

SQL REPORT PROBLEM

Gross monthly total sales report for Croma

 Attach

 Add a child issue

 Link issue



Description

As a product owner, **I need** an aggregate monthly gross sales report for Croma India customer **so that** I can track how much sales this particular customer is generating for AtliQ and manage our relationships accordingly.

The report should have the following fields,

1. Month
2. Total gross sales amount to Croma India in this month

SQL QUERY

```
SELECT
    s.date,
    SUM(ROUND(s.sold_quantity*g.gross_price,2)) as
monthly_sales
FROM fact_sales_monthly s
JOIN fact_gross_price g
ON g.fiscal_year=get_fiscal_year(s.date) AND
g.product_code=s.product_code
WHERE
customer_code=90002002
GROUP BY date;
```

FUNCTION CREATED FOR QUERY

```
CREATE FUNCTION
`get_fiscal_year`(calendar_date DATE)
    RETURNS int
    DETERMINISTIC
    BEGIN
        DECLARE fiscal_year INT;
        SET fiscal_year =
YEAR(DATE_ADD(calendar_date, INTERVAL
4 MONTH));
        RETURN fiscal_year;
    END
```

QUERY VIEW

The screenshot displays the MySQL Workbench interface. On the left, the 'SCHEMAS' pane shows a tree view of databases and functions. The 'Functions' section is expanded, showing the 'get_fiscal_year' function. Below this, the 'Parameters' section lists 'calendar_date: DATE' and the 'Returns' section indicates 'int'.

The main query editor shows a SQL query for 'Query 1' in the 'dim customer sql project' database. The query is as follows:

```
33
34 • SELECT
35     s.date,
36     SUM(ROUND(s.sold_quantity*g.gross_price,2)) as monthly_sales
37 FROM fact_sales_monthly s
38 JOIN fact_gross_price g
39     ON g.fiscal_year=get_fiscal_year(s.date) AND g.product_code=s.product_code
40 WHERE
41     customer_code=90002002
42 GROUP BY date
43 limit 10000;
44
```

The 'Result Grid' at the bottom shows the results of the query, with columns 'date' and 'monthly_sales'. The results are as follows:

date	monthly_sales
2017-09-01	122407.57
2017-10-01	162687.56
2017-12-01	245673.84
2018-01-01	127574.73
2018-02-01	144799.54
2018-04-01	130643.92

REQUIRED RESULT

The screenshot displays the MySQL Workbench interface. On the left, the 'SCHEMAS' pane shows a tree view of databases and tables. The 'Functions' section is expanded, showing the 'get_fiscal_year' function. The main query editor shows a SQL query that selects the date and a calculated monthly sales value. The 'Result Grid' pane at the bottom displays the output of the query, which is a table with two columns: 'date' and 'monthly_sales'. The table contains 18 rows of data, showing monthly sales figures from September 2017 to June 2019.

Function: get_fiscal_year

Parameters:
calendar_date: DATE

Returns: int

Query 1:

```
SELECT  
    s.date,  
    SUM(ROUND(s.sold_quantity*g.gross_price,2)) as monthly_sales
```

Result Grid:

date	monthly_sales
2017-09-01	122407.57
2017-10-01	162687.56
2017-12-01	245673.84
2018-01-01	127574.73
2018-02-01	144799.54
2018-04-01	130643.92
2018-05-01	139165.06
2018-06-01	125735.36
2018-08-01	125409.90
2018-09-01	343337.14
2018-10-01	440562.10
2018-12-01	653944.72
2019-01-01	359025.06
2019-02-01	356607.19
2019-04-01	379549.74
2019-05-01	340152.29
2019-06-01	343792.08