5.3 PL\SQL BLOCKS

1.1. TRIGGERS

This "INS_CP" Trigger will be called after New Data is inserted into "CARTPRODUCT" table and update the value of "QUANTITY" in "PRODUCT" table and value of "TTL_AMT" and "TTL_QTY" in "CART" table.

```
CREATE TRIGGER INS_CP AFTER INSERT ON CARTPRODUCT FOR EACH ROW
BEGIN
        DECLARE SELL INT;

        SELECT QUANTITY INTO SELL FROM PRODUCT WHERE PROD_ID=NEW.PROD_ID;
        SET SELL = SELL - NEW.QTY;

        UPDATE PRODUCT SET QUANTITY = SELL WHERE PROD_ID=NEW.PROD_ID;

        UPDATE CART SET TTL_AMT = TTL_AMT+(NEW.QTY*(SELECT PRICE FROM PRODUCT
        WHERE PROD_ID=NEW.PROD_ID)) WHERE CART_ID=NEW.CART_ID;

        UPDATE CART SET TTL_QTY=TTL_QTY+NEW.QTY WHERE CART_ID=NEW.CART_ID;
END
```

BEFORE :

CART_ID	PROD_ID	QTY
1	2	1
1	9	1
1	14	1
1	15	2
2	5	1
2	10	2

CART_ID	TTL_AMT	TTL_QTY
1	193000.00	5
2	180000.00	4
3	67000.00	1
4	90000.00	3
5	1699.00	1

14	3	6	Earbuds	10	3000.00	20.00
15	1	7	Reno	9	30000.00	5.00
16	4	7	A23	3	20000.00	0.00
17	2	7	Narzo	21	23999.00	20.00
18	2	8	V5	12	19999.00	0.00
19	2	8	V9	4	26000.00	25.00

AFTER :

INSERT INTO CARTPRODUCT VALUES (1,17,2);

CART_ID	PROD_ID	QTY
1	2	1
1	9	1
1	14	1
1	15	2
1	17	2
2	5	1
2	10	2

CART_ID	TTL_AMT	TTL_QTY
1	240998.00	7
2	180000.00	4
3	67000.00	1
4	90000.00	3
5	1699.00	1

14	3	6	Earbuds	10	3000.00	20.00
15	1	7	Reno	9	30000.00	5.00
16	4	7	A23	3	20000.00	0.00
17	2	7	Narzo	19	23999.00	20.00
18	2	8	V5	12	19999.00	0.00
19	2	8	V9	4	26000.00	25.00

1.2 For "INSERT" , "DELETE" and "UPDATE" action performed in "USER_INFO" table for that we require one "LOG" table to store the information that which action is performed on which "USER_ID" , "DATE" and "TIME".

```
CREATE TABLE LOG (
        LOG_ID INT AUTO_INCREMENT PRIMARY KEY ,
        USER_ID INT NOT NULL ,
        ACTION VARCHAR(10) NOT NULL ,
        CUR_DATE DATETIME NOT NULL
);
```

1.2.1 THIS "INS_LOG" TRIGGER WILL BE CALLED WHEN NEW DATA IS INSERTED INTO "USER_INFO" TABLE AND GET THE "DATE" AND "TIME" IN "LOG" TABLE.

```
CREATE TRIGGER INS_LOG AFTER INSERT ON USER_INFO FOR EACH ROW
BEGIN
        INSERT INTO LOG VALUES(NULL,NEW.USER_ID,'Inserted',NOW());
END
```

BEFORE :

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat

AFTER :

CALL NEW_USER_INFO(16,'Hemal','Makwana','hemal006@gmail.com',
'2001-08-20',7600080522,NULL,'S/5,Shrusti-5',380008,'Ahmedabad','Gujarat');

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat
16	Hemal	Makwana	hemal006@gmail.com	2001-08-20	7600080522	NULL	S/5,Shrusti-5	380008	Ahmedabad	Gujarat

LOG_ID	USER_ID	ACTION	CUR_DATE
1	16	Inserted	2020-10-31 12:09:29

1.2.2 THIS "UPD_LOG" TRIGGER WILL BE CALLED WHEN DATA IS UPDATED FROM "USER_INFO" TABLE AND GET THE "DATE" AND "TIME" IN "LOG" TABLE.

CREATE TRIGGER UPD_LOG AFTER UPDATE ON USER_INFO FOR EACH ROW
BEGIN
        INSERT INTO LOG VALUES(NULL,NEW.USER_ID,'Updated',NOW());
END

BEFORE :

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat
16	Hemal	Makwana	hemal006@gmail.com	2001-08-20	7600080522	NULL	S/5,Shrusti-5	380008	Ahmedabad	Gujarat

AFTER :

UPDATE USER_INFO SET PHN_NO1 = 7600080411 WHERE USER_INFO.USER_ID = 16;

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat
16	Hemal	Makwana	hemal006@gmail.com	2001-08-20	7600080411	NULL	S/5,Shrusti-5	380008	Ahmedabad	Gujarat

LOG_ID	USER_ID	ACTION	CUR_DATE
1	16	Inserted	2020-10-31 12:09:29
2	16	Updated	2020-10-31 12:11:42

1.2.3 THIS "DEL_LOG" TRIGGER WILL BE CALLED WHEN DATA IS DELETED FROM "USER_INFO" TABLE AND GET THE "DATE" AND "TIME" IN "LOG" TABLE.

CREATE TRIGGER DEL_LOG BEFORE DELETE ON USER_INFO FOR EACH ROW
BEGIN
        INSERT INTO LOG VALUES(NULL,OLD.USER_ID,'Deleted',NOW());
END

BEFORE :

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat
16	Hemal	Makwana	hemal006@gmail.com	2001-08-20	7600080522	NULL	S/5,Shrusti-5	380008	Ahmedabad	Gujarat

AFTER :

DELETE FROM USER_INFO WHERE USER_INFO.USER_ID = 16

12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat

LOG_ID	USER_ID	ACTION	CUR_DATE
1	16	Inserted	2020-10-31 12:09:29
2	16	Updated	2020-10-31 12:11:42
3	16	Deleted	2020-10-31 12:14:23

2. PROCEDURE

This "NEW_USER_INFO" Procedure Insert The Values Into "USER_INFO" Table With Checking if "EMAIL_ID" Is Valid Or Not and also in "CUSTOMER" table.

```
CREATE PROCEDURE NEW_USER_INFO(
        IN USER_ID INT, IN FIRST_NAME VARCHAR(10), IN LAST_NAME VARCHAR(10),
        IN EMAIL_ID VARCHAR(30), IN DOB DATE, IN PHN_NO1 DECIMAL(10),
        IN PHN_NO2 DECIMAL(10), IN ADDRESS VARCHAR(100), IN PINCODE DECIMAL(6),
        IN CITY VARCHAR(30), IN STATE VARCHAR(30))
BEGIN
        DECLARE ID INT;
        IF EMAIL_ID LIKE '%@gmail.com' THEN
                INSERT INTO USER_INFO VALUES
                        (USER_ID, FIRST_NAME, LAST_NAME, EMAIL_ID, DOB,
                        PHN_NO1,PHN_NO2,ADDRESS,PINCODE,CITY,STATE);
                SELECT MAX(CUST_ID) INTO ID FROM CUSTOMER;
                SET ID=ID+1;
                INSERT INTO CUSTOMER VALUES(ID,USER_ID);
                SELECT 'YOUR DATA IS INSERTED' AS MESSAGE;
        ELSE
                SELECT 'YOUR EMAIL ID IS NOT VALID' AS ERROR;
        END IF;
END
```

BEFORE :

11	Dhruv	Kumar	dhruv@gmail.com	NULL	9787300760	NULL	S/45,J G road	335009	Surat	Gujarat
12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat

7	12
8	13
9	14
10	15

AFTER :

CALL NEW_USER_INFO(16,'Hemal','Makwana',**'hemal006@gmail.com'**,
'2001-08-20',7600080411,NULL,'S/5,Shrusti-5',380008,'Ahmedabad','Gujarat');

11	Dhruv	Kumar	dhruv@gmail.com	NULL	9787300760	NULL	S/45,J G road	335009	Surat	Gujarat
12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat
16	Hemal	Makwana	hemal006@gmail.com	2001-08-20	7600080411	NULL	S/5,Shrusti-5	380008	Ahmedabad	Gujarat

7	12
8	13
9	14
10	15
11	16

MESSAGE

YOUR DATA IS INSERTED

BEFORE :

11	Dhruv	Kumar	dhruv@gmail.com	NULL	9787300760	NULL	S/45,J G road	335009	Surat	Gujarat
12	Raj	Makwana	raj@gmail.com	1999-05-27	9638312340	NULL	A/2,A Apartment	320008	Ahmedabad	Gujarat
13	Ajay	Mistry	ajay@gmail.com	2001-01-01	6928300760	6928300761	C/3,Navratna nagar	335009	Surat	Gujarat
14	Ronak	Patel	ronakp@gmail.com	2000-11-30	7383478760	7383478761	H/12,Red Apartment	300018	Baroda	Gujarat
15	Jinal	Parmar	jinal@gmail.com	NULL	9996300760	NULL	B/15,Dharmabhumi	335009	Surat	Gujarat

AFTER :

CALL NEW_USER_INFO(17,'Parth','Prajapati',**'parth997@yahoo.com'**,
'1999-08-29',8758854412,8787454521,'C-3/3,Navratna app.',380026,'Ahmedabad','Gujarat');

ERROR

YOUR EMAIL ID IS NOT VALID

3. CURSORS

This "INFORMATION" Cursor gives the information about Customer "First Name", "Last name" and "Total Amount" of his Order.

```
CREATE PROCEDURE INFORMATION()
BEGIN
        DECLARE FIRST_NAME VARCHAR(10);
        DECLARE LAST_NAME VARCHAR(10);
        DECLARE TTL_AMT DECIMAL(11,2) ;
        DECLARE FINISHED INT(2);
```

```
DECLARE INFOR CURSOR FOR SELECT U.FIRST_NAME,U.LAST_NAME,P.TTL_AMT
FROM USER_INFO U, CUSTOMER C, ORDERS O, PAYMENT P
WHERE U.USER_ID=C.USER_ID
AND C.CUST_ID=O.CUST_ID
AND P.ORD_ID=O.ORD_ID;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET FINISHED=1;

OPEN INFOR;

EXITS: LOOP
        FETCH INFOR INTO FIRST_NAME,LAST_NAME,TTL_AMT;

        IF FINISHED=1 THEN
                LEAVE EXITS;
        END IF;

        SELECT FIRST_NAME,LAST_NAME,TTL_AMT AS 'TOTAL_AMOUNT';
END LOOP EXITS;

        CLOSE INFOR;
END
```

OUTPUT :

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
Akash	Mistry	193000.00

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
Ronak	Patel	180000.00

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
Kavish	Khatri	67000.00

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
Dhruv	Kumar	90000.00

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
Ronak	Agnani	1699.00

4. FUNCTIONS

This "REMAINING" Function prints the subcategory name, product name, old quantity and new quantity of particular "PROD_ID" from "CartProduct" table .

```
CREATE FUNCTION REMAINING(PROD_ID INT) RETURNS INT
BEGIN
        DECLARE STOCK INT;
        DECLARE SELL INT;
        DECLARE REMAIN INT;

        SELECT P.QUANTITY INTO STOCK FROM PRODUCT P WHERE P.PROD_ID=PROD_ID;
        SELECT SUM(C.QTY) INTO SELL FROM CARTPRODUCT C WHERE C.PROD_ID=PROD_ID;

        SET REMAIN=STOCK-SELL;

        RETURN REMAIN;
END
```

INPUT :

FOR PROD_ID = 9;

```
SELECT DISTINCT S.SCAT_NAME,P.PROD_NAME,
P.QUANTITY AS 'OLD_QTY',REMAINING(9) AS 'NEW_QTY'
FROM PRODUCT P,CARTPRODUCT C,SUBCATEGORY S
WHERE P.PROD_ID=9 AND P.PROD_ID=C.PROD_ID AND P.SCAT_ID=S.SCAT_ID;
```

SCAT_NAME	PROD_NAME	OLD_QTY	NEW_QTY
Mini Laptop	HP Mini	3	1

FOR PROD_ID = 22;

```
SELECT DISTINCT S.SCAT_NAME,P.PROD_NAME,
P.QUANTITY AS 'OLD_QTY',REMAINING(22) AS 'NEW_QTY'
FROM PRODUCT P,CARTPRODUCT C,SUBCATEGORY S
WHERE P.PROD_ID=22 AND P.PROD_ID=C.PROD_ID AND P.SCAT_ID=S.SCAT_ID;
```

SCAT_NAME	PROD_NAME	OLD_QTY	NEW_QTY
Apple	iphone XR	1	0