DBMS ASSIGNMENT-4

-RAMYA SRI ACHANTA (2300293)

1)

**Implement the above database Customer Order using DDL query.**

**CUSTOMER TABLE:**

CREATE TABLE customer (

CID INTEGER PRIMARY KEY,

NAME TEXT CHECK(LENGTH(NAME) <= 20),

ADDRESS TEXT CHECK(LENGTH(ADDRESS) <= 50),

CONTACT\_NO INTEGER CHECK(CONTACT\_NO >= 0 AND CONTACT\_NO <= 9999999999),

CREDIT\_LIMIT INTEGER CHECK(CREDIT\_LIMIT >= 0)

);

A screenshot of a computer code

Description automatically generated

select \* from customer;

A screenshot of a computer

Description automatically generated

**PRODUCT TABLE:**

CREATE TABLE product (

PID INTEGER PRIMARY KEY,

NAME TEXT CHECK(LENGTH(NAME) <= 20),

COLOR TEXT CHECK(LENGTH(COLOR)<=20),

MANUFACTURED\_DATE DATE,

MANUFACTURED\_ID INTEGER,

FOREIGN KEY(MANUFACTURED\_ID) REFERENCES manufacturer(MID) ON DELETE CASCADE ON UPDATE CASCADE

);

A computer code with text

Description automatically generated

select \* from product;

A screenshot of a computer

Description automatically generated

**MANUFACTURER TABLE:**

CREATE TABLE manufacturer (

MID INTEGER PRIMARY KEY,

MANUFACTURER\_NAME TEXT CHECK(LENGTH(MANUFACTURER\_NAME) <= 20),

ADDRESS TEXT CHECK(LENGTH(ADDRESS) <= 50),

CONTACT INTEGER CHECK(CONTACT >= 0 AND CONTACT <= 9999999999)

);

A screenshot of a computer program

Description automatically generated

select \* from manufacturer;

A screenshot of a computer

Description automatically generated

**ORDER TABLE:**

CREATE TABLE [order] (

OID INTEGER PRIMARY KEY,

CUSTOMER\_ID INTEGER,

REFERENCE TEXT CHECK(LENGTH(REFERENCE) <= 20),

PURCHASE\_DATE DATE,

FOREIGN KEY(CUSTOMER\_ID) REFERENCES customer(CID) ON DELETE CASCADE ON UPDATE CASCADE

);

A screenshot of a computer

Description automatically generated

select \* from [order];

A screenshot of a computer

Description automatically generated

**ORDER-ITEMS TABLE:**

CREATE TABLE [order-items] (

OID INTEGER,

PRODUCT\_ID INTEGER,

QTY INTEGER,

WARRANTY INTEGER,

UNIT\_PRICE INTEGER,

PRIMARY KEY (OID, PRODUCT\_ID),

FOREIGN KEY(OID) REFERENCES [order](OID) ON DELETE CASCADE ON UPDATE CASCADE,

FOREIGN KEY(PRODUCT\_ID) REFERENCES product(PID) ON DELETE CASCADE ON UPDATE CASCADE

);

A computer screen shot of a computer code

Description automatically generated with medium confidence

select \* from [order-items];

A screenshot of a computer

Description automatically generated

2)

**By following the above schema, give *SQL* for the following problems:**

a)

SELECT CID,NAME from customer

where ADDRESS like'%Bay Area Blvd%' AND

CREDIT\_LIMIT BETWEEN 1000 AND 10000;

A screenshot of a computer

Description automatically generated

b)

SELECT c.NAME

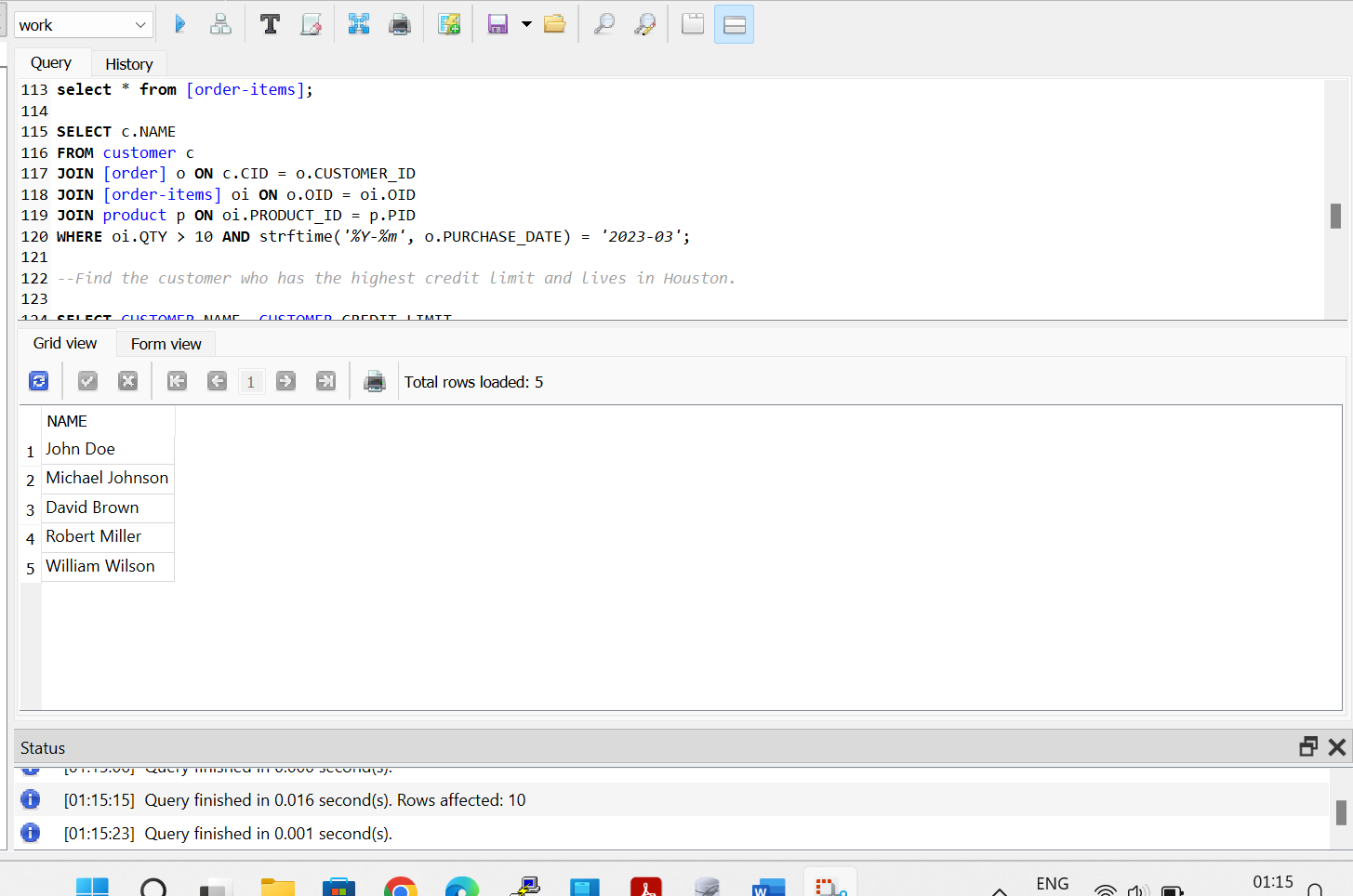
FROM customer c

JOIN [order] o ON c.CID = o.CUSTOMER\_ID

JOIN [order-items] oi ON o.OID = oi.OID

JOIN product p ON oi.PRODUCT\_ID = p.PID

WHERE oi.QTY > 10 AND strftime('%Y-%m', o.PURCHASE\_DATE) = '2023-03';



c)

SELECT CUSTOMER.NAME, CUSTOMER.CREDIT\_LIMIT

FROM CUSTOMER

WHERE CUSTOMER.ADDRESS LIKE'%Houston%'

order by CUSTOMER.CREDIT\_LIMIT DESC

LIMIT 1;

A screenshot of a computer

Description automatically generated

d)

SELECT product.NAME,product.COLOR FROM PRODUCT

JOIN MANUFACTURER ON MANUFACTURER.MID=PRODUCT.MANUFACTURED\_ID

JOIN [ORDER-ITEMS] ON [ORDER-ITEMS].PRODUCT\_ID=PRODUCT.PID

WHERE MANUFACTURER.MANUFACTURER\_NAME='Rainbow' and [ORDER-ITEMS].UNIT\_PRICE between 50 and 500;

A screenshot of a computer

Description automatically generated

e)

SELECT m.MANUFACTURER\_NAME

FROM manufacturer m

JOIN product p ON m.MID = p.MANUFACTURED\_ID

JOIN [order-items] oi ON p.PID = oi.PRODUCT\_ID

GROUP BY m.MANUFACTURER\_NAME

HAVING SUM(oi.QTY) > 20;

A screenshot of a computer

Description automatically generated

f)

SELECT c.NAME

FROM customer c

JOIN [order] o ON c.CID = o.CUSTOMER\_ID

JOIN [order-items] oi ON o.OID = oi.OID

JOIN product p ON oi.PRODUCT\_ID = p.PID

WHERE oi.UNIT\_PRICE = (

SELECT MAX(UNIT\_PRICE)

FROM [order-items]

);

A screenshot of a computer

Description automatically generated

g)

SELECT m.MANUFACTURER\_NAME

FROM manufacturer m

JOIN product p ON m.MID = p.MANUFACTURED\_ID

JOIN [order-items] oi ON p.PID = oi.PRODUCT\_ID

GROUP BY m.MANUFACTURER\_NAME

HAVING AVG(oi.QTY) > (

SELECT AVG(QTY)

FROM [order-items]

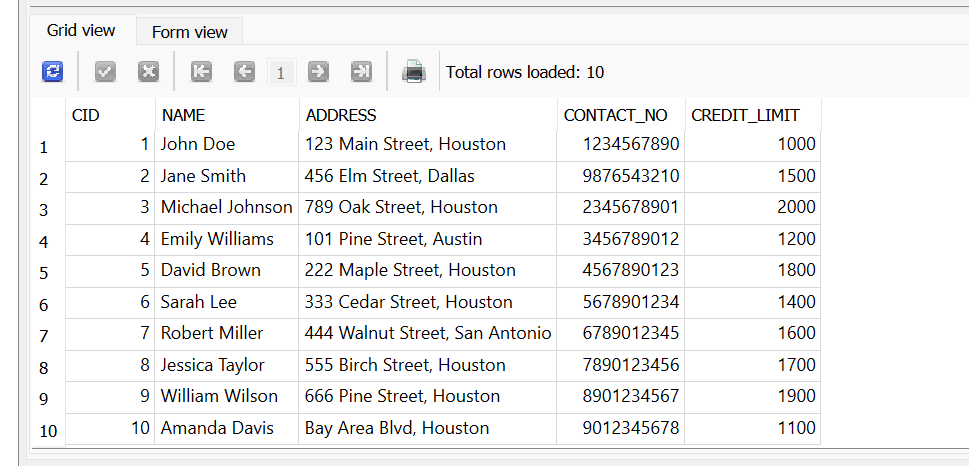
);

A screenshot of a computer

Description automatically generated

h)

customer table before updating:



Query to update:

UPDATE customer

SET CREDIT\_LIMIT = CASE

WHEN EXISTS (SELECT 1 FROM [order] o JOIN [order-items] oi ON o.OID = oi.OID WHERE o.CUSTOMER\_ID = customer.CID GROUP BY o.CUSTOMER\_ID HAVING SUM(oi.QTY) >= 10) THEN ROUND(CREDIT\_LIMIT \* 1.20, 2)

WHEN EXISTS (SELECT 1 FROM [order] o JOIN [order-items] oi ON o.OID = oi.OID WHERE o.CUSTOMER\_ID = customer.CID GROUP BY o.CUSTOMER\_ID HAVING SUM(oi.QTY) > 5 AND SUM(oi.QTY) < 10) THEN ROUND(CREDIT\_LIMIT \* 1.15, 2)

ELSE ROUND(CREDIT\_LIMIT \* 1.10, 2)

END

WHERE CID IN (SELECT DISTINCT CUSTOMER\_ID FROM [order]);

Customer table after updating

A screenshot of a computer

Description automatically generated

i)

Manufacturer before updating:

A screenshot of a computer

Description automatically generated

Product table before updating :

A screenshot of a computer

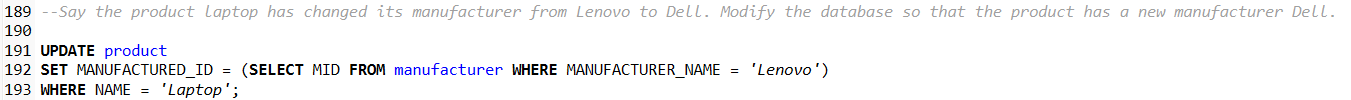
Description automatically generated

Query to update:

UPDATE product

SET MANUFACTURED\_ID = (SELECT MID FROM manufacturer WHERE MANUFACTURER\_NAME = 'Lenovo')

WHERE NAME = 'Laptop';



Manufacturer table after updating:

A screenshot of a computer

Description automatically generated

Product table after updating:A screenshot of a computer

Description automatically generatedj)

Before updating customer table:

A screenshot of a computer

Description automatically generated

DELETE FROM customer

WHERE ADDRESS LIKE '%Houston%';

A screenshot of a computer

Description automatically generated

Order table rows were also updated as customer\_id has “on delete cascade”

A screenshot of a computer

Description automatically generated

Order-items table rows were also updated as customer\_id has “on delete cascade”

A screenshot of a computer

Description automatically generated

3)

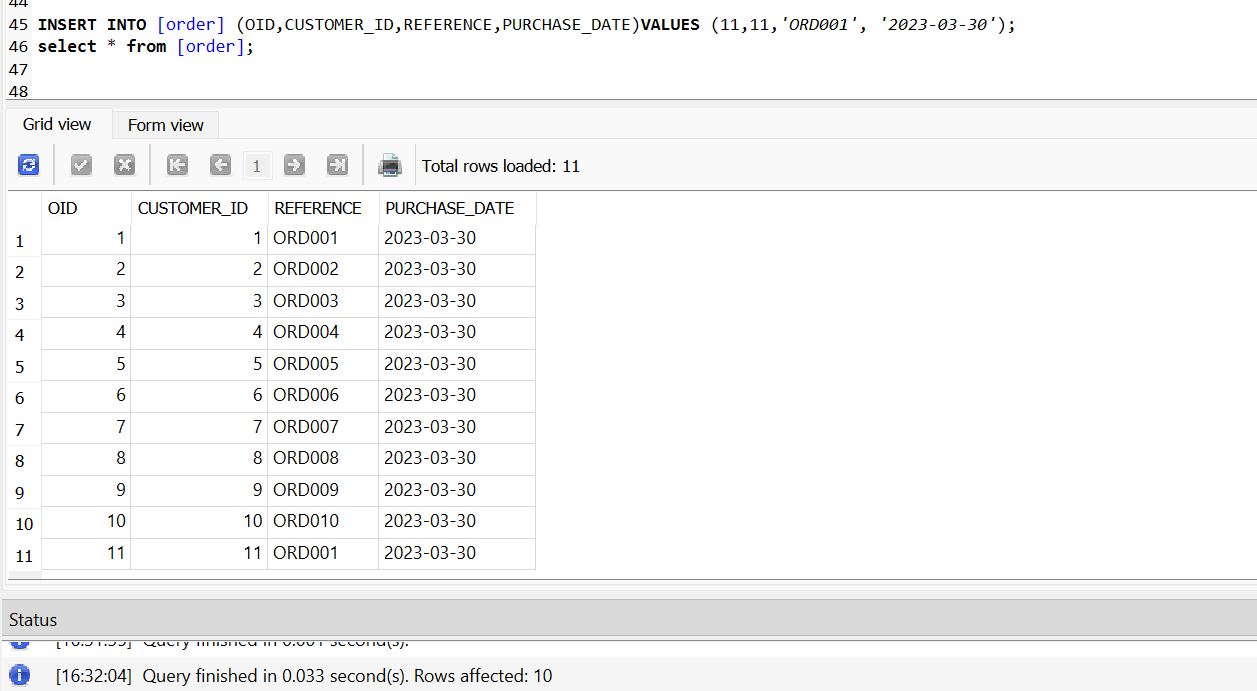
**By following the above schema, give *SQL* Triggers for the following problems**

a)

customer table:

A screenshot of a computer

Description automatically generated

On insert of order table  


Updating of customer using trigger:

A screenshot of a computer code

Description automatically generated

CREATE TRIGGER remove\_order\_if\_negative\_credit

AFTER INSERT ON "order"

BEGIN

UPDATE customer

SET CREDIT\_LIMIT = 0

WHERE CID = NEW.CUSTOMER\_ID AND CREDIT\_LIMIT < 0;

DELETE FROM "order"

WHERE CUSTOMER\_ID IN (SELECT CID FROM customer WHERE CREDIT\_LIMIT < 0);

DELETE FROM "order-items"

WHERE OID NOT IN (SELECT OID FROM "order");

END;

A screenshot of a computer

Description automatically generated

b)

order-audit table create:

CREATE TABLE IF NOT EXISTS order\_audit (

audit\_id INTEGER PRIMARY KEY,

order\_id INTEGER,

operation TEXT,

old\_values TEXT,

new\_values TEXT,

date\_time TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

A screenshot of a computer

Description automatically generated

Trigger statement for update:

CREATE TRIGGER order\_update\_audit

AFTER UPDATE ON [order]

BEGIN

INSERT INTO order\_audit (order\_id, operation, old\_values, new\_values)

VALUES (OLD.OID, 'UPDATE', OLD.REFERENCE || ', ' || OLD.CUSTOMER\_ID || ', ' || OLD.PURCHASE\_DATE,

NEW.REFERENCE || ', ' || NEW.CUSTOMER\_ID || ', ' || NEW.PURCHASE\_DATE);

END;

A close-up of a computer code

Description automatically generated

Order table before updating :

A screenshot of a computer

Description automatically generated

Query to update:

UPDATE "order"

SET REFERENCE = 'ORD123', CUSTOMER\_ID = 2, PURCHASE\_DATE = '2024-03-31'

WHERE OID = 1;

order table after updating:

A screenshot of a computer

Description automatically generated

Order\_audit table after updating:

A screenshot of a computer

Description automatically generated

Trigger statement for delete:

CREATE TRIGGER order\_delete\_audit

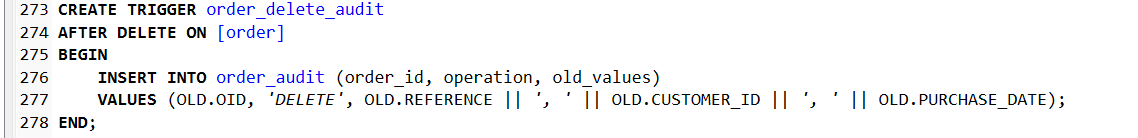
AFTER DELETE ON [order]

BEGIN

INSERT INTO order\_audit (order\_id, operation, old\_values)

VALUES (OLD.OID, 'DELETE', OLD.REFERENCE || ', ' || OLD.CUSTOMER\_ID || ', ' || OLD.PURCHASE\_DATE);

END;



Order table before deleting :

A screenshot of a computer

Description automatically generated

Query to delete:

DELETE FROM "order"WHERE OID = 1;

order table after deleting:

A screenshot of a computer

Description automatically generated

Order\_audit table after deleting:

A screenshot of a computer

Description automatically generated