AtliQ Hardware

Finance Analytics

Presentation

Project Overview

AtilQ is a company that sells Hardware Like pc, mouse, printers, and so on to different customers,
this model is similar to Hp, or Dell

They sell or make hardware,
and they Sell it to stores
such as BEST BUY, CROMA, STAPLES
even online stores like Amazon and Flipkart

AtliQ Operates In Four Regions Namely
APAC(Asia Pacific),
EU(Europe),
NA(North America),
And LATAM(Latin America)

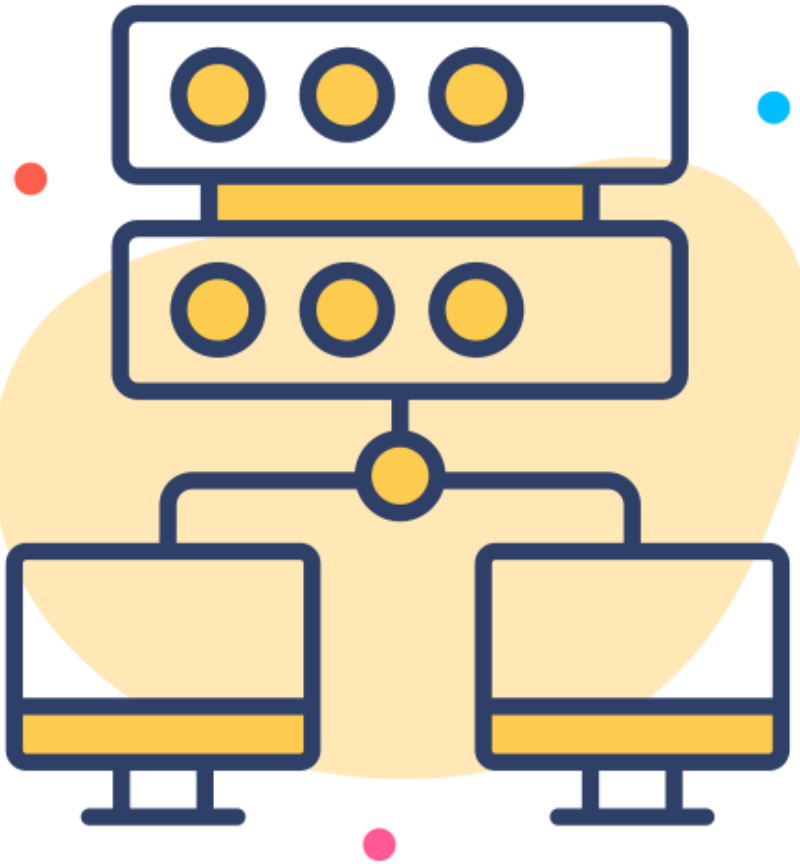
Task

- Generate monthly Gross sale reports
- Identify the financial position of The company over various Fiscal years
- Identify Top products , Markets, And customers

Dataset overview

- **Loaded a million rows of data into a Mysql database and Executed multiple queries**
- **Analyzed the following tables to derive the sql concepts presented :**

1. Dim_customer
2. Dim_date
3. Dim_product
4. Fact_act_est
5. Fact_forecast_monthly
6. Fact_freight_cost
7. Fact_gross_price
8. Fact_manufacturing_cost
9. Fact_post_invoice_deduction
10. Fact_pre_invoice_deduction
11. Fact_sales_monthly



- **Created Two User Defined SQL function**

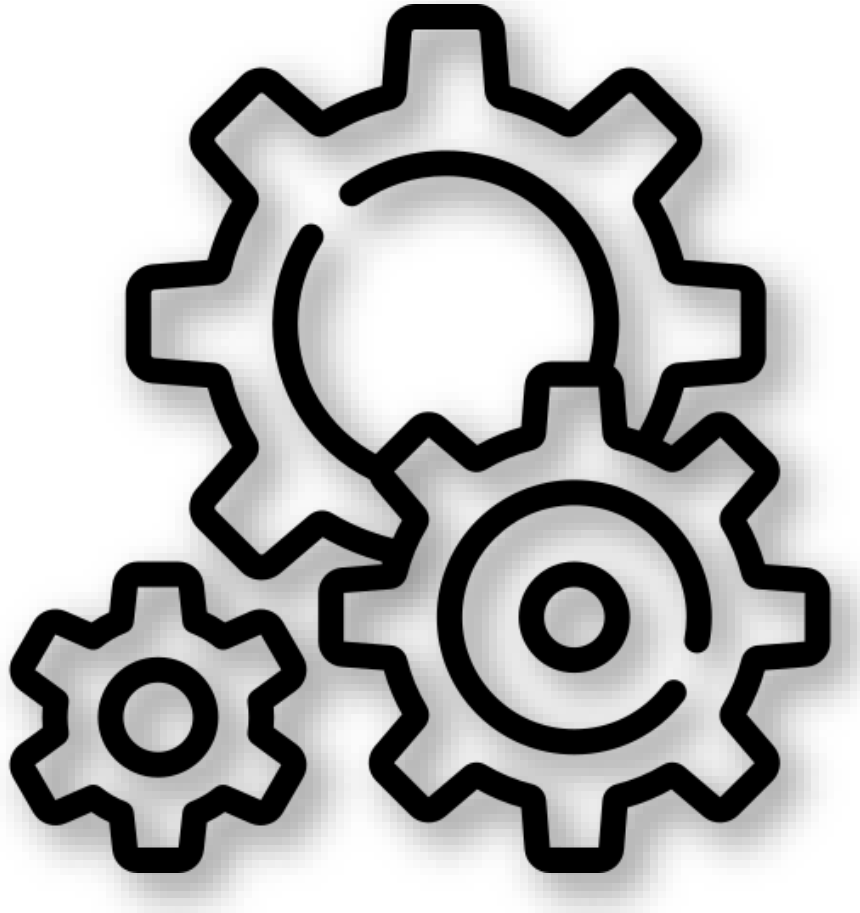
1. Get_fiscal_year
2. Get_fiscal_quaters

- **Five_stored_procedure**

1. Get_market_badge
2. Get_monthly_grosssales_for_given_customer
3. Get_top_n_customer_by_net_sales
4. Get_top_n_markets_by_net_sales
5. Get_top_n_products_by_net_sales

- **Four Views**

1. Gross_sales
2. Net_sales
3. Sales_postinvoice_discount
4. Sales_preinvoice_discount



Implementing user ---Defined SQL functions

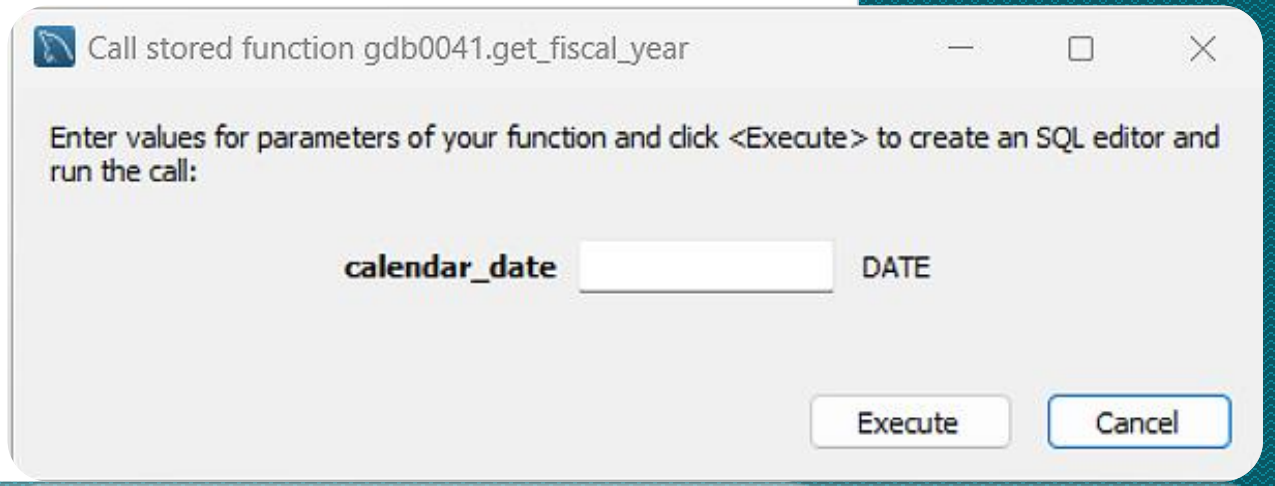
```
11
12 •  SELECT * FROM fact_sales_monthly
13        WHERE
14                customer_code=90002002 AND
15                get_fiscal_year(date)=2021
16        ORDER BY date asc
17
```

Result Grid | Filter Rows: | Edit:

date	product_code	customer_code	sold_quantity
2020-09-01	A0118150101	90002002	202
2020-09-01	A5419110207	90002002	5
2020-09-01	A0118150102	90002002	162
2020-09-01	A4218110207	90002002	26
2020-09-01	A5519110301	90002002	3
2020-09-01	A2918150102	90002002	296
2020-09-01	A0118150103	90002002	193
2020-09-01	A5621110408	90002002	14
2020-09-01	A3220150403	90002002	781
2020-09-01	A0118150104	90002002	146
2020-09-01	A5419110205	90002002	7
2020-09-01	A0219150201	90002002	149
2020-09-01	A4021150404	90002002	75
2020-09-01	A3120150304	90002002	831
2020-09-01	A6419160302	90002002	204
2020-09-01	A0219150202	90002002	107

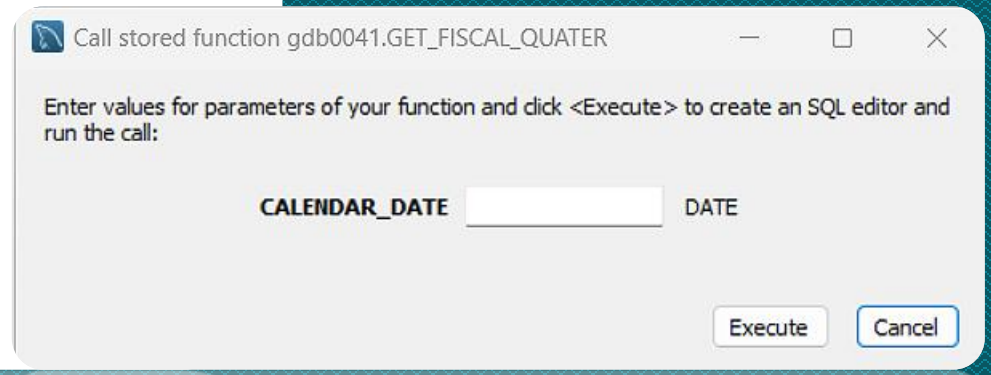
CREATING A FUNCTION FOR FISCAL YEAR

```sql
CREATE DEFINER=`root`@`localhost` 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
```



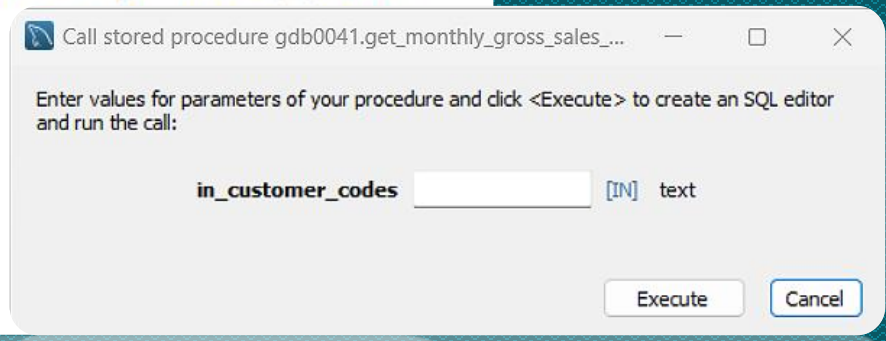
CREATING A FUNCTION FOR FISCAL YEAR

```
1  CREATE DEFINER=`root`@`localhost` FUNCTION `GET_FISCAL_QUATER`(CALENDAR_DATE DATE
2  ) RETURNS char(2) CHARSET utf8mb4
3      DETERMINISTIC
4  BEGIN
5  declare m tinyint;
6  declare qtr char(2);
7  SET m = month (calendar_date);
8  CASE
9  when m in (9,10,11)then
10 set qtr ="Q1";
11 when m in (12,1,2)then
12 set qtr ="Q2";
13 when m in (3,4,5)then
14 set qtr ="Q3";
15 else
16 set qtr = "q4";
17 end case ;
18 RETURN qtr ;
19 END
```



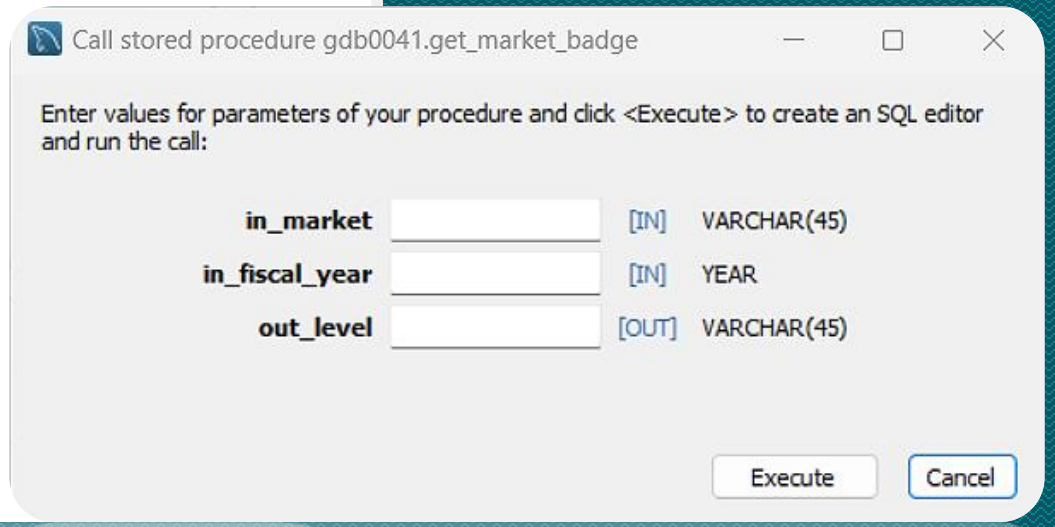
Generate monthly gross sales report for customer using stored procedure

```sql
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_monthly_gross_sales_for_custo
in_customer_codes text
)
BEGIN
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
find_in_set(s.customer_code,in_customer_codes)>0
    GROUP BY date;
END
```



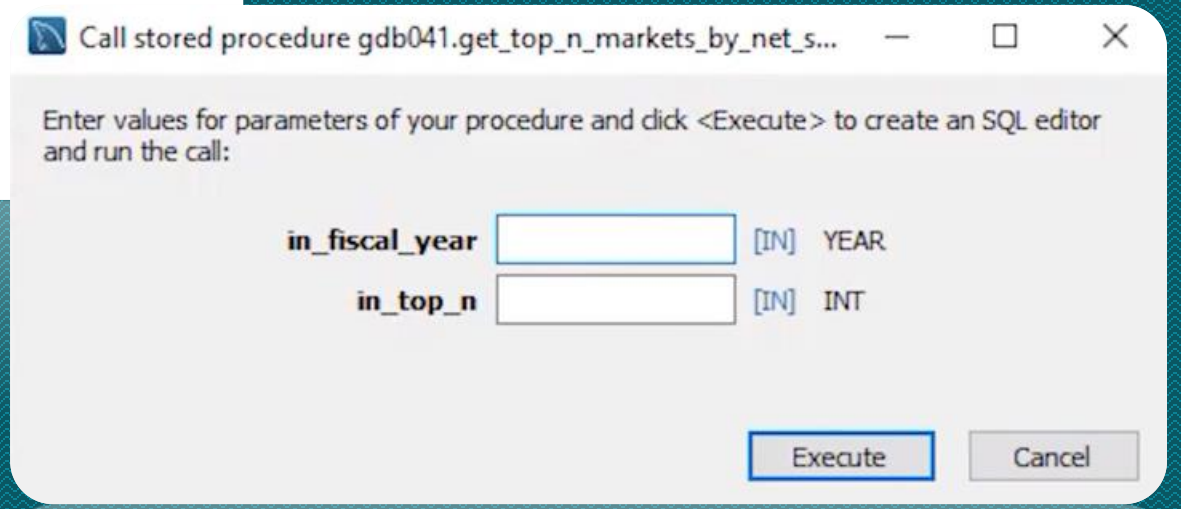
Creating stored procedures to generate report for Top—N customer by Net sales

```sql
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_market_badge`(
            IN in_market VARCHAR(45),
            IN in_fiscal_year YEAR,
            OUT out_level VARCHAR(45)
    )
BEGIN
             DECLARE qty INT DEFAULT 0;

             # Default market is India
             IF in_market = "" THEN
                  SET in_market="India";
             END IF;

             # Retrieve total sold quantity for a given market in a given year
             SELECT
                  SUM(s.sold_quantity) INTO qty
             FROM fact_sales_monthly s
             JOIN dim_customer c
             ON s.customer_code=c.customer_code
             WHERE
                  get_fiscal_year(s.date)=in_fiscal_year AND
                  c.market=in_market;

             # Determine Gold vs Silver status
             IF qty > 5000000 THEN
                  SET out_level = 'Gold';
             ELSE
                  SET out_level = 'Silver';
             END IF;
```



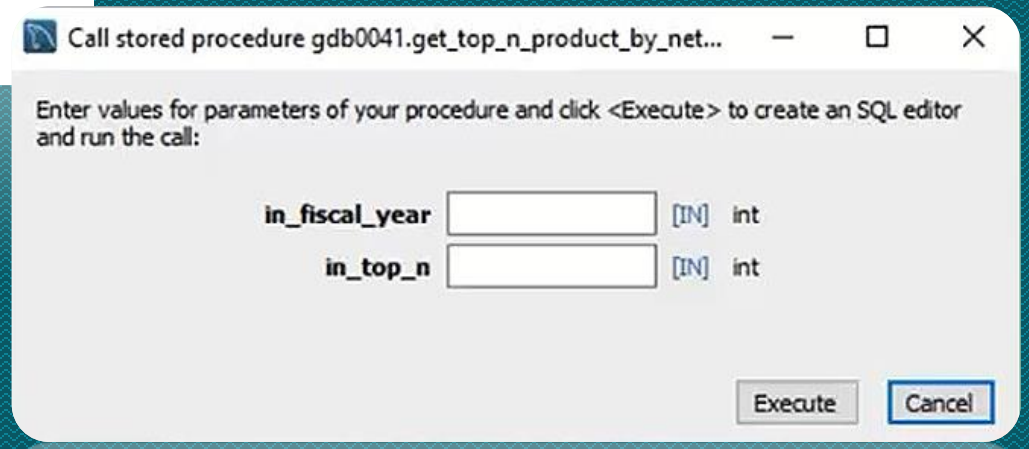
Creating stored procedures to generate report for Top—N Market by Net sales

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_top_n_market_by_net_sales`(
in_fisacl_year int,
in_top_n int
)
BEGIN
    SELECT
    market,
    round(sum(net_sales)/1000000,2) as net_sales_mln
     FROM gdb0041.net_sales
     where fiscal_year = in_fisacl_year
     group by market
     order by net_sales_mln desc
     limit in_top_n;
END
```



Creating stored procedures to generate report for Top—N Product by Net sales

```sql
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_top_n_product_by_net_sales`(
    in_fiscal_year int,
    in_top_n int
)
BEGIN
SELECT
product,
round(sum(net_sales)/1000000,2) as net_sales_mln
 FROM gdb0041.net_sales
 where fiscal_year = in_fiscal_year
 group by product
 order by net_sales_mln desc
 limit in_top_n;
END
```



Thank You