05 - Writing Complex Query

Tuesday, April 2, 2024 12:07 AM

Sub Queries

Sub query is another select statement return some results that play good role in your main query

```
SELECT *
FROM employees
WHERE salary > (
   SELECT AVG(salary)
   FROM employees
);
```

NOTE
In case of the single returned result that's okay
BUT
In Case of the multi results query to look in them you need the **IN operator**

```
SELECT *
FROM clients
WHERE client_id NOT IN (
SELECT client_id
FROM invoices
)
```

The ALL Keyword

It's super nice it's like saying
Look at this values give me bigger than them , min than them and so on . . .. . .

```
SELECT *
FROM invoices
WHERE invoice_total > ALL (

   SELECT invoice_total
   FROM invoices
   WHERE client_id = 3
)
```

==

```
SELECT *
FROM invoices
WHERE invoice_total >  (

   SELECT MAX(invoice_total)
   FROM invoices
```

```
    WHERE client_id = 3
)
```

Any Operator

Any operator look like IN exactly look at this

```
SELECT *
FROM invoices
WHERE client_id = ANY (
SELECT client_id
FROM invoices
GROUP BY client_id
HAVING COUNT(*) >= 2 )
```

```
SELECT *
FROM invoices
WHERE client_id IN (
SELECT client_id
FROM invoices
GROUP BY client_id
HAVING COUNT(*) >= 2 )
```

Correlated Subqueries

What that Mean ?

Look at all sub queries that we typed until now there is no dependency between inner & outer Query

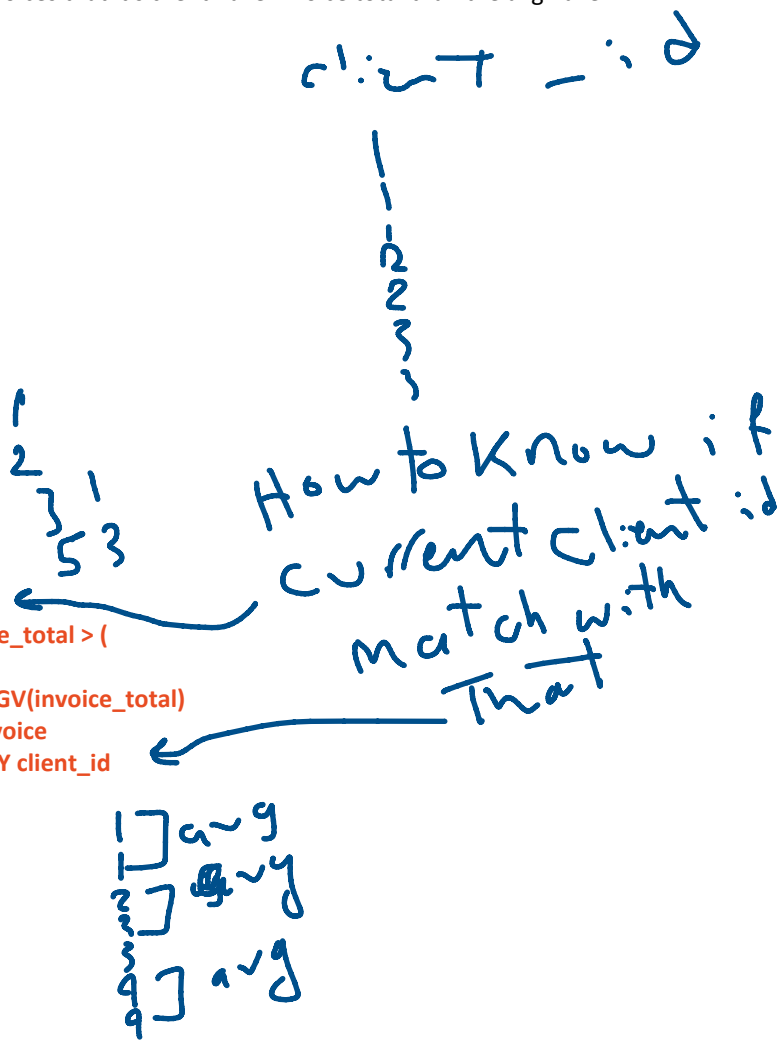
But look at this case

- Get invoice that are larger than client's avg invoice amount

invoice_id	number	client_id	invoice_total	payment_total	invoice_date	due_date	payment_date
1	91-953-3396	2	101.79	0.00	2019-03-09	2019-03-29	NULL
2	03-898-6735	5	175.32	8.18	2019-06-11	2019-07-01	2019-02-12
3	20-228-0335	5	147.99	0.00	2019-07-31	2019-08-20	NULL
4	56-934-0748	3	152.21	0.00	2019-03-08	2019-03-28	NULL
5	87-052-3121	5	169.36	0.00	2019-07-18	2019-08-07	NULL
6	75-587-6626	1	157.78	74.55	2019-01-29	2019-02-18	2019-01-03
7	68-093-9863	3	133.87	0.00	2019-09-04	2019-09-24	NULL
8	78-145-1093	1	189.12	0.00	2019-05-20	2019-06-09	NULL
9	77-593-0081	5	172.17	0.00	2019-07-09	2019-07-29	NULL
10	48-266-1517	1	159.50	0.00	2019-06-30	2019-07-20	NULL

The problem is that

-- return all invoices that it's client have invoice total than the avg have



If you say

```
SELECT *
FROM invoice
WHERE invoice_total > (

    SELECT AGV(invoice_total)
    FROM invoice
    GROUP BY client_id
)
```

SO you need something to tell you that we are same client_id

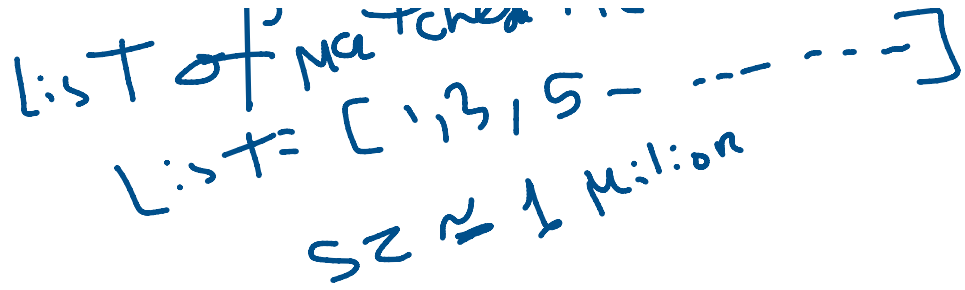
```
SELECT *
FROM invoices i
WHERE invoice_total > (
    SELECT AVG(invoice_total)
  FROM invoices
  WHERE client_id = i.client_id
);
```

The Exist Operator

What is the problem with **IN** operator ?

```
SELECT *
FROM invoices
```

```
SELECT *
FROM invoices
WHERE client_id IN (
SELECT client_id
FROM invoices
GROUP BY client_id
HAVING COUNT(*) >= 2 )
```

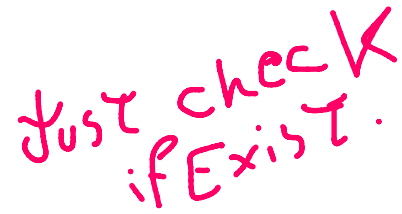


Do you notice the problem ?

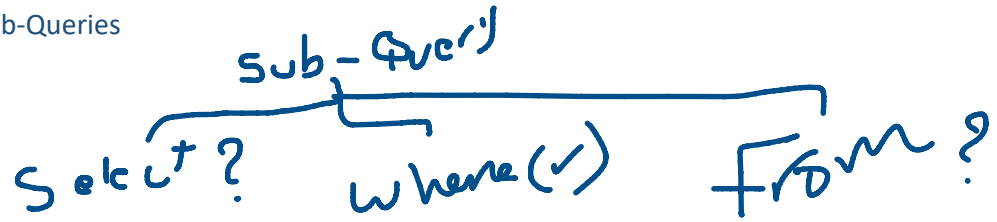
Super high space usage and that have a bad effect on the query performance

EXIST HERE FOR SOLVE THAT

```
SELECT *
FROM clients c
WHERE EXISTS (
  SELECT client_id
  FROM invoices
  WHERE client_id = c.client_id
)
```

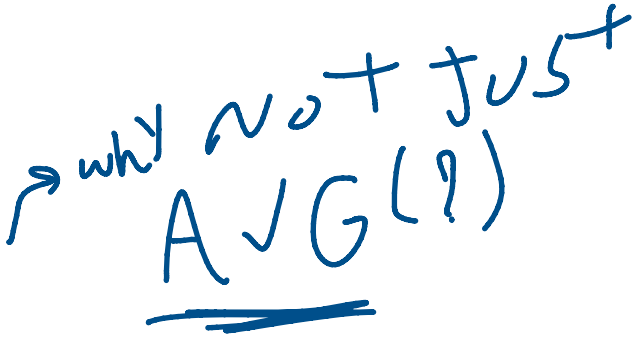


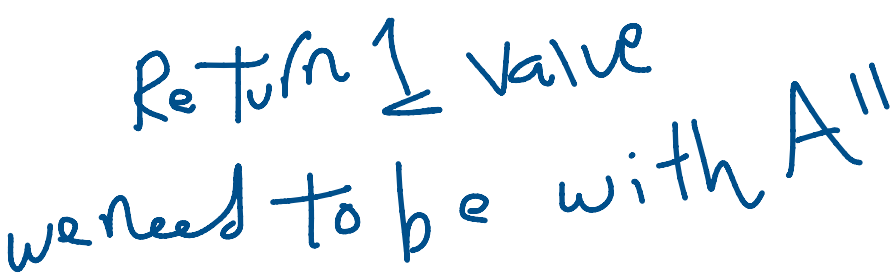
Notes about Sub-Queries



- **SELECT Sub Query**

```
SELECT
invoice_id,
invoice_total,
(SELECT AVG(invoice_total) FROM invoices ) AS 'avg_invoices',
(SELECT invoice_total - (avg_invoices) ) AS 'DIFF'
FROM invoices
```





```
SELECT
client_id ,
name ,
(SELECT SUM(invoice_total) FROM invoices WHERE client_id = c.client_id) AS 'Total sells' ,
(SELECT AVG(invoice_total) FROM invoices WHERE client_id = c.client_id) AS 'AVG sells'

FROM clients c
```

- **FROM as Sub query**

```
SELECT *
FROM (

        SELECT
            client_id ,
            name ,
            (SELECT SUM(invoice_total) FROM invoices WHERE client_id = c.client_id) AS 'Total_Sales' ,
            (SELECT AVG (invoice_total) FROM invoices WHERE client_id = c.client_id) AS 'AVG_Sales' ,
            (SELECT Total_Sales - AVG_Sales ) AS 'DIFF_Sales'

        FROM clients c
) AS Sales_Summary
```