Project report:

Vidur Goel

team 80

2021364

Queries:

1. Selection:

SELECT * FROM Online_Retail_Store.orders WHERE order_status='Delivered';

Relational Algebra applied here:

σ(order_status='Delivered')(Online_Retail_Store.orders)

This above mentioned query here uses the following relational algebra operation:

σ (Selection): Selects only the tuples in the Online_Retail_Store.orders relation where the value of the order_status attribute is ' Delivered '.

2. Projection:

SELECT product_name,product_cost,Quantity FROM Online_Retail_Store.product;

Relational Algebra applied here:π(product_name,product_cost,Quantity)(Online_Retail_Store.product)

π (Projection): what this relation does is that it will projects the product_name,product_cost,Quantity attributes of the relation Online_Retail_Store.product

3. Update:

SELECT * FROM Online_Retail_Store.category;

UPDATE Online_Retail_Store.category

SET category_name='B-Series20'

WHERE category_id='1'

Here we are updating the category_name of a category with the category_id =’1’.

4. Product and Selection:

SELECT * FROM Online_Retail_Store.delivery_boy,Online_Retail_Store.tracking_details WHERE
Online_Retail_Store.delivery_boy.delivery_boy_id=Online_Retail_Store.tracking_details.delivery_boy_id;

Here we have used product and selection operations on relations

5. Various Boolean Operations and Selection:

SELECT * FROM Online_Retail_Store.product

WHERE Brand_Name='Gas Relief' OR product_name='Bagelers';Here we have used product and selection operations on relations

6.Using ALTER and various boolean operations:

ALTER TABLE Online_Retail_Store.cart ADD CONSTRAINT check_cart_cost CHECK (total_cost >= 0);

Here we are using alter operation to add the constraint that the total_cost in the cart should be greater than or equal to 0.

7. Using of join and the various Boolean operations as well as select operations:

select
Online_Retail_Store.orders.delivery_boy_id,Online_Retail_Store.orders.order_placed_time,Online_Retail_Store.orders.order_status,Online_Retail_Store.delivery_boy.delivery_boy_name,Online_Retail_Store.delivery_boy.delivery_boy_average_rating

from Online_Retail_Store.orders

left join Online_Retail_Store.delivery_boy

on Online_Retail_Store.orders.delivery_boy_id =
Online_Retail_Store.delivery_boy.delivery_boy_id

where Online_Retail_Store.orders.total_cost>=190.05

8. **Using of temp assignment and the various Boolean operations as well as select operations:**

select del.contact_number,tr.location,del.delivery_boy_name

from Online_Retail_Store.delivery_boy as del, Online_Retail_Store.tracking_details as tr, Online_Retail_Store.orders as o

where o.delivery_boy_id=del.delivery_boy_id and o.track_id=tr.track_id

;

9. Using of reassigning,count,distinct,select as well as Boolean operations in this query

```
SELECT COUNT(DISTINCT pro.Brand_name)

from Online_Retail_Store.product as pro, Online_Retail_Store.tracking_details as tr,
Online_Retail_Store.orders as o

where pro.Brand_name!='Tasigna';
```

10. USING OF SUM IN QUERY WITH REASSIGNMENT

```
SELECT SUM(pro.product_cost*pro.Quantity)

from Online_Retail_Store.product as pro, Online_Retail_Store.tracking_details as tr,
Online_Retail_Store.orders as o;
```

Relational Schema:

customer= (customer_id ,customer_name ,contact_number ,customer_username , customer_password ,email_id ,customer_address)

admin=(admin_id, admin_name, admin_username, admin_password)

delivery_boy= (delivery_boy_id, delivery_boy_name, delivery_boy_username, delivery_boy_password, delivery_boy_average_rating, contact_number, email_id)

category = (category_id, category_name, description)

product = (product_id , product_name, product_cost, Brand_name, Quantity)

dealer = (dealer_id, dealer_name, username, password, address_of_operations, contact_number, email_id)

product_feedback = (feedback_id, rating_given, review, likes, date_published)

coupon= (coupon_id, coupon_expiry_date, minimum_order_value, percentage_discount, maximum_discount, terms_and_conditions, coupon_code)

cart = (cart_id, total_cost, description, product_id, coupon_id)

bill = (bill_id, order_summary, mode_of_payment)

tracking_details = (track_id, delivery_boy_id, contact_number, delivery_status, location)

orders = (order_id, delivery_boy_id, total_cost, delivery_address, order_status, order_placed_time, expected_delivery_time, track_id)