```sql
                                -- Business Analysis Queries
-- Total Sales Revenue Over Time (Monthly)
-- 1.1 Querying to retrieve monthly sales revenue for business trend analysis
SELECT
    FORMAT(sale_date, 'yyyy-MM') AS SaleMonth,
    SUM(total_sales) AS TotalRevenue
FROM sales
GROUP BY FORMAT(sale_date, 'yyyy-MM')
ORDER BY SaleMonth ASC;

GO


-- Top 10 Best-Selling Products by Revenue
-- 2.1 Querying to identify the top 10 products contributing highest revenue
SELECT
    p.product_name,
    SUM(s.total_sales) AS ProductRevenue
FROM sales s
INNER JOIN products p ON s.product_id = p.product_id
GROUP BY p.product_name
ORDER BY ProductRevenue DESC
OFFSET 0 ROWS FETCH NEXT 10 ROWS ONLY;

GO


-- Top 5 Sales Representatives by Revenue
-- 3.1 Querying to find which reps are generating most sales
SELECT
    sr.rep_name,
    SUM(s.total_sales) AS RepRevenue
FROM sales s
INNER JOIN sales_representatives sr ON s.rep_id = sr.rep_id
GROUP BY sr.rep_name
ORDER BY RepRevenue DESC
OFFSET 0 ROWS FETCH NEXT 5 ROWS ONLY;

GO


-- Sales by Region
-- 4.1 Analysing how each region is performing in sales
SELECT
    r.region_name,
    SUM(s.total_sales) AS region_revenue
FROM sales s
INNER JOIN sales_representatives sr ON s.rep_id = sr.rep_id
INNER JOIN regions r ON sr.region_id = r.region_id
GROUP BY r.region_name
ORDER BY region_revenue DESC;

GO
```

```sql
-- Product Categories Performance
-- 5.1 Understanding how much each product category contributes to total revenue
SELECT
    c.category_name,
    SUM(s.total_sales) AS category_revenue
FROM sales s
INNER JOIN products p ON s.product_id = p.product_id
INNER JOIN categories c ON p.category_id = c.category_id
GROUP BY c.category_name
ORDER BY category_revenue DESC;

-- Payment Method Usage Analysis
-- 6.1 Finding out which payment method is most popular
SELECT
    pm.method_name,
    COUNT(s.sale_id) AS payment_count,
    SUM(s.total_sales) AS total_sales_via_payment_method
FROM sales s
INNER JOIN payment_methods pm ON s.payment_method_id = pm.payment_method_id
GROUP BY pm.method_name
ORDER BY total_sales_via_payment_method DESC;

GO


-- Customer Type Revenue Contribution
-- 7.1 See how different types of customers contribute to sales
SELECT
    c.customer_type,
    COUNT(s.sale_id) AS number_of_sales,
    SUM(s.total_sales) AS revenue_by_customertype
FROM sales s
INNER JOIN customers c ON s.customer_id = c.customer_id
GROUP BY c.customer_type
ORDER BY revenue_by_customertype DESC;

GO


-- Pending and Cancelled Orders Summary
-- 8.1 Understanding the number and value of orders that are pending or
  cancelled
SELECT
    order_status,
    COUNT(sale_id) AS number_of_orders,
    SUM(total_sales) AS value_of_orders
FROM sales
WHERE order_status IN ('Pending', 'Cancelled')
GROUP BY order_status;

GO
```

```sql
-- Average Sale Value Per Representative
-- 9.1 Calculating average sale size per sales rep
SELECT
    sr.rep_name,
    AVG(s.total_sales) AS average_sale_value
FROM sales s
INNER JOIN sales_representatives sr ON s.rep_id = sr.rep_id
GROUP BY sr.rep_name
ORDER BY average_sale_value DESC;

GO


-- Month-on-Month Sales Growth
-- 10. Calculating month-to-month sales growth
WITH MonthlySales AS (
    SELECT
        YEAR(sale_date) AS sale_year,
        MONTH(sale_date) AS sale_month,
        SUM(total_sales) AS total_revenue
    FROM sales
    GROUP BY YEAR(sale_date), MONTH(sale_date)
)
SELECT
    a.sale_year,
    a.sale_month,
    a.total_revenue,
    CAST(ROUND(((a.total_revenue - b.total_revenue) / NULLIF(b.total_revenue,
      0)) * 100, 2) AS FLOAT) AS MoM_growth_percent
FROM MonthlySales a
LEFT JOIN MonthlySales b
    ON (a.sale_year = b.sale_year AND a.sale_month = b.sale_month + 1)
    OR (a.sale_year = b.sale_year + 1 AND a.sale_month = 1 AND b.sale_month =
      12)
ORDER BY a.sale_year, a.sale_month;

GO
```