

Ejercicio de Optimización de SQL

Vamos a trabajar con una base de datos simple que contiene información sobre empleados y sus departamentos. El objetivo es crear las tablas, insertar datos, y realizar algunas consultas optimizadas.

Estructura de la Base de Datos

Supongamos que tenemos dos tablas: **employees** y **departments**.

Tabla employees:

employee_id	name	age	department_id
1	John Doe	30	1
2	Jane Smith	25	2
3	Sam Brown	28	1
4	Lucy Green	35	3

Tabla departments:

department_id	department_name
1	Sales
2	Marketing
3	IT

Pasos para el Ejercicio

1. Crear las Tablas y Insertar Datos:

Primero, creamos las tablas **employees** y **departments** y las llenamos con algunos datos.

```
CREATE TABLE departments ( department_id INT PRIMARY KEY, department_name VARCHAR(50) );
```

```
CREATE TABLE employees ( employee_id INT PRIMARY KEY, name VARCHAR(50), age INT, department_id INT, FOREIGN KEY (department_id) REFERENCES departments(department_id) );
```

```
INSERT INTO departments (department_id, department_name) VALUES (1, 'Sales'), (2, 'Marketing'), (3, 'IT');
```

```
INSERT INTO employees (employee_id, name, age, department_id) VALUES (1, 'John Doe', 30, 1), (2, 'Jane Smith', 25, 2), (3, 'Sam Brown', 28, 1), (4, 'Lucy Green', 35, 3);
```

2. Consulta Básica:

Escribe una consulta que seleccione todos los empleados junto con el nombre de su departamento.

3. **Añadir Índices:**

Para mejorar el rendimiento de la consulta anterior, añade índices a las columnas que se utilizan para la unión y la búsqueda.

4. **Filtrado de Resultados:**

Modifica la consulta para seleccionar solo a los empleados del departamento de 'Sales'.

5. **Eliminar Subconsultas Innecesarias:**

A veces, las subconsultas pueden hacer que las consultas sean más lentas. Reescribe la siguiente consulta utilizando un **JOIN** en lugar de una subconsulta.

6. **Uso de LIMIT:**

Si solo necesitas una parte del conjunto de resultados, usa **LIMIT** para reducir la cantidad de datos procesados.

Ejercicio 7: Contar Empleados por Departamento

Escribe una consulta que cuente el número de empleados en cada departamento y optimízala.

Optimización:

- Usa **GROUP BY** para agrupar por departamento.
- Asegúrate de que la columna utilizada en **GROUP BY** tenga un índice.

Ejercicio 2: Obtener Edad Promedio por Departamento

Escribe una consulta que calcule la edad promedio de los empleados en cada departamento.

Optimización:

- Utiliza índices en las columnas que participan en las uniones y agregaciones.

Ejercicio 3: Filtrar por Edad

Escribe una consulta que seleccione todos los empleados mayores de 30 años junto con su nombre de departamento.

Ejercicio 4: Encontrar el Empleado Más Joven de Cada Departamento

Escribe una consulta que encuentre el empleado más joven en cada departamento.

Ejercicio 5: Ordenar Empleados por Edad y Departamento

Escribe una consulta que seleccione todos los empleados y los ordene primero por el nombre del departamento y luego por la edad en orden descendente.

Ejercicio 6: Buscar Empleados por Nombre Parcial

Escribe una consulta que seleccione todos los empleados cuyos nombres contienen la letra 'a'.

Optimización:

- Utiliza índices de texto completo si el motor de la base de datos lo permite (ej. FULLTEXT en MySQL).

Ejercicio 7: Empleados y sus Jefes

Supongamos que cada empleado tiene un jefe, y queremos listar a cada empleado junto con el nombre de su jefe. Para esto, añadimos una columna `manager_id` a la tabla `employees`.

Instrucciones para los Alumnos

1. Crear y Poblar las Tablas:

Ejecuta las sentencias de creación y poblamiento de tablas en tu gestor de base de datos SQL.

2. Ejecutar las Consultas:

Ejecuta cada una de las consultas en el orden dado para observar los resultados y entender cómo se mejoran con la optimización.

3. Analizar el Rendimiento:

Usa las herramientas de análisis de tu base de datos (como EXPLAIN en MySQL) para ver cómo cambian los planes de ejecución antes y después de añadir índices y optimizar consultas.

4. Modificar y Experimentar:

Prueba a modificar las consultas y los índices para ver cómo afectan los cambios al rendimiento de las consultas.