\*\*\* **EXPERIMENT NO. : 02** \*\*\*

-------------------------------------------------------------------------------------

Author : Atharva Paliwal

Roll No : 40 [5B]

Date : 16 - AUGUST -2020

-------------------------------------------------------------------------------------

**AIM:** To establish a multi-relation database and execute SQL queries involving insertions, deletions and updating on it.

**PROBLEM STATEMENT:**

Establish an environment for executing the queries based on the logical schemata and

the database structuring for the SalesCo database given below...

CUSTOMER (C\_CODE, LNAME, FNAME, C\_AREA, C\_PHONE, BALANCE)

INVOICE (INV\_NUM, C\_CODE, INV\_DATE)

LINE (INV\_NUM, L\_NUM, P\_CODE, L\_UNITS, L\_PRICE)

PRODUCT (P\_CODE, DESCRIPT, P\_DATE, QTY, P\_MIN, P\_PRICE, P\_DISC, V\_CODE)

VENDOR (V\_CODE, V\_NAME, V\_CONTACT, V\_AREA, V\_PHONE, V\_STATE, V\_ORDER)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 02:** For each table of SalesCo database, list all the enforced constraints. (Use

USER CONSTRAINTS view).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT TABLE\_NAME,CONSTRAINT\_NAME,CONSTRAINT\_TYPE,OWNER**

**FROM USER\_CONSTRAINTS**

**WHERE TABLE\_NAME IN('CUSTOMER','INVOICE','VENDOR','PRODUCT','LINE');**

TABLE\_NAME CONSTRAINT\_NAME C OWNER

---------- --------------------------- - -----

CUSTOMER SYS\_C0011365 C CS540

CUSTOMER SYS\_C0011366 C CS540

CUSTOMER SYS\_C0011367 C CS540

CUSTOMER SYS\_C0011368 C CS540

CUSTOMER SYS\_C0011369 C CS540

CUSTOMER SYS\_C0011370 C CS540

CUSTOMER CUSTOMER\_CK\_C\_CODE C CS540

CUSTOMER CUSTOMER\_CK\_C\_AREA C CS540

CUSTOMER CUSTOMER\_PK\_C\_CODE P CS540

INVOICE SYS\_C0011374 C CS540

INVOICE SYS\_C0011375 C CS540

INVOICE SYS\_C0011376 C CS540

INVOICE INVOICE\_CK\_INV\_NUM C CS540

INVOICE INVOICE\_PK\_INV\_NUM P CS540

INVOICE INVOICE\_CUSTOMER\_FK\_C\_CODE R CS540

LINE SYS\_C0011405 C CS540

LINE LINE\_CK\_L\_NUM C CS540

LINE LINE\_CK\_L\_UNITS C CS540

LINE LINE\_CK\_L\_PRICE C CS540

LINE LINE\_PK\_INV\_NUM\_L\_NUM P CS540

LINE LINE\_INVOICE\_FK\_INV\_NUM R CS540

LINE LINE\_PRODUCT\_FK\_P\_CODE R CS540

TABLE\_NAME CONSTRAINT\_NAME C OWNER

---------- --------------------------- - -----

LINE SYS\_C0011401 C CS540

LINE SYS\_C0011402 C CS540

LINE SYS\_C0011403 C CS540

LINE SYS\_C0011404 C CS540

PRODUCT SYS\_C0011391 C CS540

PRODUCT SYS\_C0011392 C CS540

PRODUCT SYS\_C0011393 C CS540

PRODUCT SYS\_C0011394 C CS540

PRODUCT SYS\_C0011395 C CS540

PRODUCT SYS\_C0011396 C CS540

PRODUCT SYS\_C0011397 C CS540

PRODUCT PRODUCT\_CK\_P\_MIN C CS540

PRODUCT PRODUCT\_PK\_P\_CODE P CS540

VENDOR SYS\_C0011380 C CS540

VENDOR SYS\_C0011381 C CS540

VENDOR SYS\_C0011382 C CS540

VENDOR SYS\_C0011383 C CS540

VENDOR SYS\_C0011384 C CS540

VENDOR SYS\_C0011385 C CS540

VENDOR SYS\_C0011386 C CS540

VENDOR VENDOR\_CK\_V\_CODE C CS540

VENDOR VENDOR\_CK\_V\_AREA C CS540

TABLE\_NAME CONSTRAINT\_NAME C OWNER

---------- --------------------------- - -----

VENDOR VENDOR\_CK\_V\_STATE C CS540

VENDOR VENDOR\_PK\_V\_CODE P CS540

47 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 03:** Write SQL code to insert a LINE record - 1009, 1, HW15X, 20, 15.50. What

are the problems encountered? Assume that the 60 units of the product "Hi Veld

Hammer" were supplied by "Indian Masters" located in "KY' at unit price of 15.50 on

January 10, 2020. Minimum stock quantity was anticipated to be 15. The line was

billed to You" located in area 904 with phone 3562098 and a balance of 500.00 on June

22, 2020. The supplier with ID 24992 has a contact named "Your Sibling" with phone

2863322. Write appropriate SELECT statements to showcase the effects of the query.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**INSERT INTO LINE**

**VALUES(1009,1,'HW15X',20,15.50);**

**INSERT INTO LINE**

**\***

**ERROR at line 1:**

**ORA-02291: integrity constraint (CS540.LINE\_PRODUCT\_FK\_P\_CODE) violated -**

**parent key not found**

PROBLEM ENCOUNTERED:

The above error came because the table LINE has a foreign key constraint on table

INVOICE. Also other tables of SalesCo database were also inter-related.

**INSERT INTO VENDOR**

**VALUES(24992,'INDIAN MASTERS','GAURAV',501,2863322,'KY','N');**

1 row created.

**INSERT INTO PRODUCT**

**VALUES('HW15X','HiVeld Hammer','10-JAN-2020',60,15,15.50,0,24992);**

1 row created.

**INSERT INTO CUSTOMER**

**VALUES(10020,'PALIWAL','ATHARVA',904,3562098,500);**

1 row created.

**INSERT INTO INVOICE**

**VALUES(1009,10020,'22-JAN-2020');**

1 row created.

**INSERT INTO LINE**

**VALUES(1009,1,'HW15X',20,15.50);**

1 row created.

**SELECT \* FROM CUSTOMER WHERE C\_CODE=10020;**

C\_CODE LNAME FNAME C\_AREA C\_PHONE BALANCE

--------- ---------- ---------- ---------- ---------- ----------

10020 PALIWAL ATHARVA 904 3562098 500

**SELECT \* FROM INVOICE WHERE INV\_NUM=1009;**

INV\_NUM C\_CODE INV\_DATE

---------- ---------- ---------

1009 10020 22-JAN-20

**SELECT \* FROM VENDOR WHERE V\_CODE=24992;**

V\_CODE V\_NAME V\_CONTACT V\_AREA V\_PHONE V\_ V

---------- --------------- -------------------- ---------- ---------- -- -

24992 INDIAN MASTERS GAURAV 501 2863322 KY N

**SELECT \* FROM PRODUCT WHERE P\_CODE='HW15X' AND V\_CODE=24992;**

P\_COD DESCRIPT P\_DATE QTY P\_MIN P\_PRICE P\_DISC V\_CODE

----- --------------- --------- ---------- ---------- ---------- ---------- ---------

HW15X HiVeld Hammer 10-JAN-20 60 15 15.5 0 24992

**SELECT \* FROM LINE WHERE P\_CODE='HW15X';**

INV\_NUM L\_NUM P\_COD L\_UNITS L\_PRICE

---------- ---------- ----- ---------- ----------

1009 1 HW15X 20 15.5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 04:** Write SQL code that will list P\_CODE, DESCRIPT, V\_CODE for all products

that are some kind of hammers or screws.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT P\_CODE,DESCRIPT,V\_CODE**

**FROM PRODUCT**

**WHERE UPPER(DESCRIPT) LIKE '%HAMMER%' OR UPPER(DESCRIPT) LIKE '%SCREW%';**

P\_COD DESCRIPT V\_CODE

----- ------------------------------ ----------

CH10X Claw Hammer 21225

SH100 Sledge Hammer

MC001 Metal Screw 21225

WC025 2.5in wide Screw 21231

HW15X HiVeld Hammer 24992

5 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 05:** Write the SQL code that will list all products which were added to

inventory during 2020.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT P\_CODE,DESCRIPT,P\_DATE FROM PRODUCT**

**WHERE P\_DATE>='01-JAN-2020';**

P\_COD DESCRIPT P\_DATE

----- ------------------------------ ---------

CL025 Hrd. Spring 1/4in 15-JAN-20

CL050 Hrd. Spring 1/2in 15-JAN-20

CD00X Cordless Drill 20-JAN-20

CH10X Claw Hammer 20-JAN-20

SH100 Sledge Hammer 02-JAN-20

HC100 Hicut Chain Saw 07-FEB-20

PP101 PVC Pipe 20-FEB-20

MC001 Metal Screw 01-MAR-20

WC025 2.5in wide Screw 24-FEB-20

SM48X Steel Malting Mesh 17-JAN-20

HW15X HiVeld Hammer 10-JAN-20

11 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 06:** Write SQL code that will list all invoices billed to customers Elena

Johnson. Your query must account for combining the FNAME and LNAME

attributes while creating and testing the predicate.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT INV\_NUM,INVOICE.C\_CODE,INV\_DATE FROM INVOICE,CUSTOMER**

**WHERE CUSTOMER.C\_CODE=INVOICE.C\_CODE AND UPPER(LNAME)='JOHNSON' AND UPPER(FNAME)='ELENA';**

INV\_NUM C\_CODE INV\_DATE

---------- ---------- ---------

1002 10011 16-JAN-20

1005 10011 17-JAN-20

1008 10011 17-JAN-20

3 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 07:** Write SQL code that will add following records to PRODUCT Table.

ABIII, Power Drill, Today, 15, 5, 125, 0.1, 24992

PP102, PVC Pipe, Tomorrow, 50, 12, 15.25, 0.02, 24992

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**INSERT INTO PRODUCT**

**VALUES('AB111','Power Drill',SYSDATE,15,5,125,0.1,24992);**

1 row created.

**INSERT INTO PRODUCT**

**VALUES('PP102', 'PVC Pipe',SYSDATE+1,50,12,15.25,0.02,24992);**

1 row created.

P\_COD DESCRIPT P\_DATE QTY P\_MIN P\_PRICE P\_DISC V\_CODE

----- ----------- --------- ---------- ---------- ---------- ---------- ---------

AB111 POWER DRILL 16-AUG-20 15 5 125 .1 24992

PP102 PVC PIPE 17-AUG-20 50 12 15.25 .02 24992

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 08:** Write SQL code that will remove the vendor 23119. Explain the problem(s)

encountered (if any). Now, if the policy decision has been to allow such

removals from vendor list by removing the depending relation tuples; modify

the constraints in PRODUCT table. On modifying the constraints, remove the

said vendor and check the changes in database. Revert to the database state

as before executing this query.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**DELETE FROM VENDOR**

**WHERE V\_CODE=23119;**

**DELETE FROM VENDOR**

**\***

**ERROR at line 1:**

**ORA-02292: integrity constraint (CS540.PRODUCT\_VENDOR\_FK\_V\_CODE) violated -**

**child record found**

PROBLEM ENCOUNTERED:

This error is because of foreign key constraints on child table,

giving problem in deleting tuple of parent table.

To remove this error, the foreign key constraint needs to be taken out

**ALTER TABLE PRODUCT**

**DROP CONSTRAINT PRODUCT\_VENDOR\_FK\_V\_CODE;**

Table altered.

**SELECT TABLE\_NAME,CONSTRAINT\_NAME,CONSTRAINT\_TYPE,OWNER**

**FROM USER\_CONSTRAINTS**

**WHERE TABLE\_NAME IN('CUSTOMER') AND CONSTRAINT\_TYPE='R';**

no rows selected

**SELECT \* FROM VENDOR**

**WHERE V\_CODE=23119;**

V\_CODE V\_NAME V\_CONTACT V\_AREA V\_PHONE V\_ V

---------- ---------------- -------------------- ---------- ---------- -- -

23119 Blackman Sisters Svetlana Han 901 3562429 GA Y

**DELETE FROM VENDOR**

**WHERE V\_CODE=23119;**

1 row deleted.

**SELECT \* FROM VENDOR**

**WHERE V\_CODE=23119;**

no rows selected

**INSERT INTO VENDOR**

**VALUES(23119,'Blackman Sisters','Svetlana Han',901,3562429,'GA','Y');**

V\_CODE V\_NAME V\_CONTACT V\_AREA V\_PHONE V\_ V

---------- ---------------- -------------------- ---------- ---------- -- -

23119 Blackman Sisters Svetlana Han 901 3562429 GA Y

**ALTER TABLE PRODUCT**

**ADD CONSTRAINT PRODUCT\_VENDOR\_FK\_V\_CODE FOREIGN KEY(V\_CODE) REFERENCES VENDOR(V\_CODE);**

Table altered.

**SELECT \* FROM VENDOR**

**WHERE V\_CODE=23119;**

V\_CODE V\_NAME V\_CONTACT V\_AREA V\_PHONE V\_ V

---------- ---------------- -------------------- ---------- ---------- -- -

23119 Blackman Sisters Svetlana Han 901 3562429 GA Y

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 09:** Write SQL code that lists all products that were supplied by vendors

belonging to the state 'KY' arranged in increasing sequence of vendor

code. The output should include vendor code, vendor's name, product code,

product description, vendor contact, and inventory purchase date.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT VENDOR.V\_CODE,V\_NAME,P\_CODE,DESCRIPT,V\_CONTACT,P\_DATE FROM VENDOR,PRODUCT**

**WHERE UPPER(V\_STATE)='KY' AND VENDOR.V\_CODE=PRODUCT.V\_CODE**

**ORDER BY VENDOR.V\_CODE;**

V\_CODE V\_NAME P\_COD DESCRIPT V\_CONT P\_DATE

---------- --------------- ----- -------------------- ------------ ---------

21344 Gomez Sons SB900 9.00 in Saw Blade Mark Welder 13-NOV-19

21344 Gomez Sons SB725 7.25in Saw Blade Mark Welder 13-DEC-19

21344 Gomez Sons RF100 Rat Tail File Mark Welder 15-DEC-19

24992 INDIAN MASTERS AB111 Power Drill GAURAV 14-AUG-20

24992 INDIAN MASTERS PP102 PVC Pipe GAURAV 15-AUG-20

24992 INDIAN MASTERS HW15X HiVeld Hammer GAURAV 10-JAN-20

6 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 10:** Write SQL code that will list details of customers who purchased the

products CDOOX or P P101. The output must include customer name

(combination of FName & LName), product code and date of purchase.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT DISTINCT FNAME||' '||LNAME "CUST\_NAME",P\_CODE,INV\_DATE FROM CUSTOMER,INVOICE,LINE**

**WHERE CUSTOMER.C\_CODE=INVOICE.C\_CODE AND INVOICE.INV\_NUM=LINE.INV\_NUM AND (P\_CODE='CD00X' OR P\_CODE='PP101')**

CUST\_NAME P\_COD INV\_DATE

--------------------- ----- ---------

Kathy Smith CD00X 16-JAN-20

Elena Johnson PP101 17-JAN-20

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 11:** Write SQL code that for each customer lists invoices in decreasing order.

You must but keep ascending sequence for customers in the output. The

output should show customer code,invoice number, line units and line price.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT CUSTOMER.C\_CODE,INVOICE.INV\_NUM,L\_UNITS,L\_PRICE FROM CUSTOMER,INVOICE,LINE**

**WHERE CUSTOMER.C\_CODE=INVOICE.C\_CODE AND INVOICE.INV\_NUM=LINE.INV\_NUM**

**ORDER BY INV\_NUM DESC;**

C\_CODE INV\_NUM L\_UNITS L\_PRICE

---------- ---------- ---------- ----------

10020 1009 20 15.5

10011 1008 5 5.87

10011 1008 3 119.95

10011 1008 1 9.95

10015 1007 1 4.99

10015 1007 2 14.99

10014 1006 3 6.99

10014 1006 1 109.92

10014 1006 1 9.95

10014 1006 1 256.99

10011 1005 12 5.87

10018 1004 2 9.95

10018 1004 3 4.99

10012 1003 1 39.95

10012 1003 1 38.95

10012 1003 5 14.99

10011 1002 2 4.99

10014 1001 1 9.95

10014 1001 1 14.99

19 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUERY 12:** Write SQL code that will modify Query-Il, to include the subtotals for each

of the line with invoice numbers. [You are required compute a derived

column SUBTOTAL as L\_UNITS \* LPRICE].

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SELECT CUSTOMER.C\_CODE,INVOICE.INV\_NUM,L\_UNITS,L\_PRICE,LINE.L\_UNITS\*LINE.L\_PRICE AS SUBTOTAL FROM CUSTOMER,INVOICE,LINE**

**WHERE CUSTOMER.C\_CODE=INVOICE.C\_CODE AND INVOICE.INV\_NUM=LINE.INV\_NUM**

**ORDER BY INV\_NUM DESC;**

C\_CODE INV\_NUM L\_UNITS L\_PRICE SUBTOTAL

---------- ---------- ---------- ---------- ----------

10020 1009 20 15.5 310

10011 1008 5 5.87 29.35

10011 1008 3 119.95 359.85

10011 1008 1 9.95 9.95

10015 1007 1 4.99 4.99

10015 1007 2 14.99 29.98

10014 1006 3 6.99 20.97

10014 1006 1 109.92 109.92

10014 1006 1 9.95 9.95

10014 1006 1 256.99 256.99

10011 1005 12 5.87 70.44

10018 1004 2 9.95 19.9

10018 1004 3 4.99 14.97

10012 1003 1 39.95 39.95

10012 1003 1 38.95 38.95

10012 1003 5 14.99 74.95

10011 1002 2 4.99 9.98

10014 1001 1 9.95 9.95

10014 1001 1 14.99 14.99

19 rows selected.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**VIVA VOICE**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUESTION 1: Bring out differences among super key, candidate key and primary key.**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Super Key - A super key is a group of single or multiple keys which identifies rows in a table.

Primary Key - is a column or group of columns in a table that uniquely identify every row in that table.

Candidate Key - is a set of attributes that uniquely identify tuples in a table.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUESTION 2: Differentiate between primary key constraint and unique constraint.**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.Primary key will not accept NULL values whereas Unique key can accept one NULL value.

2.A table can have only primary key whereas there can be multiple unique key on a table.

3.A Clustered index automatically created when a primary key is defined whereas Unique key generates the non-clustered index.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**3. How DROP TABLE differs from TRUNCATE?**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. DROP :

DROP is a DDL(Data Definition Language) command and is used to remove table definition and indexes, data, constraints, triggers etc for that table.

Performance-wise the DROP command is quick to perform but slower than TRUNCATE because it gives rise to complications.

Unlike DELETE we can’t rollback the data after using the DROP command. In the DROP command, table space is freed from memory because it permanently delete table as well as all its contents.

Syntax of DROP command –

DROP TABLE table\_name;

2. TRUNCATE :

TRUNCATE is a DDL(Data Definition Language) command. It is used to delete all the tuples from the table. Like the DROP command, the TRUNCATE command also does not contain a WHERE clause.The TRUNCATE command is faster than both the DROP and the DELETE command. Like the DROP command we also can’t rollback the data after using the this command.

Syntax of TRUNCATE command –

TRUNCATE TABLE table\_name;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**QUESTION 4: How does DEFAULT differ from CHECK constraint?**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The CHECK constraint in SQL is basically used to put a value limit on the values that can be put in a column. A DEFAULT constraint, on the other hand, is used to assign default values to the columns.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**INFERENCES:**

Studied and learnt about:

* Establishing an environment for executing the queries based on the logical schemata and the database structuring.
* Differencesamong super key, candidate key and primary key.