SQL QUERIES AND OUTPUT:

Ad Hoc Request 1:

Objective: Retrieve the list of markets where the customer *"Atliq Exclusive"* operates within the APAC region.

```
SELECT DISTINCT market
FROM dim_customer
WHERE customer = 'Atliq Exclusive'
  AND region = 'APAC';
```

Ad Hoc Request 2:

Objective: Calculate the percentage increase in unique products from 2020 to 2021. The final output should include the fields: unique_products_2020, unique_products_2021, and percentage_chg.

```
WITH X AS (
    SELECT COUNT(DISTINCT product_code) AS unique_products_2020
    FROM fact_sales_monthly
    WHERE fiscal_year = 2020
),
Y AS (
    SELECT COUNT(DISTINCT product_code) AS unique_products_2021
    FROM fact_sales_monthly
    WHERE fiscal_year = 2021
```

```
)
SELECT
    X.unique_products_2020,
    Y.unique_products_2021,
    ROUND(((Y.unique_products_2021 - X.unique_products_2020) / X.unique_products_2020) *
100, 2) AS percentage_chg
FROM X, Y;
```

Ad Hoc Request 3:

Objective: Generate a report showing the unique product counts for each segment, sorted in descending order by product count. The final output should include two fields: segment and product_count.

```
SELECT
    segment,
    COUNT(DISTINCT product_code) AS product_count
FROM
    dim_product
GROUP BY
    segment
ORDER BY
    product_count DESC;
```

Ad Hoc Request 4:

Objective: Identify the segment with the highest increase in unique products from 2020 to 2021. The final output should include the fields: segment, product_count_2020, product_count_2021, and difference.

```
WITH x AS (
    SELECT
        p.segment,
        COUNT(DISTINCT s.product_code) AS product_count_2020
    FROM
        dim_product p
    JOIN
        fact_sales_monthly s ON p.product_code = s.product_code
    WHERE
        s.fiscal_year = 2020
    GROUP BY
        p.segment
),
y AS (
    SELECT
        p.segment,
        COUNT(DISTINCT s.product_code) AS product_count_2021
    FROM
```

```
    dim_product p
  JOIN
    fact_sales_monthly s ON p.product_code = s.product_code
  WHERE
    s.fiscal_year = 2021
  GROUP BY
    p.segment
)
SELECT
  x.segment,
  x.product_count_2020,
  y.product_count_2021,
  ABS(x.product_count_2020 - y.product_count_2021) AS difference
FROM
  x
JOIN
  y ON x.segment = y.segment
ORDER BY
  difference DESC;
```

Ad Hoc Request 5:

Objective: Retrieve products with the highest and lowest manufacturing

costs. The final output should include the fields: product_code, product, and manufacturing_cost.

```
SELECT
    m.product_code,
    p.product,
    m.manufacturing_cost
FROM
    fact_manufacturing_cost m
JOIN
    dim_product p USING (product_code)
WHERE
    m.manufacturing_cost = (SELECT MAX(manufacturing_cost) FROM
fact_manufacturing_cost)
    OR m.manufacturing_cost = (SELECT MIN(manufacturing_cost) FROM
fact_manufacturing_cost)
ORDER BY
    m.manufacturing_cost DESC;
```

Ad Hoc Request 6:

Objective: Generate a report of the top 5 customers who received the highest average pre-invoice discount percentage in the Indian market for the fiscal year 2021. The final output should include the fields: customer_code, customer, and average_discount_percentage.

```
SELECT
    i.customer_code,
    c.customer,
    ROUND(AVG(i.pre_invoice_discount_pct) * 100, 2) AS average_discount_percentage
FROM
    fact_pre_invoice_deductions i
JOIN
    dim_customer c USING (customer_code)
WHERE
    fiscal_year = 2021
    AND c.market = 'India'
GROUP BY
    i.customer_code,
    c.customer
ORDER BY
    average_discount_percentage DESC
LIMIT 5;
```

Ad Hoc Request 7:

Objective: Retrieve the complete report of gross sales amounts for the customer *"Atliq Exclusive"* for each month. This analysis will help identify low and high-performing months, aiding strategic decision-

making. The final report should include the columns: Month, Year, and Gross Sales Amount.

```
SELECT
    MONTHNAME(s.date) AS month,
    s.fiscal_year AS year,
    ROUND(SUM(g.gross_price * s.sold_quantity), 2) AS gross_sales_amt
FROM
    fact_sales_monthly s
JOIN
    dim_customer c USING (customer_code)
JOIN
    fact_gross_price g USING (product_code)
WHERE
    c.customer = 'Atliq Exclusive'
GROUP BY
    MONTHNAME(s.date),
    s.fiscal_year
ORDER BY
    s.fiscal_year;
```

Ad Hoc Request 8:

Objective: Identify which quarter of 2020 had the maximum total sold

quantity. The final output should include the fields: Quarter and Total Sold Quantity, sorted by the Total Sold Quantity in descending order.

```
SELECT
    CASE
        WHEN MONTH(date) IN (9, 10, 11) THEN 'Q1'
        WHEN MONTH(date) IN (12, 1, 2) THEN 'Q2'
        WHEN MONTH(date) IN (3, 4, 5) THEN 'Q3'
        ELSE 'Q4'
    END AS Quarter,
    SUM(sold_quantity) AS total_sold_qty
FROM
    fact_sales_monthly
WHERE
    fiscal_year = 2020
GROUP BY
    Quarter
ORDER BY
    total_sold_qty DESC;
```

Ad Hoc Request 9:

Objective: Identify which channel contributed the most to gross sales in the fiscal year 2021 and calculate the percentage of contribution. The

final output should include the fields: Channel, Gross Sales (MLN), and Percentage.

```
WITH x AS (
    SELECT
        c.channel,
        ROUND(SUM(g.gross_price * s.sold_quantity) / 1000000, 2) AS gross_sales_mln
    FROM
        fact_sales_monthly s
    JOIN
        dim_customer c USING (customer_code)
    JOIN
        fact_gross_price g USING (product_code)
    WHERE
        s.fiscal_year = 2021
    GROUP BY
        c.channel
)
SELECT
    channel,
    gross_sales_mln,
    ROUND((gross_sales_mln / (SELECT SUM(gross_sales_mln) FROM x)) * 100, 2) AS
percentage
FROM
```

```
    x
ORDER BY
    gross_sales_mln DESC;
```

Ad Hoc Request 10:

Objective: Retrieve the top 3 products in each division with the highest total sold quantities in fiscal year 2021. The final output should include the fields: Division and Product Code.

```
WITH x AS (
    SELECT
        P.division,
        S.product_code,
        P.product,
        SUM(S.sold_quantity) AS total_sold_quantity,
        RANK() OVER(PARTITION BY P.division ORDER BY SUM(S.sold_quantity) DESC) AS rank_order
    FROM
        dim_product P
    JOIN
        fact_sales_monthly S ON P.product_code = S.product_code
    WHERE
        S.fiscal_year = 2021
    GROUP BY
```

```
        P.division, S.product_code, P.product
)
SELECT
    division,
    product_code
FROM
    x
WHERE
    rank_order IN (1, 2, 3)
ORDER BY
    division, rank_order;
```