1. Find Top 10 Highest Revenue Generating Products

```
select  product_id,
   round(sum(sale_price * quantity), 2) as total_revenue
from
   retail_data
group by
   product_id
order by
   total_revenue desc
limit 10;
```

product_id	total_revenue
TEC-CO-10004722	245056
OFF-BI-10000545	163777.7
TEC-MA-10002412	130406.4
FUR-CH-10002024	120090.7
TEC-PH-10001459	113041.9
TEC-CO-10001449	107388
OFF-BI-10003527	97082.9
TEC-MA-10000822	89622.3
FUR-BO-10002213	84014.8
TEC-MA-10001047	81549

2. Find Top 5 Highest Selling Products In Each Region

```
select region, product_id, total_quantity
from (
   select region, product_id,sum(quantity) as total_quantity,
      ROW_NUMBER() OVER (partition by  region order by sum(quantity) desc) as rn
   from retail_data
   group by region, product_id
) as ranked
where rn <= 5
order by  region, total_quantity desc;
```

region	product_id	total_quantity
Central	OFF-BI-10000301	34
Central	OFF-BI-10000756	33
Central	OFF-BI-10001249	29
Central	OFF-BI-10000546	29
Central	FUR-CH-10002304	27
East	OFF-PA-10001970	33
East	OFF-BI-10003656	32
East	FUR-FU-10004848	31
East	OFF-FA-10000621	31
East	OFF-FA-10002780	29
South	OFF-ST-10003716	26
South	FUR-CH-10000513	24
South	OFF-BI-10004728	24
South	OFF-BI-10000014	23
South	FUR-FU-10001731	21
West	TEC-AC-10003832	45
West	OFF-BI-10000174	32
West	OFF-BI-10001036	31
West	OFF-ST-10002486	29
West	OFF-BI-10001670	29

3. Find Month Over Month Growth Comparison For 2022 and 2023 Sales eg: Jan 2022 vs Jan 2023

```
select
    curr.sales_month,
    prev.total_sales as sales_2022,
    curr.total_sales as sales_2023,
    round(((curr.total_sales - prev.total_sales) / prev.total_sales) * 100, 2) as growth_percent
from (
    select
        month(order_date) as sales_month,
        round(sum(sale_price * quantity), 2) as total_sales
    from retail_data
    where year(order_date) = 2023
    group by sales_month
) as curr
join (
    select
        month(order_date) as sales_month,
```

```
    round(sum(sale_price * quantity), 2) as total_sales
  from retail_data
  where year(order_date) = 2022
  group by sales_month
) as prev
on curr.sales_month = prev.sales_month
order by curr.sales_month;
```

sales_month	sales_2022	sales_2023	growth_percent
1	437431.3	434765.5	-0.61
2	444011.1	731638.8	64.78
3	394105.2	393051.9	-0.27
4	476400.9	543231.5	14.03
5	413625.5	410707.9	-0.71
6	465300.3	328939	-29.31
7	375278.4	422533.7	12.59
8	534562.4	465010.3	-13.01
9	433887	420620.5	-3.06
10	601707.8	626498.3	4.12
11	451809.6	334940.6	-25.87
12	447421.8	491848.9	9.93

4. In Which Year And Which Month Did Each Category Had The Highest Sales?

```
select category ,sales_month,sales_year ,total_sales from
(select category ,
month(order_date) as sales_month ,
year(order_date) as sales_year,
round(sum(sale_price * quantity), 2) as total_sales,
row_number() over( partition by category order by sum(sale_price * quantity) desc) as rn
from retail_data
group by sales_month,category,sales_year ) as ranked
where rn = 1
order by category;
```

category	sales_month	sales_year	total_sales
Furniture	8	2023	230523.5
Office Supplies	2	2023	287244.6
Technology	10	2023	295586.5

5. Which SubCategory Had Highest Profit Growth In 2023 Compared To 2022 ?

```
with profit_2022 as (
    select sub_category, sum(profit) as profit_2022
    from  retail_data
    where year(order_date) = 2022
    group by sub_category
),
profit_2023 as (
    select sub_category, sum(profit) as profit_2023
    from  retail_data
    where year(order_date) = 2023
    group by sub_category
),

joined_profits as (
    select
        coalesce(p22.sub_category, p23.sub_category) as sub_category,
        coalesce(p22.profit_2022, 0) as profit_2022,
        coalesce(p23.profit_2023, 0) as profit_2023
    from profit_2022 p22
    left join profit_2023 p23 on p22.sub_category = p23.sub_category

    union

    select
        p23.sub_category,
        0 as profit_2022,
        p23.profit_2023
    from profit_2023 p23
    left join profit_2022 p22 on p23.sub_category = p22.sub_category
    where p22.sub_category is null
)
```

```
select
    sub_category,
    profit_2022,
    profit_2023,
    round(
        ((profit_2023 - profit_2022) / nullif(profit_2022, 0)) * 100,
        2
    ) as profit_growth_percent
from joined_profits
order by profit_growth_percent desc
limit 1;
```

sub_category	profit_2022	profit_2023	profit_growth_perc...
Supplies	5161.0000286102295	9241.50012025237	79.06