

Trabajo Practico – Funciones de Agregado

Dadas las siguientes tablas y registros, resolver (script's al final):

Empleados (Empleados)

- id_empleado (INT, PRIMARY KEY)
- nombre (VARCHAR(100))
- apellido (VARCHAR(100))
- fecha_contratacion (DATE)
- salario (DECIMAL(10, 2))
- id_departamento (INT, FOREIGN KEY)

Departamentos (Departamentos)

- id_departamento (INT, PRIMARY KEY)
- nombre_departamento (VARCHAR(100))
- ubicacion (VARCHAR(100))

Proyectos (Proyectos)

- id_proyecto (INT, PRIMARY KEY)
- nombre_proyecto (VARCHAR(100))
- id_departamento (INT, FOREIGN KEY)
- presupuesto (DECIMAL(10, 2))

1. Obtener una lista de todos los empleados ordenados por su apellido en orden ascendente.
2. Mostrar el nombre de cada empleado junto con el nombre del departamento al que pertenece.
3. Obtener el salario promedio de todos los empleados.
4. Contar cuántos empleados hay en cada departamento.
5. Mostrar los departamentos con más de 5 empleados.
6. Calcular el salario total de todos los empleados por departamento, y mostrar
7. Mostrar todos los empleados cuyo salario esté entre \$30,000 y \$60,000.
8. Obtener la cantidad de empleados que fueron contratados antes del 1 de enero de 2015.
9. Listar todos los empleados y los proyectos en los que trabaja su departamento (si no trabaja en ningún proyecto, mostrar NULL).
10. Mostrar todos los proyectos con el nombre de su departamento correspondiente.
11. Obtener la lista de proyectos con su presupuesto total, agrupados por departamento.
12. Calcular la media de salarios en cada departamento y ordenarlos de mayor a menor.
13. Listar todos los empleados cuyo apellido comience con 'G'.
14. Mostrar la suma de salarios de todos los empleados que trabajan en departamentos ubicados en 'New York'.
15. Mostrar el departamento con el menor número de empleados (en caso de haber mas de 1, mostrar solo 1).
16. Obtener el nombre del proyecto con el presupuesto más alto.
17. Contar cuántos proyectos tienen un presupuesto menor a \$50,000.
18. Obtener el salario promedio por departamento y solo mostrar los departamentos cuyo salario promedio sea mayor a \$50,000.
19. Listar a todos los empleados contratados este año.
20. Mostrar los empleados contratados en el mes actual.
21. Obtener la cantidad de empleados que fueron contratados en los últimos 6 meses.
22. Listar a los empleados cuyo aniversario de contratación es este mes.
23. Calcular cuántos días lleva cada empleado trabajando en la empresa.
24. Mostrar a los empleados contratados hace más de 10 años.
25. Listar a los empleados contratados en los últimos 2 años.
26. Obtener los empleados que serán contratados dentro de los próximos 30 días, simulando que ya existe una fecha de contratación futura.

Script's:

```
CREATE TABLE Departamentos (  
    id_departamento INT PRIMARY KEY,  
    nombre_departamento VARCHAR(100) NOT NULL,  
    ubicacion VARCHAR(100) NOT NULL  
);
```

```
CREATE TABLE Empleados (  
    id_empleado INT PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    apellido VARCHAR(100) NOT NULL,  
    fecha_contratacion DATE NOT NULL,  
    salario DECIMAL(10, 2) NOT NULL,  
    id_departamento INT,  
    FOREIGN KEY (id_departamento) REFERENCES Departamentos(id_departamento)  
);
```

```
CREATE TABLE Proyectos (  
    id_proyecto INT PRIMARY KEY,  
    nombre_proyecto VARCHAR(100) NOT NULL,  
    id_departamento INT,  
    presupuesto DECIMAL(10, 2) NOT NULL,  
    FOREIGN KEY (id_departamento) REFERENCES Departamentos(id_departamento)  
);
```

```
INSERT INTO Departamentos (id_departamento, nombre_departamento, ubicacion) VALUES  
(1, 'Recursos Humanos', 'New York'),  
(2, 'Tecnología', 'San Francisco'),  
(3, 'Marketing', 'Miami'),  
(4, 'Ventas', 'New York'),  
(5, 'Finanzas', 'Chicago');
```

```
INSERT INTO Empleados (id_empleado, nombre, apellido, fecha_contratacion, salario,  
id_departamento) VALUES  
(1, 'Juan', 'Gómez', '2020-05-15', 55000, 1),  
(2, 'Ana', 'Martínez', '2017-08-20', 45000, 2),  
(3, 'Pedro', 'Hernández', '2015-03-10', 60000, 2),  
(4, 'Laura', 'García', '2010-11-01', 70000, 3),  
(5, 'María', 'Pérez', '2018-04-22', 50000, 4),  
(6, 'David', 'Jiménez', '2012-06-17', 75000, 4),  
(7, 'Carmen', 'González', '2023-01-05', 40000, 1),  
(8, 'Luis', 'Rodríguez', '2024-09-09', 48000, 5),  
(9, 'Sofía', 'Sánchez', '2021-10-01', 32000, 3),  
(10, 'Javier', 'Guerrero', '2019-07-18', 58000, 1),  
(11, 'Claudia', 'Giménez', '2024-09-12', 63000, 2),  
(12, 'Carlos', 'Gil', '2011-02-14', 90000, 5);
```

```
INSERT INTO Proyectos (id_proyecto, nombre_proyecto, id_departamento, presupuesto) VALUES  
(1, 'Optimización de Recursos', 1, 120000),  
(2, 'Desarrollo Web', 2, 95000),  
(3, 'Campaña Publicitaria', 3, 75000),  
(4, 'Expansión de Mercado', 4, 30000),  
(5, 'Automatización Financiera', 5, 60000),  
(6, 'Rediseño de Marca', 3, 50000),  
(7, 'Auditoría Interna', 5, 100000),  
(8, 'Transformación Digital', 2, 80000);
```