1. Find the number of coupons redeemed per campaign.
   * *Solution:* Use GROUP BY with COUNT and GROUP\_CONCAT on campaign\_id and redemption\_status.
2. List customers who have never redeemed a coupon.
   * *Solution:* Use LEFT JOIN between train.csv and customer\_id, then filter out redemption\_status = 1.
3. Get the average number of family members per customer.
   * *Solution:* Use AVG on family\_size from customer\_demographics.csv.
4. Retrieve the total sales value of items bought using a coupon.
   * *Solution:* Join customer\_transaction\_data.csv with train.csv on customer\_id and coupon\_id and sum the selling\_price.
5. Identify the top 3 most popular products by quantity sold under a campaign.
   * *Solution:* Use JOIN between customer\_transaction\_data.csv and coupon\_item\_mapping.csv, then GROUP BY item\_id with SUM(quantity) and ORDER BY.
6. Find the total discount amount given to customers for each coupon.
   * *Solution:* Use SUM(coupon\_discount) grouped by coupon\_id.
7. Retrieve customers who have redeemed more than 5 coupons.
   * *Solution:* Use GROUP BY on customer\_id with HAVING COUNT(coupon\_id) > 5.
8. List all the customers who have purchased a particular item.
   * *Solution:* Use JOIN between customer\_transaction\_data.csv and item\_data.csv on item\_id, filtering by item category or item\_id.
9. Find the average income bracket of customers who redeemed coupons.
   * *Solution:* Join customer\_demographics.csv with train.csv, then calculate the AVG(income\_bracket) for customers where redemption\_status = 1.
10. Determine the most frequently used coupon in a specific campaign.
    * *Solution:* Use JOIN between train.csv and campaign\_data.csv, then GROUP BY coupon\_id and COUNT for a given campaign\_id.

Complex SQL Questions Using Window Functions (20)

1. Rank customers based on the number of coupons redeemed per campaign.
   * *Solution:* Use ROW\_NUMBER() or RANK() with PARTITION BY campaign\_id ORDER BY COUNT(coupon\_id).
2. Calculate the cumulative discount for each customer across all campaigns.
   * *Solution:* Use SUM(coupon\_discount) OVER (PARTITION BY customer\_id ORDER BY campaign\_id).
3. Identify the top 5 customers by total spending on items with coupons.
   * *Solution:* Use SUM(selling\_price) OVER (PARTITION BY customer\_id) and ORDER BY to limit to top 5.
4. Calculate the moving average of coupon redemption rate for each campaign over time.
   * *Solution:* Use AVG(redemption\_status) OVER (PARTITION BY campaign\_id ORDER BY start\_date ROWS BETWEEN 4 PRECEDING AND CURRENT ROW).
5. Rank products based on the quantity sold per coupon redemption.
   * *Solution:* Use RANK() with PARTITION BY coupon\_id ORDER BY SUM(quantity).
6. Find the first purchase date for each customer who redeemed a coupon.
   * *Solution:* Use MIN(date) OVER (PARTITION BY customer\_id ORDER BY date).
7. Calculate the average sales value per customer across campaigns.
   * *Solution:* Use AVG(selling\_price) OVER (PARTITION BY customer\_id).
8. Identify the percentage change in sales for each product before and after a campaign.
   * *Solution:* Use LAG() to get previous sales and calculate the difference and percentage change.
9. Find the rank of each coupon based on total discount value in a campaign.
   * *Solution:* Use RANK() with PARTITION BY campaign\_id ORDER BY SUM(coupon\_discount).
10. Determine how many campaigns a customer participated in, with a limit to those who redeemed coupons.
    * *Solution:* Use COUNT(DISTINCT campaign\_id) OVER (PARTITION BY customer\_id) with WHERE redemption\_status = 1.
11. List the customers with the highest cumulative discount, ordered by campaign.
    * *Solution:* Use SUM(coupon\_discount) OVER (PARTITION BY customer\_id ORDER BY campaign\_id).
12. Find the median sales value of items purchased by customers who redeemed coupons.
    * *Solution:* Use PERCENTILE\_CONT(0.5) WITHIN GROUP (ORDER BY selling\_price).
13. Identify customers who have redeemed coupons in consecutive campaigns.
    * *Solution:* Use LAG(campaign\_id) to check if the previous campaign is consecutive.
14. Get the moving sum of coupon redemptions for each customer by campaign.
    * *Solution:* Use SUM(redemption\_status) OVER (PARTITION BY customer\_id ORDER BY campaign\_id ROWS BETWEEN 2 PRECEDING AND CURRENT ROW).
15. Calculate the total discount and redemption rate per customer and campaign.
    * *Solution:* Use SUM(coupon\_discount) OVER (PARTITION BY customer\_id, campaign\_id) and calculate redemption rate using COUNT.
16. Find the customers who spent more than the average in a given campaign.
    * *Solution:* Use AVG(selling\_price) OVER (PARTITION BY campaign\_id) and filter customers spending more.
17. Rank customers by total quantity purchased, considering only redeemed coupons.
    * *Solution:* Use RANK() with PARTITION BY customer\_id ORDER BY SUM(quantity) where redemption\_status = 1.
18. Identify products with the highest total coupon discount value, ordered by campaign.
    * *Solution:* Use SUM(coupon\_discount) OVER (PARTITION BY item\_id ORDER BY campaign\_id).
19. Determine customers who have redeemed at least 3 coupons in the last 5 campaigns.
    * *Solution:* Use COUNT(coupon\_id) OVER (PARTITION BY customer\_id ORDER BY campaign\_id ROWS BETWEEN 4 PRECEDING AND CURRENT ROW).
20. Find the top 5 most redeemed coupons for each campaign.
    * *Solution:* Use RANK() with PARTITION BY campaign\_id ORDER BY COUNT(coupon\_id) and limit to top 5.