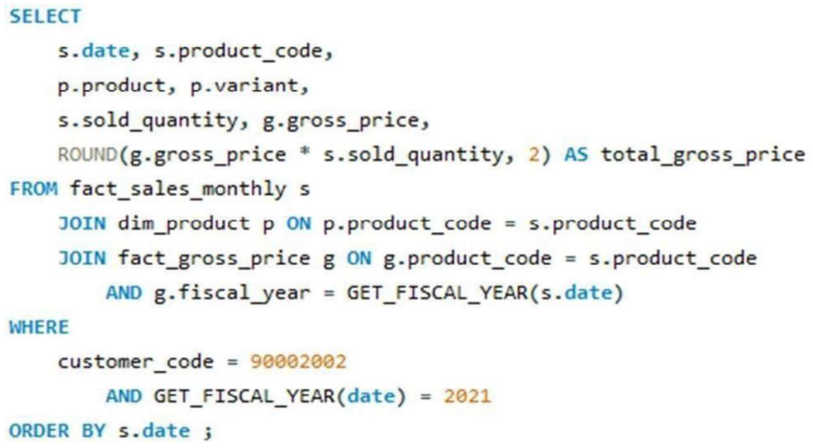
ATLIQ COMPANY INCREASE SALES

FINANCIAL ANALYTICS

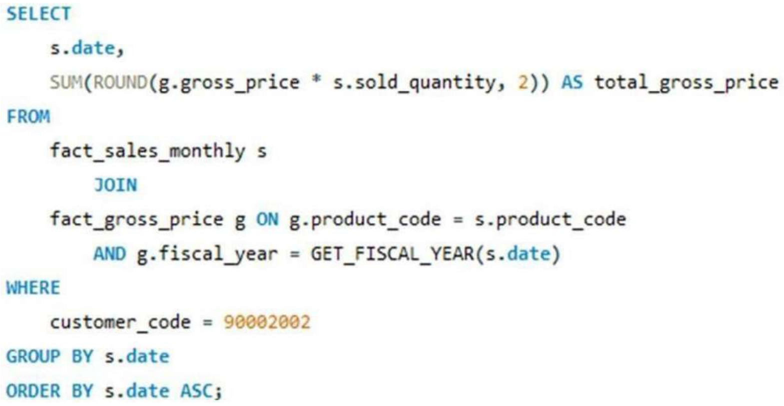
Monthly Product Transactions



```
SELECT
    s.date, s.product_code,
    p.product, p.variant,
    s.sold_quantity, g.gross_price,
    ROUND(g.gross_price * s.sold_quantity, 2) AS total_gross_price
FROM fact_sales_monthly s
    JOIN dim_product p ON p.product_code = s.product_code
    JOIN fact_gross_price g ON g.product_code = s.product_code
        AND g.fiscal_year = GET_FISCAL_YEAR(s.date)
WHERE
    customer_code = 90002002
        AND GET_FISCAL_YEAR(date) = 2021
ORDER BY s.date ;
```

date	product_code	product	variant	sold_quantity	gross_price	total_gross_price
2020-09-01	A0118150101	AQ Dracula HDD â€“ 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	Standard	202	19.0573	3849.57
2020-09-01	A0118150102	AQ Dracula HDD â€“ 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	Plus	162	21.4565	3475.95
2020-09-01	A0118150103	AQ Dracula HDD â€“ 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	Premium	193	21.7795	4203.44
2020-09-01	A0118150104	AQ Dracula HDD â€“ 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	Premium Plus	146	22.9729	3354.04
2020-09-01	A0219150201	AQ WereWolf NAS Internal Hard Drive HDD â€“ 8.89 cm	Standard	149	23.6987	3531.11
2020-09-01	A0219150202	AQ WereWolf NAS Internal Hard Drive HDD â€“ 8.89 cm	Plus	107	24.7312	2646.24
2020-09-01	A0220150203	AQ WereWolf NAS Internal Hard Drive HDD â€“ 8.89 cm	Premium	123	23.6154	2904.69
2020-09-01	A0320150301	AQ Zion Saga	Standard	146	23.7223	3463.46
2020-09-01	A0321150302	AQ Zion Saga	Plus	236	27.1027	6396.24
2020-09-01	A0321150303	AQ Zion Saga	Premium	137	28.0059	3836.81
2020-09-01	A0418150103	AQ Mforce Gen X	Standard 3	23	19.5235	449.04
2020-09-01	A0418150104	AQ Mforce Gen X	Plus 1	82	19.9239	1633.76
2020-09-01	A0418150105	AQ Mforce Gen X	Plus 2	86	20.0766	1726.59
2020-09-01	A0418150106	AQ Mforce Gen X	Plus 3	48	19.9365	956.95
2020-09-01	A0519150201	AQ Mforce Gen Y	Standard 1	138	22.3984	3090.98
2020-09-01	A0519150202	AQ Mforce Gen Y	Standard 2	72	24.9298	1794.95
2020-09-01	A0519150203	AQ Mforce Gen Y	Standard 3	38	26.5871	1010.31
2020-09-01	A0519150204	AQ Mforce Gen Y	Plus 1	149	26.1081	3890.11
2020-09-01	A0519150205	AQ Mforce Gen Y	Plus 2	29	29.7008	861.32

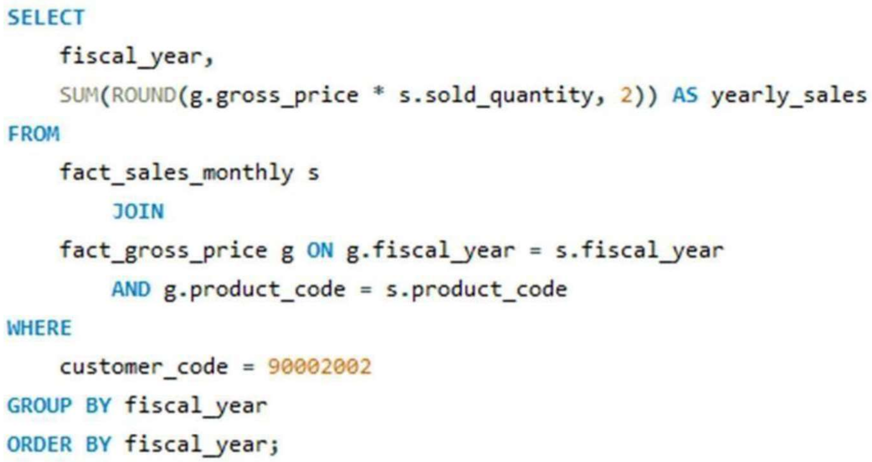
Monthly Total Sales Amount



```
SELECT
    s.date,
    SUM(ROUND(g.gross_price * s.sold_quantity, 2)) AS total_gross_price
FROM
    fact_sales_monthly s
        JOIN
    fact_gross_price g ON g.product_code = s.product_code
        AND g.fiscal_year = GET_FISCAL_YEAR(s.date)
WHERE
    customer_code = 90002002
GROUP BY s.date
ORDER BY s.date ASC;
```

date	total_gross_price
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	125409.9
2018-09-01	343337.14
2018-10-01	440562.1
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
2019-08-01	338108.87
2019-09-01	808250.42
2019-10-01	1092622.3
2019-12-01	1488174.01
2020-01-01	812929.77
2020-02-01	862762.82
2020-04-01	130520.91
2020-05-01	145049.08
2020-06-01	362545.18
2020-08-01	799327.63
2020-09-01	2296919.58
2020-10-01	3109316.96
2020-12-01	4078790.04
2021-01-01	2303086.42
2021-02-01	2355170.55
2021-04-01	2253575.01
2021-05-01	2181587.87
2021-06-01	2288587.49
2021-08-01	2349478.81
2021-09-01	11192823.18
2021-10-01	13908229.35
2021-12-01	19537146.58

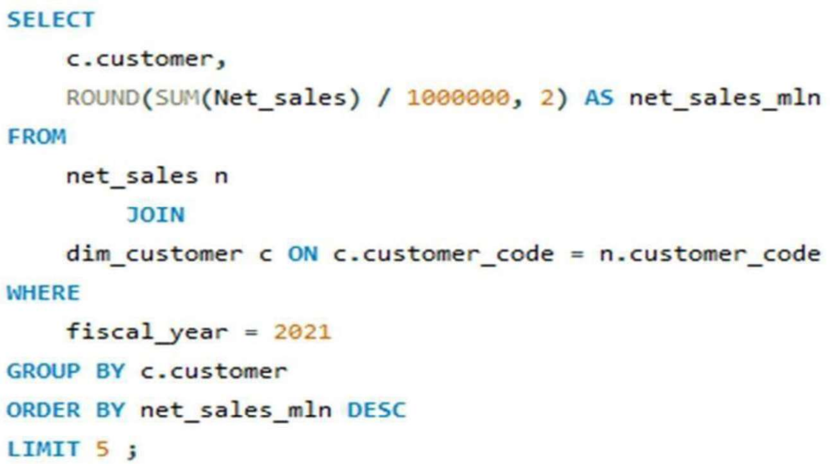
Yearly Total Sales Amount



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

fiscal_year	yearly_sales
2018	1324097.48
2019	3555079.19
2020	6502182.12
2021	23216512.73
2022	44638199.11

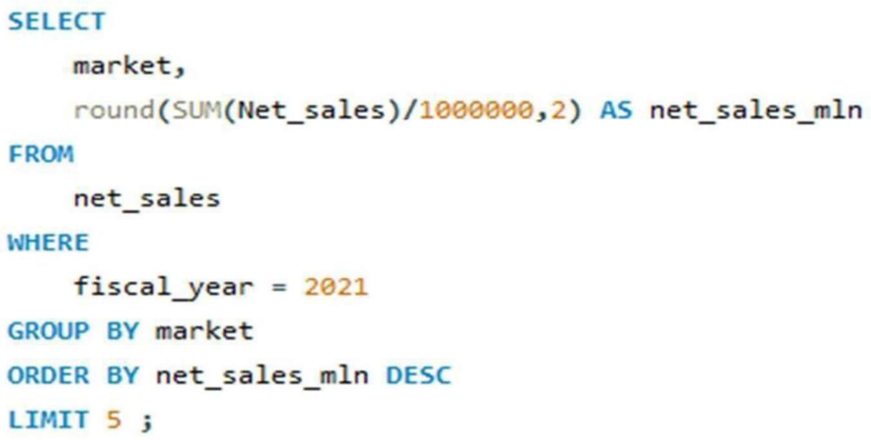
Top Customers (FY-2021)



```
SELECT
    c.customer,
    ROUND(SUM(Net_sales) / 1000000, 2) AS net_sales_mln
FROM
    net_sales n
        JOIN
    dim_customer c ON c.customer_code = n.customer_code
WHERE
    fiscal_year = 2021
GROUP BY c.customer
ORDER BY net_sales_mln DESC
LIMIT 5 ;
```

customer	net_sales_mln
Amazon	109.03
Atliq Exclusive	79.92
Atliq e Store	70.31
Sage	27.07
Flipkart	25.25

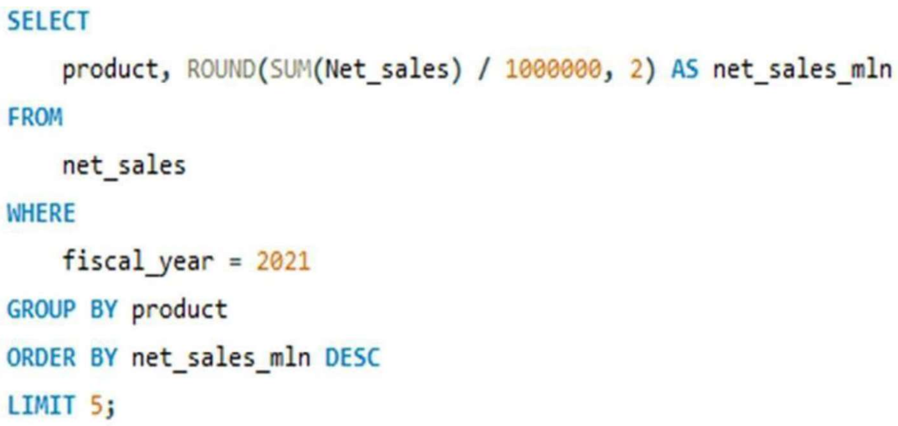
Top Markets (FY-2021)



```sql
SELECT
    market,
    round(SUM(Net_sales)/1000000,2) AS net_sales_mln
FROM
    net_sales
WHERE
    fiscal_year = 2021
GROUP BY market
ORDER BY net_sales_mln DESC
LIMIT 5 ;
```

market	net_sales_mln
India	210.67
USA	132.05
South Korea	64.01
Canada	45.89
United Kingdom	44.73

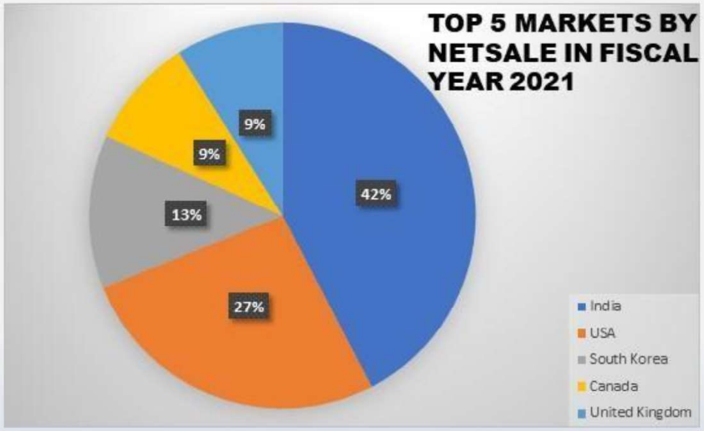
Top Products (FY-2021)

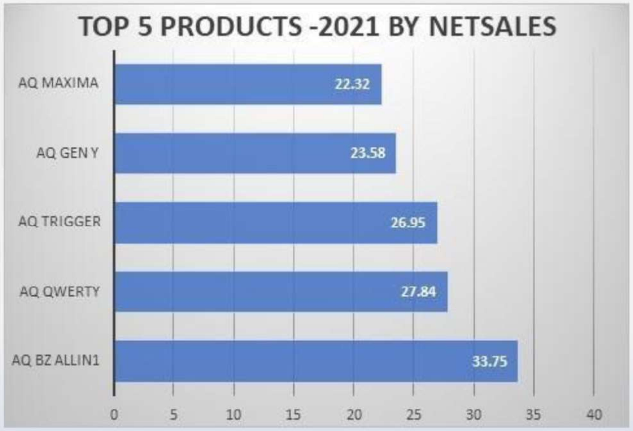


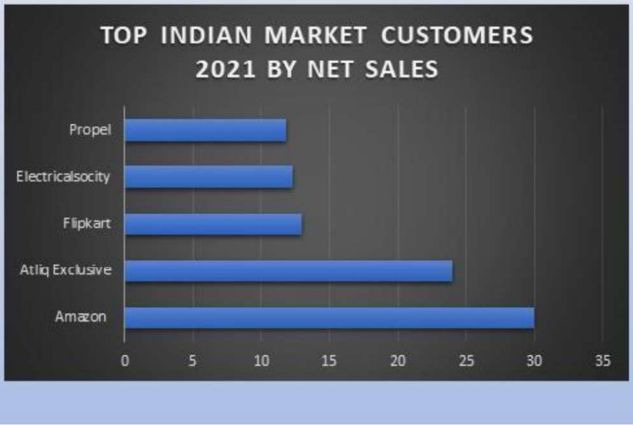
```sql
SELECT
    product, ROUND(SUM(Net_sales) / 1000000, 2) AS net_sales_mln
FROM
    net_sales
WHERE
    fiscal_year = 2021
GROUP BY product
ORDER BY net_sales_mln DESC
LIMIT 5;
```

product	net_sales_mln
AQ BZ Allin1	33.75
AQ Qwerty	27.84
AQ Trigger	26.95
AQ Gen Y	23.58
AQ Maxima	22. 32

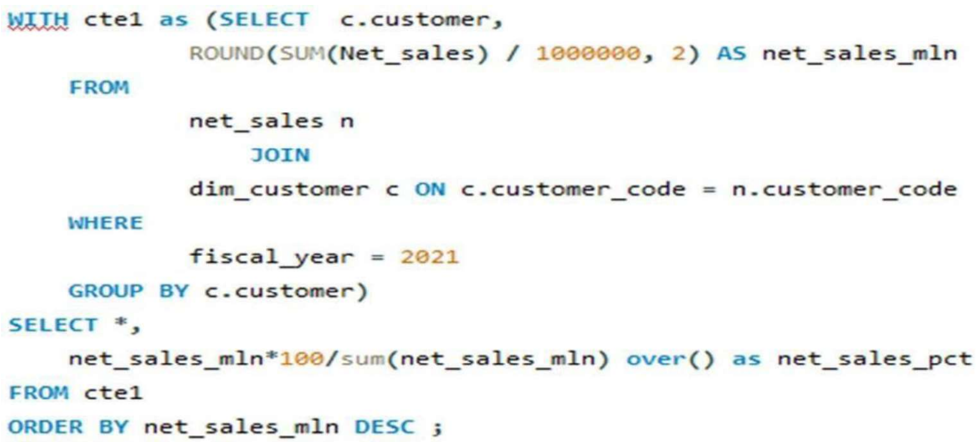
CHARTS REPRESNTATION







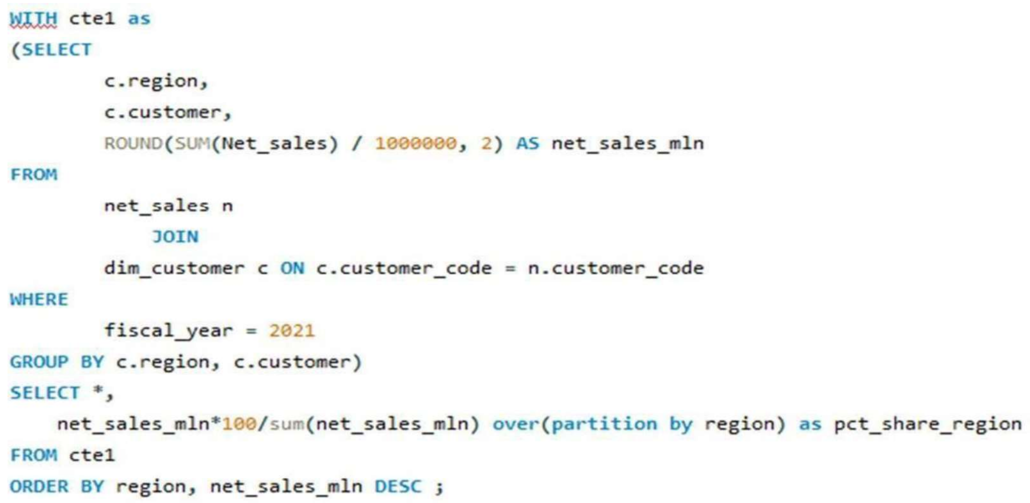
Net Sales Global Market Share % (FY-2021)



```
WITH cte1 as (SELECT  c.customer,
            ROUND(SUM(Net_sales) / 1000000, 2) AS net_sales_mln
    FROM
            net_sales n
                JOIN
            dim_customer c ON c.customer_code = n.customer_code
    WHERE
            fiscal_year = 2021
    GROUP BY c.customer)
SELECT *,
    net_sales_mln*100/sum(net_sales_mln) over() as net_sales_pct
FROM cte1
ORDER BY net_sales_mln DESC ;
```

customer	net_sales_mln	net_sales_pct
Amazon	109.03	13.23
Atliq Exclusive	79.92	9.7
Atliq e Store	70.31	8.53
Sage	27.07	3.29
Flipkart	25.25	3.06
Leader	24.52	2.98
Neptune	21.01	2.55
Ebay	19.88	2.41
Electricalsocity	16.25	1.97
Synthetic	16.1	1.95
Electricalslytical	15.64	1.9
Acclaimed Stores	14.32	1.74
Propel	14.14	1.72
Novus	12.91	1.57
Expression	12.9	1.57
Reliance Digital	12.75	1.55
walmart	12.63	1.53
Costco	12.19	1.48

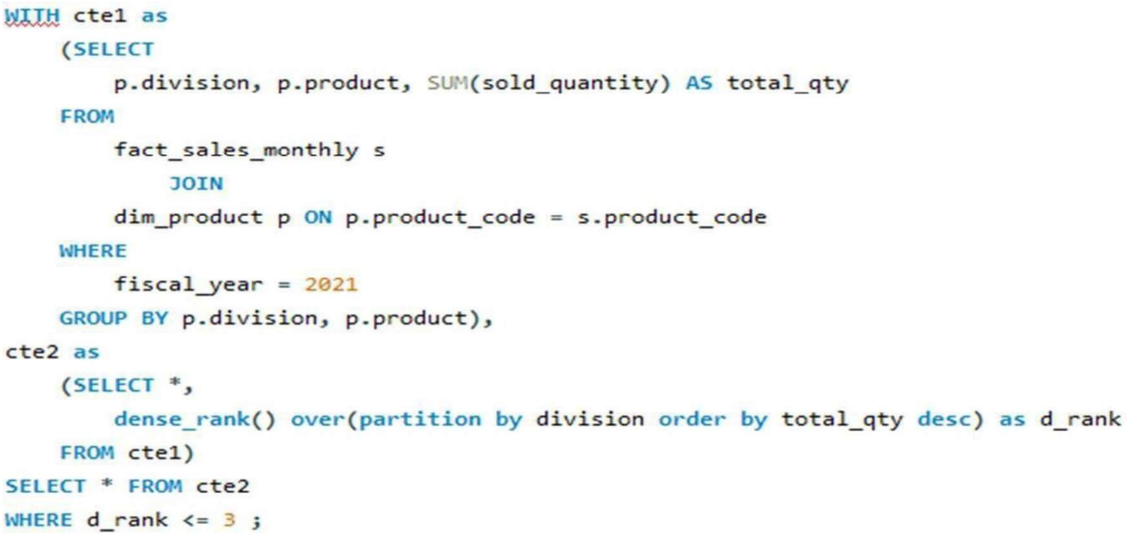
Net Sales % Share by Region (FY-2021)



```
WITH cte1 as
(SELECT
        c.region,
        c.customer,
        ROUND(SUM(Net_sales) / 1000000, 2) AS net_sales_mln
FROM
        net_sales n
            JOIN
        dim_customer c ON c.customer_code = n.customer_code
WHERE
        fiscal_year = 2021
GROUP BY c.region, c.customer)
SELECT *,
    net_sales_mln*100/sum(net_sales_mln) over(partition by region) as pct_share_region
FROM cte1
ORDER BY region, net_sales_mln DESC ;
```

region	customer	net_sales_mln	pct_share_region
APAC	Amazon	57.41	12.99
APAC	Atliq Exclusive	51.58	11.67
APAC	Atliq e Store	36.97	8.36
APAC	Leader	24.52	5.55
APAC	Sage	22.85	5.17
EU	Atliq e Store	19.83	9.87
EU	Amazon	19.77	9.84
EU	Atliq Exclusive	13.39	6.67
EU	UniEuro	9.63	4.8
EU	Expert	8.38	4.17
LATAM	Amazon	1.54	48.73
LATAM	Atliq e Store	1.09	34.49
LATAM	Electricalsbea Stores	0.53	16.77
NA	Amazon	30.31	17.03
NA	Atliq Exclusive	14.95	8.4
NA	walmart	12.63	7.1
NA	Atliq e Store	12.42	6.98
NA	Costco	12.19	6.85

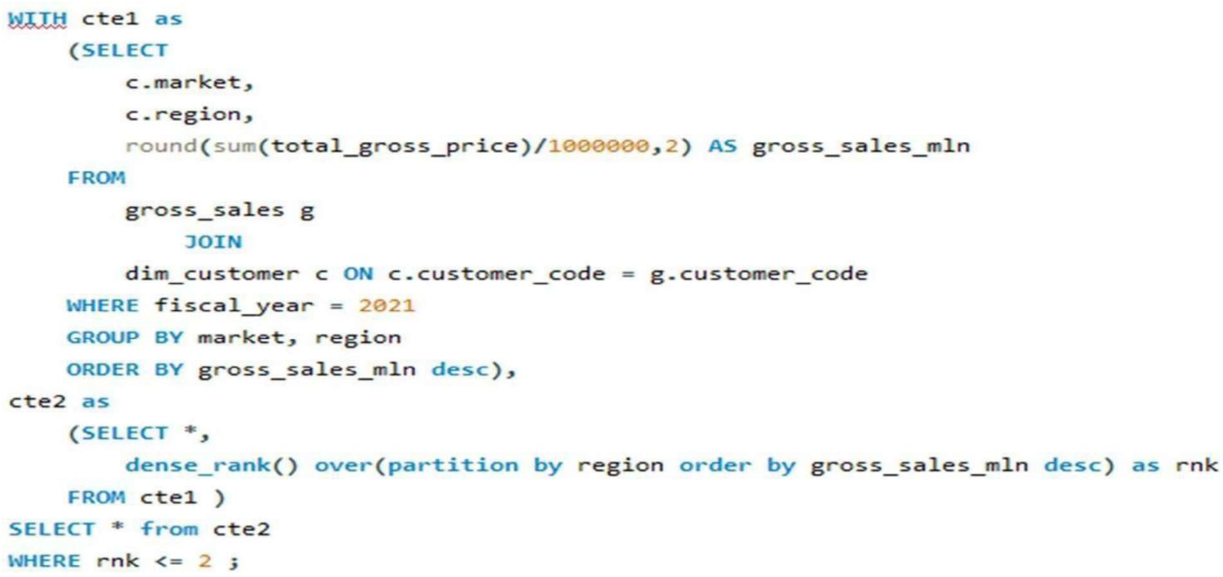
Top Products in each Division (FY-2021)



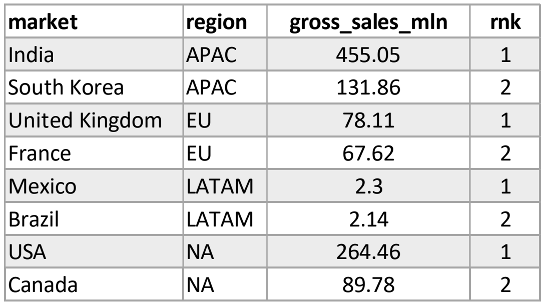
```
WITH cte1 as
    (SELECT
        p.division, p.product, SUM(sold_quantity) AS total_qty
    FROM
        fact_sales_monthly s
            JOIN
        dim_product p ON p.product_code = s.product_code
    WHERE
        fiscal_year = 2021
    GROUP BY p.division, p.product),
cte2 as
    (SELECT *,
        dense_rank() over(partition by division order by total_qty desc) as d_rank
    FROM cte1)
SELECT * FROM cte2
WHERE d_rank <= 3 ;
```

division	product	total_qty	d_rank
N & S	AQ Pen Drive DRC	2034569	1
N & S	AQ Digit SSD	1240149	2
N & S	AQ Clx1	1238683	3
P & A	AQ Gamers Ms	2477098	1
P & A	AQ Maxima Ms	2461991	2
P & A	AQ Master wireless x1 Ms	2448784	3
PC	AQ Digit	135092	1
PC	AQ Gen Y	135031	2
PC	AQ Elite	134431	3

Top Markets Per Region by Gross Sales (FY-2021)



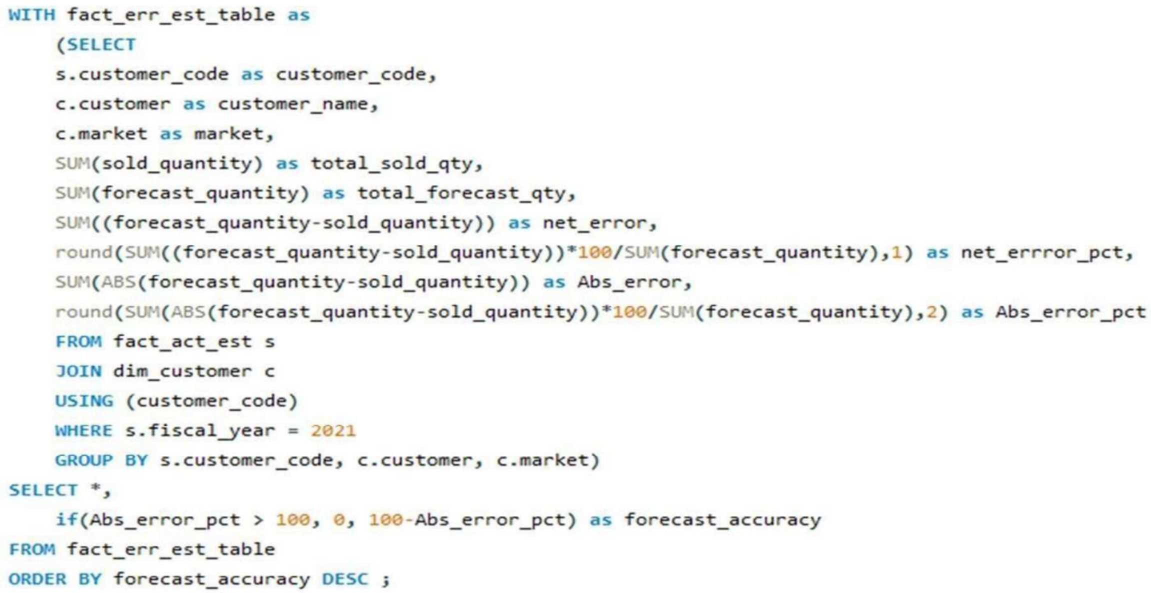
```
WITH cte1 as
    (SELECT
        c.market,
        c.region,
        round(sum(total_gross_price)/1000000,2) AS gross_sales_mln
    FROM
        gross_sales g
            JOIN
        dim_customer c ON c.customer_code = g.customer_code
    WHERE fiscal_year = 2021
    GROUP BY market, region
    ORDER BY gross_sales_mln desc),
cte2 as
    (SELECT *,
        dense_rank() over(partition by region order by gross_sales_mln desc) as rnk
    FROM cte1 )
SELECT * from cte2
WHERE rnk <= 2 ;
```



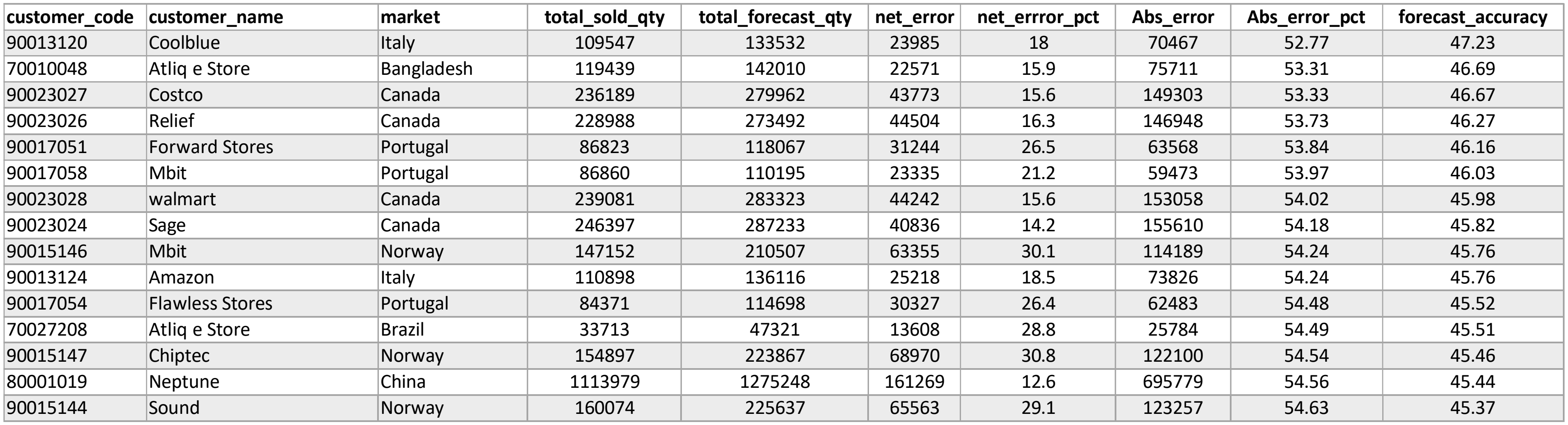
market	region	gross_sales_mln	rnk
India	APAC	455.05	1
South Korea	APAC	131.86	2
United Kingdom	EU	78.11	1
France	EU	67.62	2
Mexico	LATAM	2.3	1
Brazil	LATAM	2.14	2
USA	NA	264.46	1
Canada	NA	89.78	2

SUPPLY CHAIN ANALYTICS

Fact Accuracy Report (FY-2021)

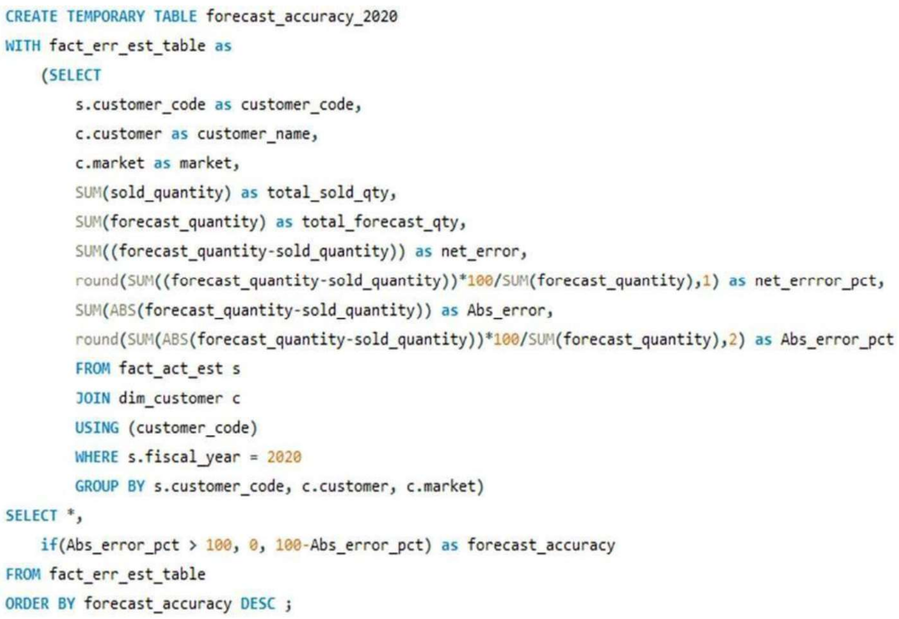


```
WITH fact_err_est_table as
    (SELECT
    s.customer_code as customer_code,
    c.customer as customer_name,
    c.market as market,
    SUM(sold_quantity) as total_sold_qty,
    SUM(forecast_quantity) as total_forecast_qty,
    SUM((forecast_quantity-sold_quantity)) as net_error,
    round(SUM((forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),1) as net_errror_pct,
    SUM(ABS(forecast_quantity-sold_quantity)) as Abs_error,
    round(SUM(ABS(forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),2) as Abs_error_pct
    FROM fact_act_est s
    JOIN dim_customer c
    USING (customer_code)
    WHERE s.fiscal_year = 2021
    GROUP BY s.customer_code, c.customer, c.market)
SELECT *,
    if(Abs_error_pct > 100, 0, 100-Abs_error_pct) as forecast_accuracy
FROM fact_err_est_table
ORDER BY forecast_accuracy DESC ;
```

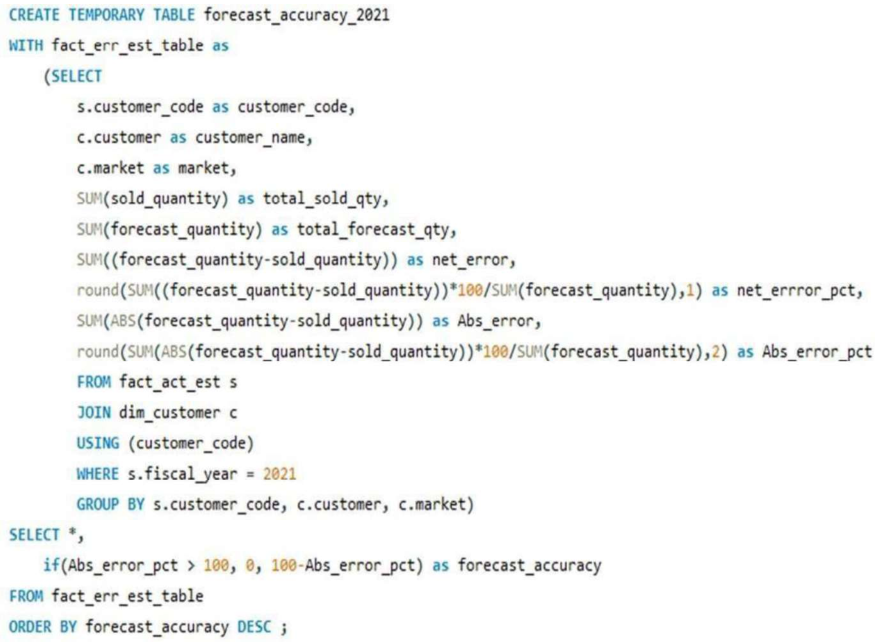


customer_code	customer_name	market	total_sold_qty	total_forecast_qty	net_error	net_errror_pct	Abs_error	Abs_error_pct	forecast_accuracy
90013120	Coolblue	Italy	109547	133532	23985	18	70467	52.77	47.23
70010048	Atliq e Store	Bangladesh	119439	142010	22571	15.9	75711	53.31	46.69
90023027	Costco	Canada	236189	279962	43773	15.6	149303	53.33	46.67
90023026	Relief	Canada	228988	273492	44504	16.3	146948	53.73	46.27
90017051	Forward Stores	Portugal	86823	118067	31244	26.5	63568	53.84	46.16
90017058	Mbit	Portugal	86860	110195	23335	21.2	59473	53.97	46.03
90023028	walmart	Canada	239081	283323	44242	15.6	153058	54.02	45.98
90023024	Sage	Canada	246397	287233	40836	14.2	155610	54.18	45.82
90015146	Mbit	Norway	147152	210507	63355	30.1	114189	54.24	45.76
90013124	Amazon	Italy	110898	136116	25218	18.5	73826	54.24	45.76
90017054	Flawless Stores	Portugal	84371	114698	30327	26.4	62483	54.48	45.52
70027208	Atliq e Store	Brazil	33713	47321	13608	28.8	25784	54.49	45.51
90015147	Chiptec	Norway	154897	223867	68970	30.8	122100	54.54	45.46
80001019	Neptune	China	1113979	1275248	161269	12.6	695779	54.56	45.44
90015144	Sound	Norway	160074	225637	65563	29.1	123257	54.63	45.37

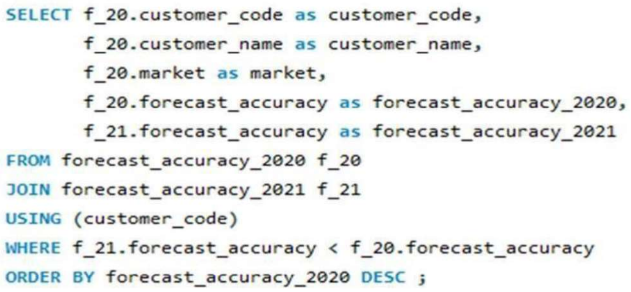
Fact Accuracy Report (FY 2020 vs FY 2021)



```sql
CREATE TEMPORARY TABLE forecast_accuracy_2020
WITH fact_err_est_table as
    (SELECT
        s.customer_code as customer_code,
        c.customer as customer_name,
        c.market as market,
        SUM(sold_quantity) as total_sold_qty,
        SUM(forecast_quantity) as total_forecast_qty,
        SUM((forecast_quantity-sold_quantity)) as net_error,
        round(SUM((forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),1) as net_errror_pct,
        SUM(ABS(forecast_quantity-sold_quantity)) as Abs_error,
        round(SUM(ABS(forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),2) as Abs_error_pct
        FROM fact_act_est s
        JOIN dim_customer c
        USING (customer_code)
        WHERE s.fiscal_year = 2020
        GROUP BY s.customer_code, c.customer, c.market)
SELECT *,
    if(Abs_error_pct > 100, 0, 100-Abs_error_pct) as forecast_accuracy
FROM fact_err_est_table
ORDER BY forecast_accuracy DESC ;
```



```sql
CREATE TEMPORARY TABLE forecast_accuracy_2021
WITH fact_err_est_table as
    (SELECT
        s.customer_code as customer_code,
        c.customer as customer_name,
        c.market as market,
        SUM(sold_quantity) as total_sold_qty,
        SUM(forecast_quantity) as total_forecast_qty,
        SUM((forecast_quantity-sold_quantity)) as net_error,
        round(SUM((forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),1) as net_errror_pct,
        SUM(ABS(forecast_quantity-sold_quantity)) as Abs_error,
        round(SUM(ABS(forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),2) as Abs_error_pct
        FROM fact_act_est s
        JOIN dim_customer c
        USING (customer_code)
        WHERE s.fiscal_year = 2021
        GROUP BY s.customer_code, c.customer, c.market)
SELECT *,
    if(Abs_error_pct > 100, 0, 100-Abs_error_pct) as forecast_accuracy
FROM fact_err_est_table
ORDER BY forecast_accuracy DESC ;
```



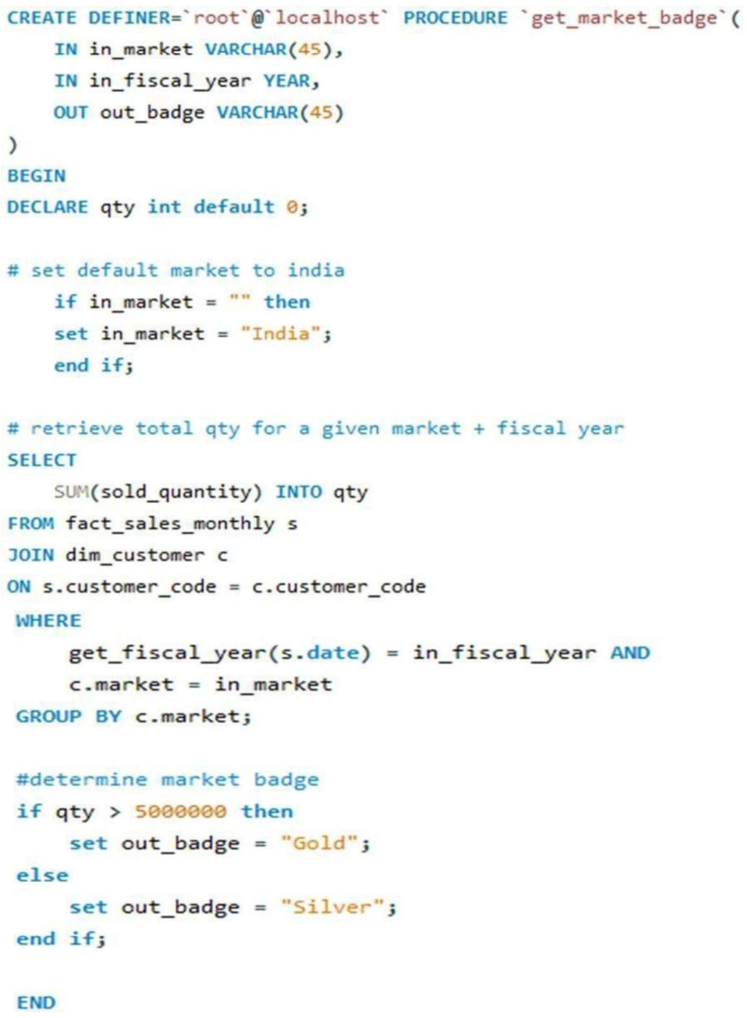
```sql
SELECT f_20.customer_code as customer_code,
       f_20.customer_name as customer_name,
       f_20.market as market,
       f_20.forecast_accuracy as forecast_accuracy_2020,
       f_21.forecast_accuracy as forecast_accuracy_2021
FROM forecast_accuracy_2020 f_20
JOIN forecast_accuracy_2021 f_21
USING (customer_code)
WHERE f_21.forecast_accuracy < f_20.forecast_accuracy
ORDER BY forecast_accuracy_2020 DESC ;
```

Fact Accuracy Report (FY 2020 vs FY 2021) (contd…)

customer_code	customer_name	market	forecast_accuracy_2020	forecast_accuracy_2021
70006158	Atliq e Store	Philiphines	42.65	24.49
70008170	Atliq e Store	Australia	40.96	38.74
90005161	Zone	Pakistan	40.08	37.1
90014140	Radio Popular	Netherlands	38.53	0
90008166	Sound	Australia	38.51	36.79
70014143	Atliq e Store	Netherlands	38.32	0
90004062	Flawless Stores	Japan	38.22	32.56
90014137	Media Markt	Netherlands	37.85	0
90014138	Mbit	Netherlands	37.83	0
70004069	Atliq Exclusive	Japan	37.62	32.09
90014136	Reliance Digital	Netherlands	37.59	0
80006154	Synthetic	Philiphines	37.49	24.63
70014142	Atliq Exclusive	Netherlands	37.43	0
90014141	Amazon	Netherlands	37.39	0
90005160	Nomad Stores	Pakistan	37.3	37.29
90006156	Amazon	Philiphines	37.21	27.94
90008164	Digimarket	Australia	37.15	36.01
90006153	Insight	Philiphines	37.11	26.88
90016173	Expert	Poland	36.76	35.26
80006155	Novus	Philiphines	36.59	25.28
90019200	Sorefoz	Sweden	36.41	24.14
70016177	Atliq Exclusive	Poland	36.22	34.87
90004066	Surface Stores	Japan	36.05	33.51
70006157	Atliq Exclusive	Philiphines	35.84	25.2

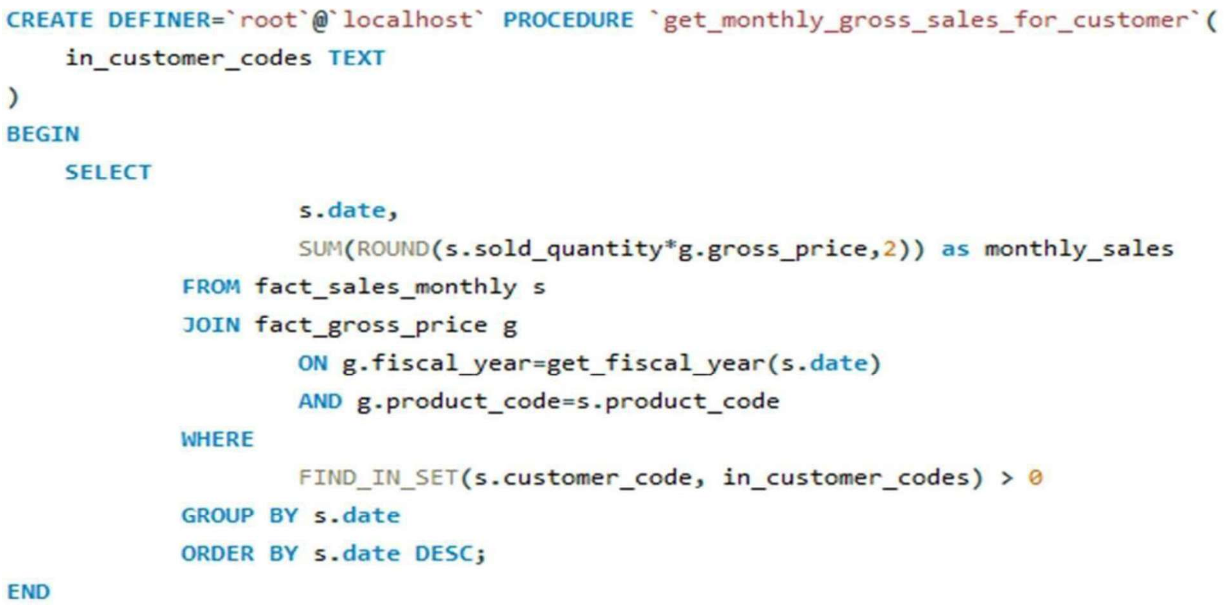
Stored Procedures

Stored Procedure to get Market badge



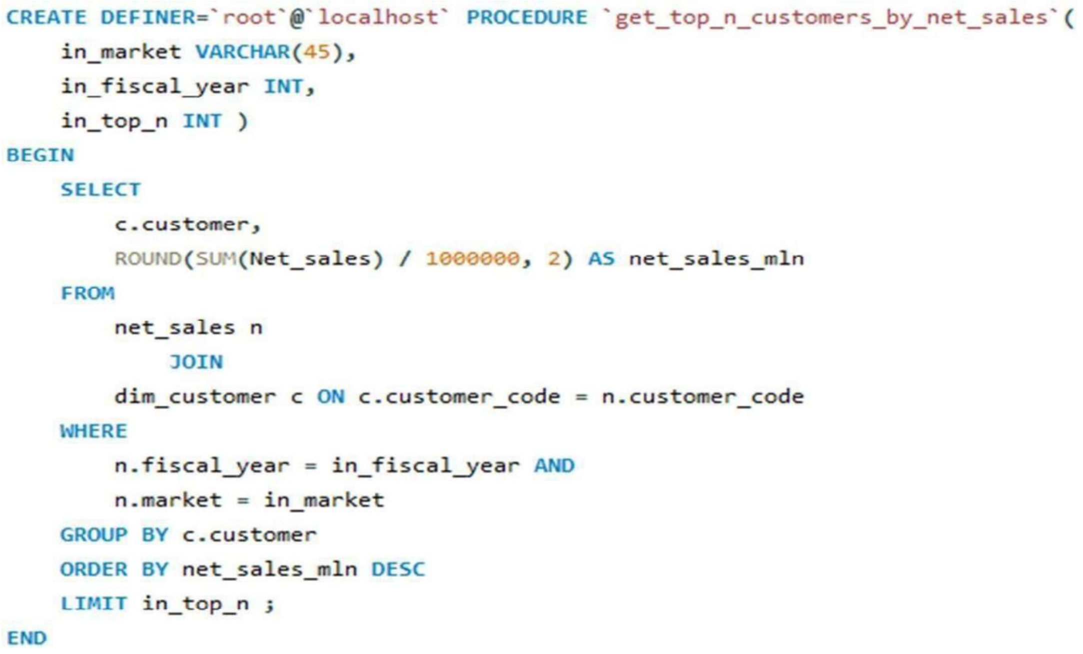
```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_market_badge`(
    IN in_market VARCHAR(45),
    IN in_fiscal_year YEAR,
    OUT out_badge VARCHAR(45)
)
BEGIN
DECLARE qty int default 0;

# set default market to india
    if in_market = "" then
    set in_market = "India";
    end if;

# retrieve total qty for a given market + fiscal year
SELECT
    SUM(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
 GROUP BY c.market;

 #determine market badge
 if qty > 5000000 then
     set out_badge = "Gold";
 else
     set out_badge = "Silver";
 end if;

 END
```

Stored Procedure to get Monthly Gross Sales for Customer



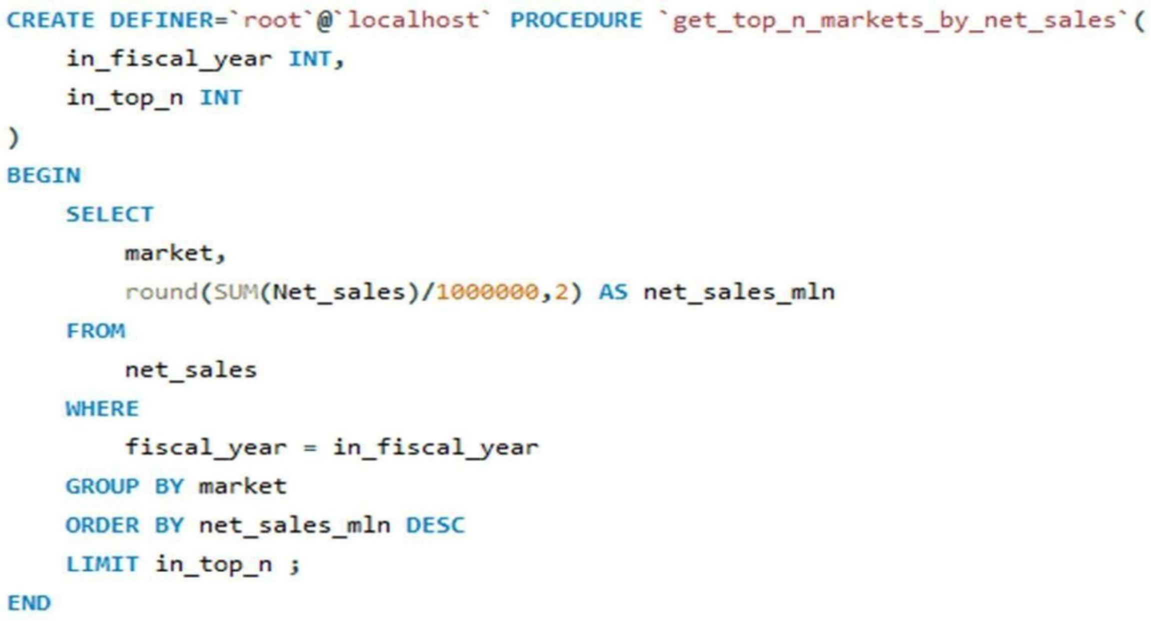
```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_monthly_gross_sales_for_customer`(
    in_customer_codes TEXT
)
BEGIN
    SELECT
                    s.date,
                    SUM(ROUND(s.sold_quantity*g.gross_price,2)) as monthly_sales
            FROM fact_sales_monthly s
            JOIN fact_gross_price 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 s.date
            ORDER BY s.date DESC;
END
```

Stored Procedure to get Top N Customers by Net Sales



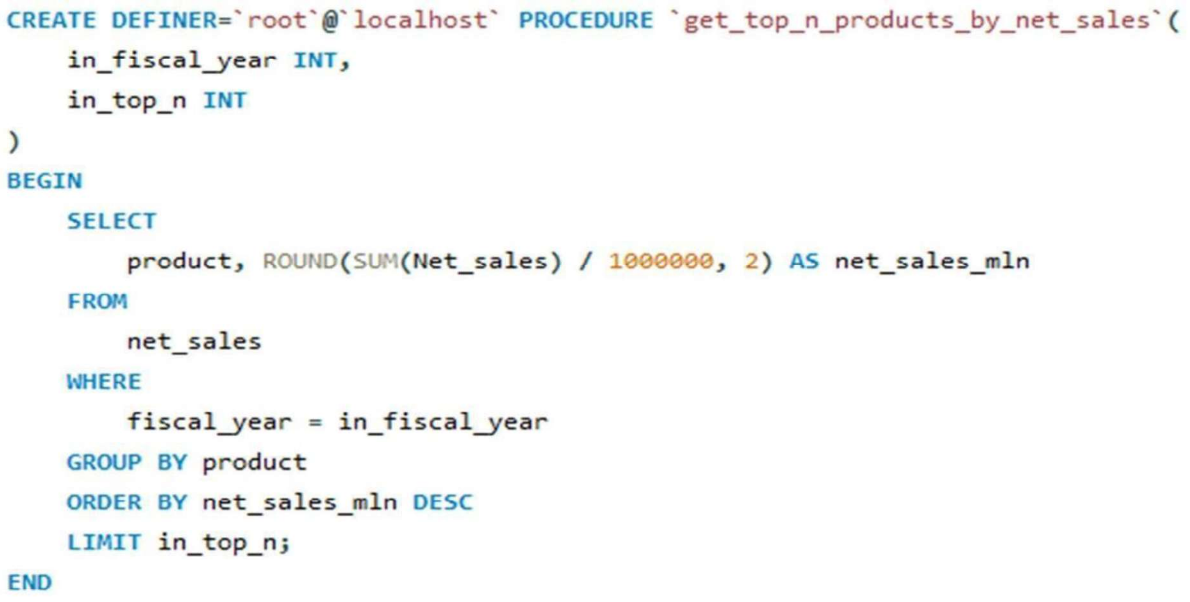
```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_top_n_customers_by_net_sales`(
    in_market VARCHAR(45),
    in_fiscal_year INT,
    in_top_n INT )
BEGIN
    SELECT
        c.customer,
        ROUND(SUM(Net_sales) / 1000000, 2) AS net_sales_mln
    FROM
        net_sales n
            JOIN
        dim_customer c ON c.customer_code = n.customer_code
    WHERE
        n.fiscal_year = in_fiscal_year AND
        n.market = in_market
    GROUP BY c.customer
    ORDER BY net_sales_mln DESC
    LIMIT in_top_n ;
END
```

Stored Procedure to get Top N Markets by Net Sales



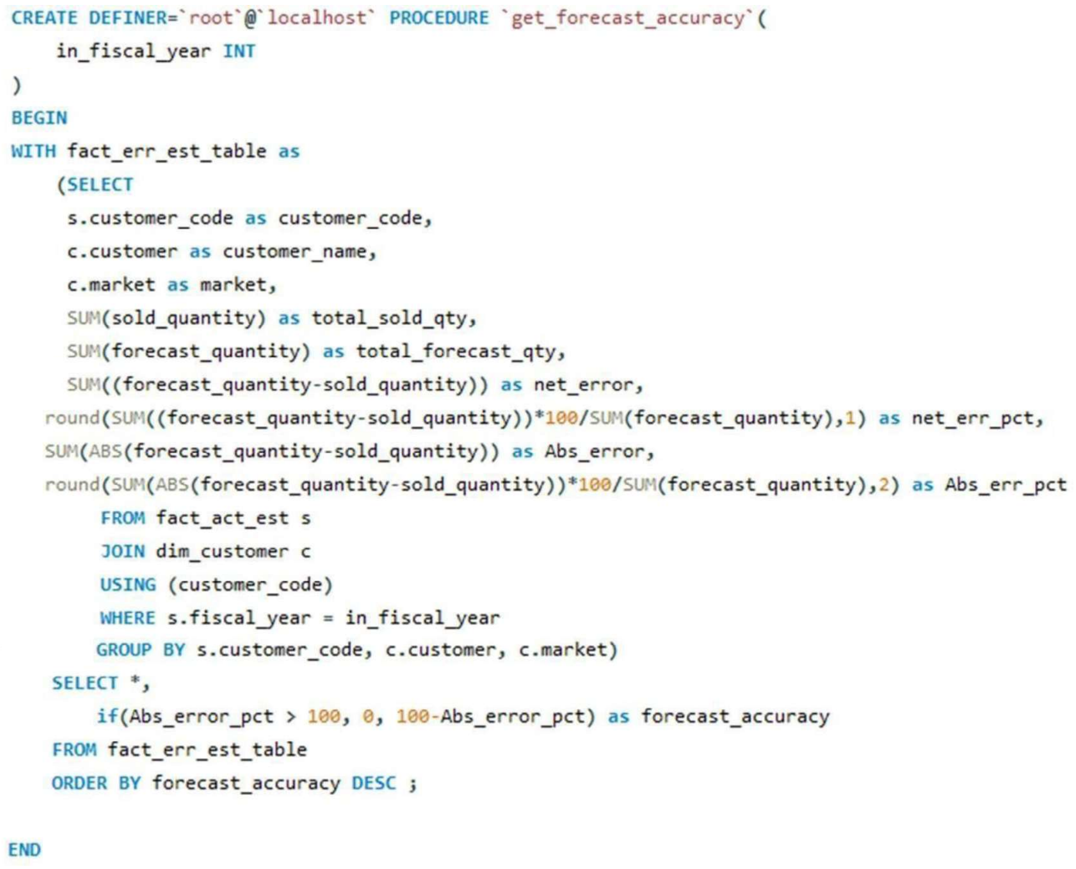
```sql
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_top_n_markets_by_net_sales`(
    in_fiscal_year INT,
    in_top_n INT
)
BEGIN
    SELECT
        market,
        round(SUM(Net_sales)/1000000,2) AS net_sales_mln
    FROM
        net_sales
    WHERE
        fiscal_year = in_fiscal_year
    GROUP BY market
    ORDER BY net_sales_mln DESC
    LIMIT in_top_n ;
END
```

Stored Procedure to get Top N Products by Net Sales



```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_top_n_products_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
        net_sales
    WHERE
        fiscal_year = in_fiscal_year
    GROUP BY product
    ORDER BY net_sales_mln DESC
    LIMIT in_top_n;
END
```

Stored Procedure to get Fact Accuracy



```
CREATE DEFINER=`root`@`localhost` PROCEDURE `get_forecast_accuracy`(
    in_fiscal_year INT
)
BEGIN
WITH fact_err_est_table as
    (SELECT
     s.customer_code as customer_code,
     c.customer as customer_name,
     c.market as market,
     SUM(sold_quantity) as total_sold_qty,
     SUM(forecast_quantity) as total_forecast_qty,
     SUM((forecast_quantity-sold_quantity)) as net_error,
   round(SUM((forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),1) as net_err_pct,
   SUM(ABS(forecast_quantity-sold_quantity)) as Abs_error,
   round(SUM(ABS(forecast_quantity-sold_quantity))*100/SUM(forecast_quantity),2) as Abs_err_pct
        FROM fact_act_est s
        JOIN dim_customer c
        USING (customer_code)
        WHERE s.fiscal_year = in_fiscal_year
        GROUP BY s.customer_code, c.customer, c.market)
    SELECT *,
        if(Abs_error_pct > 100, 0, 100-Abs_error_pct) as forecast_accuracy
    FROM fact_err_est_table
    ORDER BY forecast_accuracy DESC ;

END
```

THANKYOU