2020-07-15'),
('Héctor Ramírez', 'Marketing', 4550.50, '2021-08-09'),
('Silvia Navarro', 'Recursos Humanos', 4050.00, '2019-10-31'),
('Diego Soto', 'IT', 5100.60, '2020-06-23'),
('Rebeca Ruiz', 'Finanzas', 4750.85, '2018-05-11'),
('Andrés Ortiz', 'Ventas', 3550.40, '2019-02-19'),
('Esteban Torres', 'Marketing', 4450.75, '2021-12-08'),
('Beatriz Salinas', 'Recursos Humanos', 4000.30, '2017-11-13'),
('Fabián Pérez', 'IT', 5400.45, '2020-01-28'),
('Eva Morales', 'Ventas', 3350.90, '2021-10-01'),
('Gerardo Sánchez', 'Marketing', 4200.60, '2019-08-16'),
('Marta Gómez', 'Recursos Humanos', 4050.35, '2020-04-17'),
('Alfonso López', 'Finanzas', 4850.10, '2018-07-29'),
('Inés Vargas', 'IT', 5150.85, '2021-03-01'),
('Pablo Romero', 'Ventas', 3550.00, '2019-01-04'),
('Clara Fernández', 'Marketing', 4350.45, '2020-02-09'),
('Marcelo Ramírez', 'Recursos Humanos', 4250.60, '2021-06-14'),
('Teresa Martínez', 'Finanzas', 4700.50, '2019-07-08'),
('Arturo Delgado', 'IT', 5300.25, '2020-09-03'),
('Daniel Suárez', 'Ventas', 3450.85, '2018-03-27'),
('Sonia Lara', 'Marketing', 4450.20, '2019-05-23'),
('Ignacio Sánchez', 'Recursos Humanos', 4100.10, '2021-11-05');

Funciones agregadas SQL:

Son aquellas que devuelven un solo valor, calculado con los valores de una columna.

- **AVG():** Devuelve el promedio de los valores de una columna numérica.

Ejemplo:

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

- **COUNT():** Devuelve el número de filas que coinciden con un criterio.

Ejemplo:

```
SELECT COUNT(*) AS total_empleados FROM empleados WHERE department = 'Sales';
```

- **MIN():** Devuelve el valor mínimo de una columna.

Ejemplo:

```
SELECT MIN(salary) AS salario_minimo FROM empleados;
```

- **MAX():** Devuelve el valor máximo de una columna.

Ejemplo:

```
SELECT MAX(salary) AS salario_maximo FROM empleados;
```

- **SUM():** Devuelve la suma total de una columna numérica.

Ejemplo:

```
SELECT SUM(salary) AS total_salarios FROM empleados;
```

- **GROUP BY:** Agrupa filas que tienen los mismos valores en columnas específicas.

Ejemplo:

```
SELECT department, AVG(salary) AS promedio_suelto
FROM empleados
GROUP BY department;
```

Funciones escalares SQL:

Son aquellas que devuelven un solo valor basado en el valor de entrada.

- **UCASE()**: Convierte una cadena a mayúsculas.

Ejemplo:

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

- **LCASE()**: Convierte una cadena a minúsculas.

Ejemplo:

```
SELECT LCASE(nombre) AS nombre_minusculas FROM empleados;
```

- **MID()**: Extrae una subcadena de un campo de texto (cadena, inicio, longitud).

Ejemplo:

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

- **LENGTH()**: Devuelve la longitud de una cadena.

Ejemplo:

```
SELECT LENGTH(nombre) AS longitud_nombre FROM empleados;
```

- **NOW()**: Devuelve la fecha y hora actuales.

Ejemplo:

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

- **FORMAT()**: Formatea un número a un formato específico.

Ejemplo:

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

Otras funciones comunes en MySQL 8:

Funciones matemáticas:

- **CEIL()**: Redondea un número hacia arriba.

Ejemplo:

```
SELECT CEIL(4.3) AS redondeo_arriba;
```

- **FLOOR()**: Redondea un número hacia abajo.

Ejemplo:

```
SELECT FLOOR(4.7) AS redondeo_abajo;
```

- **ROUND()**: Redondea un número a la cantidad de decimales especificada.

Ejemplo:

```
SELECT ROUND(123.456, 2) AS numero_redondeado;
```

Funciones de fecha y hora:

- **DATE()**: Extrae la parte de fecha de una expresión de fecha y hora.

Ejemplo:

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

- **DAY()**: Devuelve el día de una fecha.

Ejemplo:

```
SELECT DAY(NOW()) AS dia_actual;
```

- **YEAR()**: Devuelve el año de una fecha.

Ejemplo:

```
SELECT YEAR(NOW()) AS año_actual;
```

- **DATEDIFF()**: Calcula la diferencia entre dos fechas.

Ejemplo:

```
SELECT DATEDIFF('2024-12-25', NOW()) AS dias_para_navidad;
```

Ejercicios

Calcular el salario promedio de los empleados

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

Contar el número de empleados en cada departamento

Enunciado: Deseamos saber cuántos empleados trabajan en cada departamento. Para ello, necesitas agrupar a los empleados por departamento y contar cuántos hay en cada uno.

Encontrar el salario más alto y más bajo

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

Convertir los nombres de los empleados a mayúsculas

Enunciado: Se necesita una lista de todos los nombres de empleados convertidos a mayúsculas.

Obtener la longitud de los nombres de los empleados

Enunciado: Queremos saber la longitud (número de caracteres) de los nombres de todos los empleados.

Extraer las primeras tres letras de cada nombre

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

Formatear los salarios a dos decimales

Enunciado: Queremos mostrar los salarios de los empleados formateados con dos decimales de manera clara y legible.

Obtener la fecha actual y calcular el tiempo desde el ingreso

Enunciado: Queremos saber cuánto tiempo ha pasado (en días) desde que cada empleado ingresó a la empresa hasta la fecha actual.

Contar cuántos empleados tienen su nombre más largo de 10 caracteres

Enunciado: Necesitamos contar cuántos empleados tienen un nombre con más de 10 caracteres.

Filtrar empleados que ingresaron hace más de 5 años

Enunciado: Queremos obtener una lista de empleados que hayan ingresado hace más de 5 años.