Task1

What are the top-selling products in each category?

```sql
CREATE OR REPLACE VIEW top_selling_products AS
WITH cte AS (
    SELECT
        category_name,
        product_name,
        ROUND(SUM(total_amount), 2) AS total_sales
    FROM fact_transactions ft
    INNER JOIN dim_category c ON c.category_id = ft.category_id
    INNER JOIN dim_product p ON p.product_id = ft.product_id
    GROUP BY category_name, product_name
),
cte2 AS (
    SELECT
        category_name,
        product_name,
        total_sales,
        RANK() OVER (PARTITION BY category_name ORDER BY total_sales DESC)
AS rank_sales
    FROM cte
)
SELECT
    category_name,
    product_name,
    total_sales
FROM cte2
WHERE rank_sales = 1
ORDER BY total_sales DESC;
```

OR

```sql
with cte as (select category_name , product_name ,sum(TOTAL_AMOUNT) as
Total_Sales ,rank() over(partition by category_name order by
sum(TOTAL_AMOUNT)  desc) as rank_sales
from fact_transactions ft inner join categorydim c
on C.CATEGORY_ID=FT.CATEGORY_ID inner join product_dim p
on P.PRODUCT_ID=FT.PRODUCT_ID
group by category_name , product_name)
select category_name , product_name,Total_Sales
from cte
```

```
where rank_sales=1
order by Total_Sales desc
```

How do purchasing patterns change based on time or customer demographics?

- based on time

```
CREATE view sales_time as (with cte as (select case when time_hour >= 6
and  time_hour <12 then 'Morning'
          when  time_hour >= 12 and  time_hour <18 then 'Afternoon'
          when  time_hour >= 18 and  time_hour <24 then 'Early night'
          when  time_hour >= 0 and  time_hour <6 then 'AfterMid'
end as Time_Period,TOTAL_AMOUNT
from fact_transactions )
select Time_Period,sum(TOTAL_AMOUNT) as Total_Sales
 from cte
group by Time_Period
order by Total_Sales desc)
```

- based on age

```
CREATE view sales_age as (with cte as (select case when age >= 5 and  age
<=12 then 'Children'
                  when  age >= 13 and  age <=19 then 'Teenagers'
                  when  age >= 20 and  age <=59 then 'Adults'
                  when  age >= 60 then 'Seniors' end as Age_Groups
                   ,TOTAL_AMOUNT
from fact_transactions ft inner join dim_customer c on C.CUSTOMER_ID=
FT.CUSTOMER_ID )
select Age_Groups,sum(TOTAL_AMOUNT) as Total_Sales
 from cte
group by Age_Groups
order by Total_Sales desc)


CREATE view sales_age1 as (with cte as (select case when age >= 10
and  age <=19 then '10s'
                  when age >= 20 and  age <=29 then '20s'
                  when age >= 30 and  age <=39 then '30s'
```

```
                    when age >= 40 and  age <=49 then '40s'
                    when age >= 50 and  age <=59 then '50s'
                    when age >= 60 and  age <=69 then '60s'
                    when age >= 70 and  age <=79 then '70s'
                    when age >= 80  then 'Above 80s'
                     end as Age_Groups
                     ,TOTAL_AMOUNT
from fact_transactions ft inner join dim_customer c on C.CUSTOMER_ID=
FT.CUSTOMER_ID )
select Age_Groups,sum(TOTAL_AMOUNT) as Total_Sales
 from cte
group by Age_Groups
order by Total_Sales desc)
```

- based on Gender

```
CREATE view sales_gender as (select gender ,sum(TOTAL_AMOUNT) as
Total_Sales
from fact_transactions ft inner join dim_customer c
on C.CUSTOMER_ID= FT.CUSTOMER_ID
group by gender
order by Total_Sales desc)
```

- based on Loyalty Status

```
CREATE view sales_loyality as (select LOYALTY_STATUS ,sum(TOTAL_AMOUNT) as
Total_Sales
from fact_transactions ft inner join dim_customer c on C.CUSTOMER_ID=
FT.CUSTOMER_ID
group by LOYALTY_STATUS
order by Total_Sales desc)
```

- based on City

```
CREATE view sales_city as (select city ,sum(TOTAL_AMOUNT) as Total_Sales
from fact_transactions ft inner join dim_customer c on C.CUSTOMER_ID=
FT.CUSTOMER_ID
group by city
order by Total_Sales desc)
```

Which types of promotions result in the highest sales?

```
CREATE view sales_promotion as (select  PROMOTION_NAME,sum(TOTAL_AMOUNT)
as Total_Sales
from fact_transactions ft inner join dim_promotion p
on P.PROMOTION_ID =FT.PROMOTION_ID
group by PROMOTION_NAME
order by Total_Sales desc
)
```

Task2

Write a query to find products that customers frequently purchase together

```
CREATE view product_combination as (with product_combinations as (
select transaction_id,product_name,rank()over(partition by
transaction_id,category_id order by product_name ) as rank
FROM fact_transactions ft inner join dim_product p on
P.PRODUCT_ID=FT.PRODUCT_ID),
 Combination_Count as (select a.product_name as P1 ,b.product_name as
P2,count(*) as Combination_Count
from product_combinations a inner join product_combinations b
on a.transaction_id =b.transaction_id
where a.rank<b.rank
group by a.product_name,b.product_name)
select p1,p2,Combination_Count
from Combination_Count
order by Combination_Count desc

)
```