

OLAP QUERIES

Query1

```
-- Total revenue generated by each product category

SELECT sql_Inventory.Category.Category_Name, SUM(total_cost) as total_revenue
FROM sql_RetailStore.user_order
INNER JOIN sql_Inventory.Product ON sql_RetailStore.user_order.Product_ID = sql_Inventory.Product.Product_ID
INNER JOIN sql_Inventory.Category ON sql_Inventory.Product.Category_ID = sql_Inventory.Category.Category_ID
INNER JOIN sql_RetailStore.placed_order ON sql_RetailStore.user_order.user_order_id = sql_RetailStore.placed_order.user_order_id
INNER JOIN sql_RetailStore.bill_details ON sql_RetailStore.placed_order.bill_ID = sql_RetailStore.bill_details.bill_id
GROUP BY sql_Inventory.Category.Category_Name;
```

Query 2

```
-- total revenue earned from orders placed by each customer.

SELECT customer_id, SUM(total_cost) as total_revenue
FROM sql_RetailStore.user_order
INNER JOIN sql_RetailStore.placed_order ON sql_RetailStore.user_order.user_order_id = sql_RetailStore.placed_order.user_order_id
INNER JOIN sql_RetailStore.bill_details ON sql_RetailStore.placed_order.bill_ID = sql_RetailStore.bill_details.bill_id
GROUP BY customer_id;
```

Query 3

```
-- Top 10 Selling goods

SELECT Product_Name, SUM(product_quantity) as total_quantity_sold
FROM sql_RetailStore.user_order
INNER JOIN sql_Inventory.Product ON sql_RetailStore.user_order.Product_ID = sql_Inventory.Product.Product_ID
GROUP BY Product_Name
ORDER BY total_quantity_sold DESC
LIMIT 10;
```

Query 4

```
-- total revenue and profit generated from each order
```

```
SELECT uo.user_order_id, SUM(
    uo.product_quantity * p.Price)
    AS Total_Revenue, SUM(
    uo.product_quantity * (p.Price - (p.Price * 0.2)))
    AS Total_Profit
FROM sql_RetailStore.user_order uo
JOIN sql_Inventory.Product p ON uo.Product_ID = p.Product_ID
GROUP BY uo.user_order_id
```

Triggers

Trigger 1

```
DELIMITER $$  
CREATE TRIGGER trigger_name  
BEFORE INSERT ON sql_RetailStore.payment  
FOR EACH ROW  
BEGIN  
    IF NEW.expiry_date < CURDATE() THEN  
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = "Card is expired";  
    END IF;  
END $$  
DELIMITER ;
```

Trigger 2

```
DELIMITER $$  
CREATE TRIGGER trigger_name  
BEFORE UPDATE ON sql_inventory.Product  
FOR EACH ROW  
BEGIN  
    IF NEW.Quantity < 0 THEN  
        SET NEW.Quantity = OLD.Quantity;  
    ELSEIF NEW.Quantity >= 0 THEN  
        SET NEW.Quantity = OLD.Quantity + NEW.Quantity;  
    END IF;  
END $$  
DELIMITER ;
```