PLACEMENTDOST

SQL Assessment questions

Note:

Please be advised that these assessment inquiries are designed to accommodate interns with diverse skill levels, ranging from novices to seasoned analysts. Should you encounter any challenging questions, you are encouraged to seek solutions independently or reach out to us for assistance at intern@placementdost.com. Best wishes for success in completing the assessment!

Tools needed :- MS SQL

1. Convert Dataset to SQL Database:

- Create SQL statements to define and populate tables for menu_details and order_details using the provided dataset.

2. Basic SELECT Queries:

- Retrieve all columns from the menu_items table.

- Display the first 5 rows from the order_details table.

3. Filtering and Sorting:

- Select the item_name and price columns for items in the 'Main Course' category.

- Sort the result by price in descending order.

4. Aggregate Functions:

- Calculate the average price of menu items.

- Find the total number of orders placed.

5. Joins:

- Retrieve the item_name, order_date, and order_time for all items in the order_details table, including their respective menu item details.

6. Subqueries:

- List the menu items (item_name) with a price greater than the average price of all menu items.

7. Date and Time Functions:

- Extract the month from the order_date and count the number of orders placed in each month.

8. Group By and Having:

- Show the categories with the average price greater than $15.

- Include the count of items in each category.

9. Conditional Statements:

- Display the item_name and price, and indicate if the item is priced above $20 with a new column named 'Expensive'.

10. Data Modification - Update:

- Update the price of the menu item with item_id = 101 to $25.

11. Data Modification - Insert:

- Insert a new record into the menu_items table for a dessert item.

12. Data Modification - Delete:

- Delete all records from the order_details table where the order_id is less than 100.

13. Window Functions - Rank:

- Rank menu items based on their prices, displaying the item_name and its rank.

14. Window Functions - Lag and Lead:

- Display the item_name and the price difference from the previous and next menu item.

15. Common Table Expressions (CTE):

- Create a CTE that lists menu items with prices above $15.

- Use the CTE to retrieve the count of such items.

16. Advanced Joins:

- Retrieve the order_id, item_name, and price for all orders with their respective menu item details.

- Include rows even if there is no matching menu item.

17. Unpivot Data:

- Unpivot the menu_items table to show a list of menu item properties (item_id, item_name, category, price).

18. Dynamic SQL:

- Write a dynamic SQL query that allows users to filter menu items based on category and price range.

19. Stored Procedure:

- Create a stored procedure that takes a menu category as input and returns the average price for that category.

20. Triggers:

- Design a trigger that updates a log table whenever a new order is inserted into the order_details table.

Advanced SQL (optional)

21. Recursive Common Table Expressions (CTE):

- Implement a recursive CTE to display the hierarchy of menu items with their subcategories.

22. Temporal Tables:

- Design a temporal table structure to track changes in menu item prices over time.

23. Database Transactions:

- Write a series of SQL statements within a transaction that ensures atomicity for updating menu item prices and inserting a new order.

24. Database Security:

- Create a role in the database and assign permissions to the role to restrict access to sensitive tables.

25. Advanced Indexing:

- Analyze the menu_items table and suggest an appropriate index to improve the performance of queries involving category-based filtering.