Task 1

Colour Preference Analysis for Women Shoes - task is to discover which colors women are most interested in when selecting shoes. The goal is to identify the colors they rate highest so that we can apply a color filter whenever a customer searches for women's shoes.

```
Select color, avg(average_rating) as rating from products
where sub_category="Women/shoes"
group by color
order by rating desc;
```

Task2

Colour selection for new Product Line - task is to provide the colors in which we are selling the most items, i.e., the list of highest-selling colors. Provide me a list of **five** colors so that we can forward it to the supply chain department for procurement in those colors only. The company is planning to introduce several new items and thee colors would be preferred while procuring new items.

```
select p.color, sum(s.item_sold) as items_sold from products as p
inner join
(select max(product_id)as product_id, sum(quantity) as item_sold from sales group by product_id) as s
on s.product_id = p.product_id
group by p.color
ORDER BY `items_sold` DESC Limit 5;
```

Task3

Mens Product Analysis for improving Sales - task is to provide a list of 50 men's products that have high ratings and at least 500 reviews. The company is currently experiencing low sales of men's products. Our research shows that good ratings and a high number of reviews are the most important factors in the buying decision for men.

```
Select product_id, MAX(product_name) as product_name,AVG(average_rating) AS rating,
SUM(reviews_count) AS reviews_count from products
where sub_category like 'men%'
group by product_id
having SUM(reviews_count) > 500
order by rating limit 50;
```

Task4

Identifying Top Customers for Loyalty Points - task is to provide a list of 100 customers who have bought the maximum number of products over time along with the number of products they have bought.

```
SELECT c.customer_id, c.customer_name,SUM(s.quantity) as item_bought
from customer as c
inner join sales as s
on c.customer_id = s.customer_id
group by c.customer_id, c.customer_name
order by item_bought DESC
limit 100;
```

Task5

Brand Research for Men's Sports Shoes Category - task is to provide the names of two brands that are most preferred by men (in terms of quantity sold) when it comes to shoes.

```
SELECT p.brand_name, SUM(s.quantity) as total_items_sold
FROM products as p
INNER JOIN sales as s ON p.product_id = s.product_id
WHERE p.sub_category = 'men/shoes'
GROUP BY p.brand_name
ORDER BY total_items_sold DESC
LIMIT 2;
```

Task 6

Research and Evaluation of Profitable Sub-Categories for new product introduction - task is to prepare a list of 10 sub_categories with the highest percentage of profit margin that can be introduced into our stores and help us increase our profit

```
SELECT products.sub_category, (SUM(sales.profit) / SUM(sales.sales)) * 100 as profit_margin
FROM products
INNER JOIN sales
ON products.product_id = sales.product_id
GROUP BY products.sub_category
ORDER BY profit_margin DESC
Limit 10;
```

Task 7

Colour Stocking strategy for optimal sales and Positive review - Task is to find the list of colors and the corresponding sum of sales for only those products which have at least 100 reviews and an average rating of 4+

```
SELECT p.color, SUM(s.sales) as sum_sales
FROM products p
INNER JOIN sales s
ON p.product_id = s.product_id
WHERE p.reviews_count >= 100
GROUP BY p.color
HAVING AVG(p.average_rating) > 4
ORDER BY sum_sales DESC;
```

Task 8

Analysis of delivery performance to improve sales - task is to find out the sum of sales, the sum of profit, and the count of orders for each delivery mode in the past year.

```
SELECT delivery_mode, SUM(sales) as total_sales, SUM(profit) as total_profit,
COUNT(DISTINCT order_id) as order_count
FROM sales
WHERE order_date BETWEEN '2022-01-30' AND DATE_ADD('2022-01-30', INTERVAL 1
YEAR)
GROUP BY delivery_mode
ORDER BY total_sales DESC
LIMIT 10;
```

Task 9

Identifying brand with high average profit per order - task is to provide the names of brands along with the average average profit per order during the past year sorted by average profit per order from maximum to minimum.

```
SELECT brand_name, (total_profit / total_orders) as avg_profit_per_order
FROM (
SELECT p.brand_name, SUM(s.profit) as total_profit, COUNT(DISTINCT s.order_id) as total_orders
FROM sales s
JOIN products p
ON s.product_id = p.product_id
WHERE order_date BETWEEN  DATE_SUB('2022-12-30', INTERVAL 1 YEAR) AND '2022-12-30'
GROUP BY p.brand_name
) as c
ORDER BY avg_profit_per_order DESC;
```

Task 10

Allocation of Men's shoes inventory for Three States - task is to give me the percentage of shoes sold last year in these three states so that I can provide that percentage from our production to each of these three states.

```
Select  p.sub_category,c.state,  (SUM(s.quantity)/1365) * 100 as percentage_of_products_sold
from sales as s
Join products as p on p.product_id = s.product_id
Join customer as c on c.customer_id = s.customer_id
Where p.sub_category = 'Men/Shoes' AND order_date BETWEEN  DATE_SUB('2022-12-30', INTERVAL 1 YEAR) AND '2022-12-30' AND c.state in ('California', 'New York', 'Texas')
GROUP BY c.state, p.sub_category
order by percentage_of_products_sold DESC;
```

Task 11

Personalized email marketing to drive sales and build Customer relationship - task is to provide a list of our five best-selling (in terms of quantity in the state of California) shoes that are white in color

```
Select  max(p.brand_name) as brand_name, p.product_name, p.color, Max(p.sub_category) as sub_category ,MAX(c.state) as state, SUM(s.quantity) as total_sales

from sales as s

Join products as p on p.product_id = s.product_id

Join customer as c on c.customer_id = s.customer_id

where p.color = 'white' AND c.state = 'California' AND p.sub_category = 'men/shoes'

GROUP BY p.product_name

ORDER BY total_sales desc limit 5;
```

Task 12

Optimizing Stock for new store opening in New York based on subcategory profitability - task is to provide a list of four subcategories from each category, ranked based on the total profit they are generating in New York.

```
Select category, sub_category, total_quantity

FROM (SELECT *, Rank() Over(PARTITION BY category ORDER BY total_profit DESC) as rank_

        FROM (Select  max(p.category) as category, p.sub_category, SUM(s.quantity) as total_quantity, SUM(s.profit) as total_profit, c.state

                from sales as s

                Join products as p on p.product_id = s.product_id

                Join customer as c on c.customer_id = s.customer_id

                Where c.state = 'New York'

                GROUP BY p.sub_category

                ORDER BY total_profit desc) as subcat_profit

        )as subcat_ranked

Where rank_ <= 4
```

Task 13

Brand performance Analysis for Summer 2019 - task is to find out which brand had the highest profit during the summer season (the months of June, July and August) of 2019.

```
Select p.brand_name, SUM(s.profit) as total_profit
from products as p
JOIN sales as s
on s.product_id = p.product_id
WHERE Month(order_date) BETWEEN  6 AND 8 And Year(order_date) = 2019
GROUP BY p.brand_name
ORDER by total_profit DESC limit 1
```

Task 14

Identification of Top Brands for most profitable category with standard delivery - task is to determine the top 2 brands with the highest total sales value for the products belonging to the most profitable category during the specified time period of January 1, 2019, to June 30, 2019. It's essential that the delivery mode used is "Standard Delivery".

```
Select p.brand_name, SUM(s.sales) as total_sales
from sales as s
JOIN products as p
on p.product_id = s.product_id
Where order_date BETWEEN '2019-01-01' AND '2019-06-30' AND delivery_mode = 'Standard Delivery'
AND category = (Select Category
                From (Select category, SUM(profit) as total_profit
                      from sales as s
                      JOIN products as p
                      on p.product_id = s.product_id
                      Where order_date BETWEEN '2019-01-01' AND '2019-06-30' AND delivery_mode = 'Standard Delivery'
                      GROUP BY category
                      ORDER BY total_profit desc limit 1) as z)
Group by p.brand_name
ORDER by total_sales desc;
```