Creating a Table df_orders so the columns does take more space as required

```
create table df_orders
(
[order_id] int primary key,
[order_date] date,
[ship_mode] varchar(20),
[segment]  varchar(20),
[country]  varchar(20),
[city] varchar(20),
[state] varchar(20),
[postal_code] varchar(20),
[region]  varchar(20),
[category] varchar(20),
[sub_category] varchar(20),
[product_id] varchar(50),
[quantity] int,
[discount] decimal(7,2),
[sales_price] decimal(7,2),
[profit] decimal(7,2)
)
```

Questions to Solve using SQL

Find the 10 highest revenue generating products

```
Select Top 10 product_id,
SUM(sales_price) as Sales
from df_orders
Group by product_id
order by Sales desc
```

Find Top 5 highest selling products in each region

```
With cte as (
Select region, product_id, SUM(sales_price) as Sales
from df_orders
group by region, product_id
)
Select * from
(
select *,
ROW_NUMBER() over(partition by region order by sales desc) as row_no
from
cte) a
where row_no<=5
```

Find month on month growth comparison for 2022 and 2023 sales

```
with cte as(
Select year(order_date) as order_year,
DATENAME (mm, CONCAT('1900', FORMAT(CAST(MONTH(order_date) AS INT), '00'), '01')) as
order_month,
sum(sales_price) as Sales
from df_orders
group by
year(order_date) ,
DATENAME (mm, CONCAT('1900', FORMAT(CAST(MONTH(order_date) AS INT), '00'), '01'))
)
Select order_month,
SUM(CASE when order_year=2022 then Sales else 0 end) as Order_2022,
SUM(CASE when order_year=2023 then Sales else 0 end) as Order_2023
from cte
group by order_month
```

For each category which month has the highest sales

```
with cte as(
Select
category,
Format(order_date,'yyyyMM') as Order_month_year,
sum(sales_price) as Sales
from df_orders
group by category,
Format(order_date,'yyyyMM')
)
Select * from
(
Select *,
ROW_NUMBER() over(Partition by category order by sales desc) as rn
from cte) a
where rn=1
```

Which subcategory had the highest growth by profit in 2023 compare to 2022

```
with cte as(
Select sub_category,
year(order_date) as order_year, sum(sales_price) as Sales
from df_orders
group by
sub_category,
year(order_date)
),
cte2 as (
Select sub_category,
```

```
SUM(CASE when order_year=2022 then Sales else 0 end) as Sales_2022,
SUM(CASE when order_year=2023 then Sales else 0 end) as Sales_2023
from cte
group by sub_category
)
select  Top 1 *,
(Sales_2023-Sales_2022)*100/Sales_2022 as Profit_Growth
from
cte2
order by Profit_Growth desc
```