```
SELECT *
FROM world_ecommerce.ecommerce_dataset;

-- 1 TOTAL Revenue
SELECT SUM(total_revenue) AS total_sales
FROM world_ecommerce.ecommerce_dataset;

-- 2 Average Order Value (AOV)
SELECT AVG(total_revenue) AS average_order_value
FROM world_ecommerce.ecommerce_dataset;


-- 3 Top Product_Category Sales
SELECT product_category, SUM(total_revenue) AS Category_Sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY product_category
ORDER BY Category_Sales DESC
;

-- 4 Product category number of sale and revenue generated
SELECT
   product_category,                    -- want to know the number od sale that generate this
revenue
   COUNT(*) AS number_of_sales,
   SUM(total_revenue) AS category_sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY product_category
ORDER BY category_sales DESC;

-- 5 Revenue Generated by month per product Category
SELECT
   MONTHNAME(order_date) AS month,
   product_category,
   SUM(total_revenue) AS monthly_category_revenue
FROM world_ecommerce.ecommerce_dataset
GROUP BY
   MONTH(order_date),
   MONTHNAME(order_date),
   product_category
ORDER BY
   MONTH(order_date),
   monthly_category_revenue DESC;

   SELECT
```

```
    product_category,
    COUNT(*) AS number_of_sales,
    SUM(total_revenue) AS category_revenue,
    SUM(total_revenue * 0.30) AS estimated_profit
FROM world_ecommerce.ecommerce_dataset
GROUP BY product_category
ORDER BY estimated_profit DESC;




--  Total number of unit sold
SELECT
    product_id,
    product_category,
    SUM(quantity) AS total_units_sold
FROM world_ecommerce.ecommerce_dataset
GROUP BY product_id, product_category
ORDER BY total_units_sold DESC
LIMIT 10;



SELECT *
FROM  world_ecommerce.ecommerce_dataset;

--  Revenue by month
SELECT
    DATE_FORMAT(order_date, '%Y-%m-01') AS month,
    SUM(total_revenue) AS monthly_sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY month
ORDER BY month;


-- Top 5 product that generate the highest revenue
SELECT
    product_category,
    SUM(total_revenue) AS total_category_revenue
FROM world_ecommerce.ecommerce_dataset
GROUP BY product_category
ORDER BY total_category_revenue DESC
LIMIT 6;
```

```sql
-- Profit_margin Month-over-Month
SELECT
   month,
   monthly_sales,
   monthly_sales
     - LAG(monthly_sales) OVER (ORDER BY month) AS profit_margin
FROM (
   SELECT
      DATE_FORMAT(order_date, '%Y-%m-01') AS month,
      SUM(total_revenue) AS monthly_sales
   FROM world_ecommerce.ecommerce_dataset
   GROUP BY month
) t;



-- Revenue By Country
SELECT customer_country,
          SUM(total_revenue) AS country_revenue
FROM  world_ecommerce.ecommerce_dataset
GROUP BY customer_country
ORDER BY country_revenue DESC ;

-- Impact Of Discounts On Sales
SELECT discount_rate,
          COUNT(order_id) AS number_of_orders,
   SUM(total_revenue) AS total_sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY discount_rate
ORDER BY discount_rate ;

-- Payment Method Analysis
SELECT payment_method,
          COUNT(order_id) AS total_orders,
   SUM(total_revenue) AS total_sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY payment_method
ORDER BY total_orders DESC ;

-- Repeat Customer
SELECT customer_id,
          COUNT(order_id) AS number_of_orders
```

```
FROM world_ecommerce.ecommerce_dataset
GROUP BY customer_id
ORDER BY customer_id < 1 ;

-- Customer that spend the most

SELECT customer_id,
        SUM(total_revenue) AS lifetime_value
FROM world_ecommerce.ecommerce_dataset
GROUP BY customer_id
ORDER BY lifetime_value DESC
LIMIT 10 ;


-- Top 20 Customer with the highest spend
WITH customer_sales AS
(
SELECT customer_id,
        SUM(total_revenue) AS total_spent
FROM world_ecommerce.ecommerce_dataset
GROUP BY customer_id
)
SELECT
        customer_id,
   total_spent,
   RANK() OVER ( ORDER BY total_spent DESC) AS customer_rank
   FROM customer_sales
   LIMIT 20;

-- Looking for month that the company did not make profit
SELECT
   curr.month,
   curr.monthly_sales,
   prev.monthly_sales AS prev_month_sales,
   curr.monthly_sales - prev.monthly_sales AS revenue_change
FROM
(
   SELECT
      DATE_FORMAT(order_date, '%Y-%m-01') AS month,
      SUM(total_revenue) AS monthly_sales
   FROM world_ecommerce.ecommerce_dataset
   GROUP BY month
) curr
LEFT JOIN
```

```
(
   SELECT
      DATE_FORMAT(order_date, '%Y-%m-01') AS month,
      SUM(total_revenue) AS monthly_sales
   FROM world_ecommerce.ecommerce_dataset
   GROUP BY month
) prev
ON curr.month = DATE_ADD(prev.month, INTERVAL 1 MONTH)
ORDER BY curr.month;

-- Did number of orders drop

SELECT
   DATE_FORMAT(order_date, '%Y-%m-01') AS month,
   COUNT(DISTINCT order_id) AS total_orders,
   SUM(total_revenue) AS monthly_sales
FROM world_ecommerce.ecommerce_dataset
GROUP BY month
ORDER BY month;

-- customers spend less per order
SELECT
   DATE_FORMAT(order_date, '%Y-%m-01') AS month,
   SUM(total_revenue) / COUNT(DISTINCT order_id) AS avg_order_value
FROM world_ecommerce.ecommerce_dataset
GROUP BY month
ORDER BY month;

-- did top product stop selling
SELECT
   DATE_FORMAT(order_date, '%Y-%m-01') AS month,
   product_category,
   SUM(total_revenue) AS category_revenue
FROM world_ecommerce.ecommerce_dataset
GROUP BY
   DATE_FORMAT(order_date, '%Y-%m-01'),
   product_category
ORDER BY
   month,
   category_revenue DESC;
```