ONLINE RETAIL STORE

Group 102: Saketh Ragirolla 2021092 & Ritisha Singh 2021089

EmbeddedSQL Query

To write the 2 required embedded SQL queries, we have used python as our programming language.

EMBEDDED SQL SYNTAX USED

We have designed the foundation of every query on MySQL using the command:

SELECT col1, col2, col3, ….coln or *
FROM table_name
[WHERE condition**]**
[GROUP BYcolumn_list**]**
[ORDER BY column_list**];**

EMBEDDED SQL QUERIES

1. This SQL query is used to generate a report that shows the average product rating for each category, as well as the number of products in each category that have a rating above the category average.

   Tables Used: Category, Prod_feedback

   Columns Generated: Category_name, Avg_rating, Num_products_with_rating_above_avg

2. This SQL query retrieves information about customer orders that were paid for using a card payment method, where the customer's email address contains the string "pro".

   Tables Used: Cart, Customer, Payment_portal

   Columns Generated: Cart_ID**,** Cust_name, Cust_email, Total_Cost, Payment_type, Payment_date, Customer_ID

OLAP Query

In all, we have implemented 4 OLAP SQL queries.

OLAP_SQL SYNTAX USED

We have designed the foundation of every OLAP query on MySQL using the command:

SELECT col1, col2, col3, ….coln or *
FROM table_name
[WHERE condition**]**
[GROUP BYcolumn_list **WITH ROLLUP]**
[ORDER BY column_list**];**

OLAP_SQL QUERIES

1. This query retrieves the total number of products in each category and also the grand total of all products across all categories.

   Tables Used: Category, Product

   Columns Generated: Category_name, Total_products

2. This SQL query is used to retrieve the top 10 best-selling products in terms of total sales.

   Tables Used: Customer, Cart, Offer

   Columns Generated: Prod_name, Total_sales

3. This SQL query is used to generate a report of the top 10 products with the highest average ratings.

   Tables Used: Product, Prod_feedback

   Columns Generated: Prod_name, Avg_rating

4. This SQL query is used to generate a report that shows the total amount spent by each customer on their orders, along with the customer's name and ID.

   Tables Used: Customer, Order_

   Columns Generated: Cust_name, Customer_ID, total_spent

TRIGGERS

In all, we have implemented 6 Triggers.

TRIGGERS SQL SYNTAX USED

We have designed the foundation of every trigger on MySQL using the command:

DROP IF EXISTS *trigger_name;*

DELIMITER $$

CREATE TRIGGER *trigger_name trigger_time trigger_event*

ON *table_name* **FOR EACH ROW**

BEGIN

--variable declarations

--trigger code

END$$

TRIGGERS

1. The trigger updates the account data of a customer after a new order is placed.

   Tables Used: Account_data, Order_

2. The trigger to delete the older review before inserting a new one for a specific delivery partner and customer.

   Tables Used: Partner_review

3. The trigger to delete the older review before inserting a new one for a specific delivery product and customer.

   Tables Used: Product_review

4. The trigger prevents inserting deals if the minimum requirement is less than 300.

   Tables Used: Offer

5. The trigger updates the number of coupons available to a customer based on the change in their membership plan.

   Tables Used: Account_Data, Membership

6. The trigger prevents adding a Deal_Id to a Cart if the minimum requirement for that deal is greater than the total cost of the car

Tables Used: Cart, Offer

RELATIONAL SCHEMA