

Esta consulta está diseñada para encontrar clientes que tienen cuentas en **todas** las sucursales.

Consulta Original

```
SELECT c.codCliente, c.dni, c.nombreCliente
FROM Cliente c
WHERE NOT EXISTS (
    SELECT *
    FROM PropietarioCuenta p
    WHERE NOT EXISTS (
        SELECT *
        FROM Cuenta cu
        INNER JOIN Sucursal su ON cu.nombreSucursal = su.nombreSucursal
        WHERE c.codCliente = p.codCliente AND cu.nroCuenta = p.nroCuenta
    )
)
```

Para comprender cómo funciona, vamos a ver cómo se organizan las tablas y luego haremos un análisis paso a paso.

Suposiciones sobre la Base de Datos

1. **Cliente**: Cada cliente tiene un `codCliente`, `dni`, y `nombreCliente`.
2. **Sucursal**: Cada sucursal tiene un `nombreSucursal` y otros atributos.
3. **Cuenta**: Cada cuenta tiene un `nroCuenta` y está asociada a una `nombreSucursal`.
4. **PropietarioCuenta**: Relaciona clientes y cuentas, con `codCliente` y `nroCuenta`.

Imaginemos que tenemos tres sucursales: Sucursal A, Sucursal B, y Sucursal C.

Paso a Paso

1. Selección de los Clientes

```
SELECT c.codCliente, c.dni, c.nombreCliente
FROM Cliente c
```

Esta parte selecciona el `codCliente`, `dni`, y `nombreCliente` de los clientes que cumplan la condición especificada en el `WHERE`.

2. Condición `NOT EXISTS`

```
WHERE NOT EXISTS (
    ...
)
```

Esta condición `NOT EXISTS` busca encontrar clientes **para los cuales no existe** alguna sucursal en la que **no tengan una cuenta**. En otras palabras, si `NOT EXISTS` devuelve verdadero, significa que el cliente tiene una cuenta en todas las sucursales.

Ahora profundicemos en la subconsulta interna.

3. Primera Subconsulta `PropietarioCuenta`

```

SELECT *
FROM PropietarioCuenta p
WHERE NOT EXISTS (
    ...
)

```

En esta subconsulta, estamos seleccionando desde `PropietarioCuenta`. Básicamente, estamos recorriendo todas las cuentas de las que el cliente es propietario.

4. Segunda Subconsulta con `Cuenta` y `Sucursal`

```

WHERE NOT EXISTS (
    SELECT *
    FROM Cuenta cu
    INNER JOIN Sucursal su ON cu.nombreSucursal = su.nombreSucursal
    WHERE c.codCliente = p.codCliente AND cu.nroCuenta = p.nroCuenta
)

```

Esta segunda subconsulta verifica que para cada cuenta de un cliente, exista una cuenta en cada sucursal. Veamos qué significa esto:

- `INNER JOIN` entre `Cuenta` y `Sucursal`: Esto une las cuentas a sus respectivas sucursales.
- `WHERE c.codCliente = p.codCliente AND cu.nroCuenta = p.nroCuenta`: Esto asegura que estamos buscando cuentas específicas del cliente actual en todas las sucursales.

Entonces, si encontramos que no hay una cuenta para alguna sucursal en la que el cliente no es propietario, la condición `NOT EXISTS` de la primera subconsulta se cumple, y el cliente será excluido del resultado.

Ejemplo Práctico

Imagina una base de datos con estos datos simplificados:

Tabla Sucursal

nombreSucursal
A
B
C

Tabla Cliente

codCliente	dni	nombreCliente
1	123456789	Juan
2	987654321	María

Tabla Cuenta

nroCuenta	nombreSucursal
-----------	----------------

101	A
102	B
103	C
104	A
105	B

Tabla PropietarioCuenta

codCliente	nroCuenta
1	101
1	102
1	103
2	104
2	105

En este ejemplo, **Juan** (codCliente = 1) tiene cuentas en todas las sucursales A, B y C, por lo que cumpliría la condición y aparecería en el resultado.

María (codCliente = 2) solo tiene cuentas en las sucursales A y B, pero no en C, así que **no** aparecería en el resultado.

Resultado de la Consulta

Con estos datos, el resultado sería:

codCliente	dni	nombreCliente
1	123456789	Juan

Explicación Final

La consulta está diseñada para filtrar a los clientes que **no tienen cuentas en todas las sucursales**. Si el cliente tiene una cuenta en todas las sucursales, el `NOT EXISTS` de la subconsulta interna no se cumple, y entonces el cliente pasa a estar en el resultado.