SQL QUERIES AND OUTPUT:

1. Product-wise sales report for Croma India - Fiscal Year 2021:

```
SELECT
    s.date,
    s.product_code,
    s.sold_quantity,
    p.product,
    p.variant,
    g.gross_price,
    g.gross_price * s.sold_quantity AS gross_price_total
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(s.date) = 2021
ORDER BY
    s.date ASC
LIMIT
    1000000;
```

2. Gross Monthly Sales Report for Croma:

```
SELECT
   s.date,
   SUM(g.gross_price * s.sold_quantity) AS gross_price_total
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;
```

3. Annual Gross Sales Report for Croma India:

```
SELECT
   get_fiscal_year(s.date) AS fiscal_year,
   ROUND(SUM(g.gross_price * s.sold_quantity), 2) AS total_gross_sales_amount
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
    get_fiscal_year(s.date)
ORDER BY
    fiscal_year;
```

4. Top Markets for Fiscal Year 2021:

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

5. Top Customers for Fiscal Year 2021:

```
SELECT
    c.customer,
    ROUND(SUM(net_sales) / 1000000, 2) AS net_sales_mln
FROM
    net_sales s
JOIN
```

```
    dim_customer c ON c.customer_code = s.customer_code
WHERE
    s.fiscal_year = 2021
GROUP BY
    c.customer
ORDER BY
    net_sales_mln DESC
LIMIT
    5;
```

6. Net Sales % Share by Customers:

```
WITH cte1 AS (
    SELECT
        c.customer,
        ROUND(SUM(s.net_sales) / 1000000, 2) AS net_sales_mln
    FROM
        net_sales s
    JOIN
        dim_customer c ON c.customer_code = s.customer_code
    WHERE
        s.fiscal_year = 2021
    GROUP BY
        c.customer
)

SELECT
    *,
```

```
  ROUND(net_sales_mln * 100 / SUM(net_sales_mln) OVER(), 2) AS pct
FROM
  cte1
ORDER BY
  net_sales_mln DESC
LIMIT 10;
```

7. Net Sales % Share by Region - APAC:

```
WITH cte1 AS (
  SELECT
    c.customer,
    c.region,
    ROUND(SUM(s.net_sales) / 1000000, 2) AS net_sales_mln
  FROM
    net_sales s
  JOIN
    dim_customer c ON c.customer_code = s.customer_code
  WHERE
    s.fiscal_year = 2021
  GROUP BY
    c.customer, c.region
)

SELECT
  *,
  ROUND(net_sales_mln * 100 / SUM(net_sales_mln) OVER(PARTITION BY region), 2) AS pct
FROM
```

```
    cte1
ORDER BY
    region, net_sales_mln DESC;
```

8. Top 2 Markets in each Region by Gross Sales:

```
WITH cte1 AS (
    SELECT
        c.market,
        c.region,
        ROUND(SUM(s.gross_price_total) / 1000000, 2) AS gross_sales_mln
    FROM
        gross_sales s
    JOIN
        dim_customer c ON c.customer_code = s.customer_code
    WHERE
        fiscal_year = 2021
    GROUP BY
        c.market, c.region
    ORDER BY
        gross_sales_mln DESC
),
cte2 AS (
    SELECT
        *,
        DENSE_RANK() OVER(PARTITION BY region ORDER BY gross_sales_mln DESC) AS drnk
    FROM
        cte1
```

```
)

SELECT
  *
FROM
  cte2
WHERE
  drnk <= 2;
```

9. Supply Chain Forecasted Quantity:

```
WITH forecast_err_table AS (
  SELECT
    s.customer_code AS customer_code,
    c.customer AS customer_name,
    c.market AS market,
    SUM(s.sold_quantity) AS total_sold_qty,
    SUM(s.forecast_quantity) AS total_forecast_qty,
    SUM(s.forecast_quantity - s.sold_quantity) AS net_error,
    ROUND(SUM(s.forecast_quantity - s.sold_quantity) * 100 / SUM(s.forecast_quantity), 2) AS
net_error_pct,
    SUM(ABS(s.forecast_quantity - s.sold_quantity)) AS abs_error,
    ROUND(SUM(ABS(s.forecast_quantity - sold_quantity)) * 100 / SUM(s.forecast_quantity), 2)
AS abs_error_pct
  FROM
    fact_act_est s
  JOIN
    dim_customer c ON s.customer_code = c.customer_code
  WHERE
```

```
        s.fiscal_year = 2021
    GROUP BY
        s.customer_code
)

SELECT
    *,
    IF(abs_error_pct > 100, 0, 100.0 - abs_error_pct) AS forecast_accuracy
FROM
    forecast_err_table
ORDER BY
    forecast_accuracy DESC;
```