

# COMP23111 - Lab Exercise 5

James, Peach

**ID:** 8974863

December 12, 2014

# 1 Part 1 - SQL Triggers

```

SQL>
SQL> DROP TRIGGER ContractInfo_duration_calc;

Trigger dropped.

SQL>
SQL> UPDATE ContractInfo SET duration = 0;

13 rows updated.

SQL> DELETE FROM ContractInfo WHERE HasContract = 'Goldfrat' and date_from = '10-oct-2008' and date_to = '9-oct-2010';

1 row deleted.

SQL>
SQL> CREATE OR REPLACE TRIGGER ContractInfo_duration_calc
2 BEFORE
3 UPDATE OR INSERT
4 ON ContractInfo
5 FOR EACH ROW
6 DECLARE
7 BEGIN
8     :NEW.duration := :NEW.date_to - :NEW.date_from;
9 END ContractInfo_duration_calc;
10 /

Trigger created.

SQL>
SQL> SHOW ERRORS TRIGGER ContractInfo_duration_calc
No errors.
SQL>
SQL> -- SHOW IT WORKS
SQL> SELECT * FROM ContractInfo;

```

HASCONTRACT	DATE_FROM	DATE_TO	DURATION
Goldfrat	10-OCT-10	09-OCT-14	0
Simon Palm	20-AUG-05	19-AUG-07	0
Simon Palm	20-SEP-12	19-AUG-14	0
Flut	10-AUG-09	09-SEP-14	0
John Cliff	10-MAR-06	10-APR-12	0
Jay Blancard	01-JAN-04	20-JUN-14	0
Palmer John	12-FEB-06	01-JUN-08	0
Zero7	31-DEC-11	26-FEB-16	0
JZ	24-MAR-05	30-DEC-07	0
Scandal	22-OCT-12	19-NOV-16	0
JK Rowl	19-APR-03	06-JAN-09	0

```

HASCONTRACT      DATE_FROM DATE_TO      DURATION
-----
PJ Blox           21-JUN-08 16-JUL-13          0

12 rows selected.

SQL>
SQL> UPDATE ContractInfo
2 SET duration = duration;

12 rows updated.

SQL>
SQL> INSERT INTO ContractInfo VALUES
2 ('Goldfrat', '10-oct-2008', '09-oct-2010', 0);

1 row created.

SQL>
SQL> SELECT * FROM ContractInfo;

```

HASCONTRACT	DATE_FROM	DATE_TO	DURATION
Goldfrat	10-OCT-10	09-OCT-14	1460
Simon Palm	20-AUG-05	19-AUG-07	729
Simon Palm	20-SEP-12	19-AUG-14	698
Flut	10-AUG-09	09-SEP-14	1856
John Cliff	10-MAR-06	10-APR-12	2223
Jay Blancard	01-JAN-04	20-JUN-14	3823
Palmer John	12-FEB-06	01-JUN-08	840
Zero7	31-DEC-11	26-FEB-16	1518
JZ	24-MAR-05	30-DEC-07	1011
Scandal	22-OCT-12	19-NOV-16	1489
JK Rowl	19-APR-03	06-JAN-09	2089

```

HASCONTRACT      DATE_FROM DATE_TO      DURATION
-----
PJ Blox           21-JUN-08 16-JUL-13      1851
Goldfrat          10-OCT-08 09-OCT-10       729

13 rows selected.

SQL>
SQL>
SQL>
SQL> SPPOOL OFF

```

## 2 Part 2 - SQL Procedures

### 2.1 A)

```

SQL>
SQL> CREATE OR REPLACE PROCEDURE ContractInfo_insert
2  (
3      newHasContract IN ContractInfo.hasContract%TYPE,
4      newDate_from IN ContractInfo.date_from%TYPE,
5      newDate_to IN ContractInfo.date_to%TYPE
6  )
7  IS
8      no_contracts_clash NUMBER :=0;
9  BEGIN
10     IF newDate_to > newDate_from
11     THEN -- dates are in the correct order
12         SELECT count(1) into no_contracts_clash
13         FROM ContractInfo
14         WHERE
15             ( newDate_to BETWEEN date_from and date_to
16              OR newDate_from BETWEEN date_from and date_to)
17         AND newHasContract = hasContract;
18
19         IF no_contracts_clash < 1
20         THEN
21             INSERT INTO ContractInfo (hasContract,date_from,date_to)
22             VALUES (newHasContract,newDate_from,newDate_to);
23         END IF;
24     END IF;
25 END;
26 /

```

Procedure created.

```

SQL>
SQL> SHOW ERRORS PROCEDURE ContractInfo_insert
No errors.
SQL>
SQL> -- SHOW IT WORKS
SQL> DELETE FROM ContractInfo
2  WHERE hasContract = 'Coldfrat'
3  AND date_from = '10-oct-2008'
4  AND date_to = '09-oct-2010';

```

0 rows deleted.

```

SQL>
SQL> SELECT * FROM ContractInfo;

```

HASCONTRACT	DATE_FROM	DATE_TO	DURATION
Goldfrat	10-OCT-10	09-OCT-14	1460
Simon Palm	20-AUG-05	19-AUG-07	729
Simon Palm	20-SEP-12	19-AUG-14	698
Flut	10-AUG-09	09-SEP-14	1856
John Cliff	10-MAR-06	10-APR-12	2223
Jay Blancard	01-JAN-04	20-JUN-14	3823
Palmer John	12-FEB-06	01-JUN-08	840
Zero7	31-DEC-11	26-FEB-16	1518
JZ	24-MAR-05	30-DEC-07	1011
Scandal	22-OCT-12	19-NOV-16	1489
JK Rowl	19-APR-03	06-JAN-09	2089
-----			
HASCONTRACT	DATE_FROM	DATE_TO	DURATION
PJ Blox	21-JUN-08	16-JUL-13	1851

12 rows selected.

```

SQL>
SQL> EXEC ContractInfo_insert('Goldfrat','10-oct-2008','09-oct-2010');

```

PL/SQL procedure successfully completed.

```

SQL>
SQL> SELECT * FROM ContractInfo;

```

HASCONTRACT	DATE_FROM	DATE_TO	DURATION
Goldfrat	10-OCT-10	09-OCT-14	1460
Simon Palm	20-AUG-05	19-AUG-07	729
Simon Palm	20-SEP-12	19-AUG-14	698
Flut	10-AUG-09	09-SEP-14	1856
John Cliff	10-MAR-06	10-APR-12	2223
Jay Blancard	01-JAN-04	20-JUN-14	3823
Palmer John	12-FEB-06	01-JUN-08	840
Zero7	31-DEC-11	26-FEB-16	1518
JZ	24-MAR-05	30-DEC-07	1011
Scandal	22-OCT-12	19-NOV-16	1489
JK Rowl	19-APR-03	06-JAN-09	2089
-----			
HASCONTRACT	DATE_FROM	DATE_TO	DURATION
PJ Blox	21-JUN-08	16-JUL-13	1851
Goldfrat	10-OCT-08	09-OCT-10	729

13 rows selected.

```

SQL>
SQL>
SQL> SPOOL OFF

```

## 2.2 B)

```

SQL>
SQL> CREATE OR REPLACE VIEW AlbumDistribution AS
2  SELECT al.album_id, al.title,'v' is_distributed_as
3  FROM Album al
4  JOIN VinylAlbum va ON va.album_id = al.album_id
5  UNION

```

```

6  SELECT al.album_id, al.title,'c' is_distributed_as
7  FROM Album al
8  JOIN CDAlbum ca ON ca.album_id = al.album_id
9  UNION
10 SELECT al.album_id, al.title,'t' is_distributed_as
11 FROM Album al
12 JOIN TapeAlbum ta ON ta.album_id = al.album_id;

```

View created.

```

SQL>
SQL> --SHOW IT WORKS
SQL> COLUMN is_distributed_as FORMAT a20
SQL> SELECT * FROM AlbumDistribution;

```

ALBUM	TITLE	IS_DISTRIBUTED_AS
1c	20 People in a Field	c
1t	20 People in a Field	t
1v	20 People in a Field	v
2c	My Feet	c
2v	My Feet	v
3c	Debut	c

6 rows selected.

```

SQL>
SQL> CREATE OR REPLACE PROCEDURE Album_PlayList
2  (
3      album_type IN VARCHAR2,
4      album_name IN VARCHAR2
5  )
6  IS
7      sel_album_id VARCHAR2(5);
8  BEGIN
9      SELECT a.album_id INTO sel_album_id
10     FROM Album a
11     JOIN AlbumDistribution ad ON ad.album_id = a.album_id
12     WHERE ad.title LIKE album_name AND ad.is_distributed_as = album_type AND ROWNUM = 1;
13
14  DECLARE
15      CURSOR tracks IS
16          SELECT ft.released_title trackName,a.title albumName,at.sequence
17          FROM AlbumTrack at
18          JOIN Album a ON a.album_id = at.album_id
19          JOIN FinishedTrack ft ON ft.originatesFrom = at.track_id AND ft.version = at.version
20          WHERE at.album_id = sel_album_id;
21
22  BEGIN
23      FOR track IN tracks
24      LOOP
25          DBMS_OUTPUT.put_line (
26              track.sequence || ', ' || track.trackName || ' > ' || track.albumName);
27      END LOOP;
28  END;
29 END;
30 /

```

Procedure created.

```

SQL>
SQL> SHOW ERRORS PROCEDURE Album_PlayList
No errors.
SQL>
SQL> EXEC Album_PlayList('c','%F%')
1, Hello > 20 People in a Field
2, Extended Hello > 20 People in a Field
3, Me > 20 People in a Field
4, You > 20 People in a Field
5, Them > 20 People in a Field
6, Us > 20 People in a Field
7, How > 20 People in a Field
8, Be > 20 People in a Field
9, Peace > 20 People in a Field
10, Smile > 20 People in a Field

```

PL/SQL procedure successfully completed.

```

SQL>
SQL> SPPOOL OFF

```