

## Practical NO 08

### Problem Statement :-

Database trigger (All types : Row level and Statement level trigger , Before and after trigger ).

Write a database trigger on library table. The system should keep track of the

records that are being updated or deleted .The old value of updated or deleted records should

be added in library \_audit table.

library table(B\_id,Title,Edition ,NO\_of\_Copies) library\_audit table

(B\_id,Title,Edition ,NO\_of\_Copies,Date\_of\_mod,Type\_of\_operation,user who performed the operation)

Submitted By :- 1. Bhushan Asati  
2. Asha Mandale  
3. Sayaji Dhandge

SQL\*Plus: Release 10.2.0.1.0 - Production on Sat Nov 20 14:55:46 2021

Copyright (c) 1982, 2005, Oracle. All rights reserved.

SQL> connect

Enter user-name: system

Enter password:

Connected.

SQL> CREATE TABLE LIBRARY(

2 B\_ID NUMBER(4),

3 B\_NAME VARCHAR(20),

4 A\_NAME VARCHAR(20),

5 NO\_OF\_COPY NUMBER(4),

6 EDITION VARCHAR(10));

Table created.

SQL> INSERT INTO LIBRARY

VALUES(&B\_ID,&B\_NAME,&A\_NAME,&NO\_OF\_COPY,&EDITION);

Enter value for b\_id: 1

Enter value for b\_name: 'JAVA'

Enter value for a\_name:

'Pravin' Enter value for

no\_of\_copy: 10 Enter value for

edition: '2nd' old 1: INSERT

```
INTO LIBRARY  
VALUES(&B_ID,&B_NAME,&A_NAME,&NO_OF_COPY,&EDITION)  
new 1: INSERT INTO LIBRARY VALUES(1,'JAVA','Pravin',10,'2nd')
```

1 row created.

```
SQL> INSERT INTO LIBRARY
VALUES(&B_ID,&B_NAME,&A_NAME,&NO_OF_COPY,&EDITION);
Enter value for b_id: 2
Enter value for b_name: 'CNS'
Enter value for a_name: 'Mayur'
Enter value for no_of_copy: 15
Enter value for edition: '5th'
old 1: INSERT INTO LIBRARY
VALUES(&B_ID,&B_NAME,&A_NAME,&NO_OF_COPY,&EDITION)
new 1: INSERT INTO LIBRARY VALUES(2,'CNS','Mayur',15,'5th')
```

1 row created.

```
SQL> INSERT INTO LIBRARY
VALUES(&B_ID,&B_NAME,&A_NAME,&NO_OF_COPY,&EDITION);
Enter value for b_id: 3
Enter value for b_name: 'CPP'
Enter value for a_name: 'Asha'
Enter value for no_of_copy: 15
Enter value for edition: '7th'
old 1: INSERT INTO LIBRARY
VALUES(&B_ID,&B_NAME,&A_NAME,&NO_OF_COPY,&EDITION)
new 1: INSERT INTO LIBRARY VALUES(3,'CPP','Asha',15,'7th')
```

1 row created.

```
SQL> SELECT * FROM LIBRARY;
```

B_ID	B_NAME	A_NAME	NO_OF_COPY	EDITION
1	JAVA	Pravin	10	2nd
2	CNS	Mayur	15	5th
3	CPP	Asha	15	7th

```
SQL> CREATE TABLE LIBRARY_AUDIT(
2 B_ID NUMBER(4),
3 B_NAME VARCHAR(20),
4 OLD_A_NAME VARCHAR(20),
5 NEW_A_NAME VARCHAR(20),
6 NO_OF_COPY NUMBER(4),
7 EDITION VARCHAR(10),
8 USER_NAME VARCHAR(20),
9 OPERATION VARCHAR(10),
10 MO_DATE VARCHAR(20));
```

Table created.

```
SQL> CREATE OR REPLACE TRIGGER LIBRARY_AUDIT
  2 BEFORE DELETE OR UPDATE ON LIBRARY
  3 FOR EACH ROW
  4 DECLARE
  5   L_USER VARCHAR(20);
  6   L_DATE VARCHAR(30);
  7 BEGIN
  8   SELECT USER,TO_CHAR(SYSDATE,'DD/MM/YYYY HH24:MI:SS') INTO
L_USER,L_DATE FROM DUAL;
  9
 10 IF DELETING THEN
 11   INSERT INTO LIBRARY_AUDIT
(B_ID,B_NAME,OLD_A_NAME,NO_OF_COPY,EDITION,USER_NAME,OPERATION,MO_DATE)
VALUES
(:OLD.B_ID,:OLD.B_NAME,:OLD.A_NAME,:OLD.NO_OF_COPY,:OLD.EDITION,L_USER,'DE
L ETE',L_DATE);
 12
 13 ELSIF UPDATING THEN
 14   INSERT INTO LIBRARY_AUDIT
(B_ID,B_NAME,OLD_A_NAME,NEW_A_NAME,NO_OF_COPY,EDITION,USER_NAME,OPERATION,
M O_DATE) VALUES
(:OLD.B_ID,:OLD.B_NAME,:OLD.A_NAME,:NEW.A_NAME,:OLD.NO_OF_COPY,:OLD.EDITIO
N
,L_USER,'UPDATE',L_DATE);
 15
 16 END IF;
 17 END;
 18 /
```

Trigger created.

```
SQL> SELECT * FROM LIBRARY;
```

B_ID	B_NAME	A_NAME	NO_OF_COPY	EDITION
1	JAVA	Pravin	10	2nd
2	CNS	Mayur	15	5th
3	CPP	Asha	15	7th

```
SQL> SELECT * FROM LIBRARY_AUDIT;
```

no rows selected

```
SQL> DELETE FROM LIBRARY WHERE B_ID=3;
```

1 row deleted.

```
SQL> SELECT * FROM LIBRARY;
      B_ID B_NAME
```

	A_NAME	NO_OF_COPY	EDITION
1 JAVA	Pravin	10	2nd
2 CNS	Mayur	15	5th

```
SQL> UPDATE LIBRARY SET A_NAME='XYZ' WHERE B_ID=1;
```

1 row updated.

```
SQL> SELECT * FROM LIBRARY;
```

B_ID	B_NAME	A_NAME	NO_OF_COPY	EDITION
1	JAVA	XYZ	10	2nd
2	CNS	Mayur	15	5th

--2ND:

Problem Statement:

To create a trigger to check the number of copies available before issuing a book if number of copies issuing is greater than number of book available then issue the number of books which are available.

library table(B\_id,Title,Edition ,NO\_of\_Copies)

Transaction

table(T\_id,B\_id,Title,issue\_or\_return,NO\_of\_Copies\_issue\_or\_return)

```
SQL> CREATE TABLE TRANSACTION(
  2  T_ID NUMBER(4),
  3  B_ID NUMBER(4),
  4  I_R VARCHAR(4),
  5  NO_OF_COPY_IORR NUMBER(4));
```

Table created.

```
SQL> CREATE OR REPLACE TRIGGER TRANS_TRI
  2  BEFORE INSERT
  3  ON TRANSACTION
  4  FOR EACH ROW
  5  DECLARE
  6  NO NUMBER(4);
```

```

7 BEGIN
8 IF :NEW.I_R = 'I' THEN
9 SELECT NO_OF_COPY INTO NO FROM LIBRARY WHERE B_ID = :NEW.B_ID;
10 IF :NEW.NO_OF_COPY_IORR > NO THEN
11 :NEW.NO_OF_COPY_IORR := NO;
12 DBMS_OUTPUT.PUT_LINE('NOT ENOUGH BOOKS');
13
14 ELSE
15 DBMS_OUTPUT.PUT_LINE(' BOOKS ARE AVAILABLE');
16 END IF;
17 END IF;
18 END;
19 /

```

Trigger created.

```
SQL> INSERT INTO TRANSACTION VALUES(4,1,'I',11);
```

1 row created.

```
SQL> SELECT * FROM TRANSACTION;
```

T_ID	B_ID	I_R	NO_OF_COPY_IORR
4	1	I	10

```
SQL>
```

```
--3:
```

Problem Statement:

Write a database trigger on library to update the number of copies available

after the book is issued or returned.

library table(B\_id,Title,Edition ,NO\_of\_Copies)

Transaction

table(T\_id,B\_id,Title,issue\_or\_return,NO\_of\_Copies\_issue\_or\_return)

```
SQL> CREATE OR REPLACE TRIGGER UPDATE_TRI
```

```

2 AFTER INSERT
3 ON TRANSACTION
4 FOR EACH ROW
5 DECLARE
6 NO NUMBER(3);
7 BEGIN
8 SELECT NO_OF_COPY INTO NO FROM LIBRARY WHERE B_ID = :NEW.B_ID;

```

```
9  IF :NEW.I_R = 'I' THEN
```

```

10
11  UPDATE LIBRARY SET NO_OF_COPY = NO - :NEW.NO_OF_COPY_IORR WHERE B_ID =
:NEW.B_ID;
12  ELIF :NEW.I_R = 'R' THEN
13  UPDATE LIBRARY SET NO_OF_COPY = NO + :NEW.NO_OF_COPY_IORR WHERE B_ID =
:NEW.B_ID;
14  END IF;
15  END;
16 /

```

Trigger created.

```
SQL> INSERT INTO TRANSACTION VALUES(2,2,'I',13);
```

1 row created.

```
SQL> SELECT * FROM TRANSACTION;
```

T_ID	B_ID	I_R	NO_OF_COPY_IORR
4	1	I	10
2	2	I	13

```
SQL>
```



