

Task-#11 Triggers, views and Exceptions

Aim:-

To conduct events, views, exceptions or can operations for detecting phenomenon.

To create a trigger in PL/SQL that automatically inserts a new record in the match-results when a new record is inserted into the match table.

CREATE OR REPLACE TRIGGER insertion - match-
results AFTER INSERT ON Match.

FOR EACH ROW

BEGIN

INSERT INTO match-results (Match ID, Team ID,
Result)

VALUES (:new, match ID; new: Team ID, pending); --
Assuming a default value of pending for result. End;

To create a view that displays the details of player along with their team details.

SQL STATEMENT VIEW Players Team Details AS SELECT player.coach,
p.captain AS player captain. (SELECT t.name FROM team t.
WHERE t.teamID = p.teamID) AS team name. (SELECT b.
coach FROM t. WHERE t.team ID = p.team ID).
AS team coach, (SELECT t.captain FROM team t. WHERE
t.team =). p.team ID = p AS team captain FROM player p)

SQL > SELECT from Player Team Details;

To write a recursive PL/SQL procedure to even mutual player IDs.

CREATE OR REARRANGE PROCEDURES

GET even player Id's for Tournament (in, Tournament, id
Number).

BEGIN

out-even-player-ids := Sys. DCI Number T(); --
Initialize the collection.

DELAET

Tournament -id Number = 123; -- Replace with the desired
Tournament: ID even-player -ids sys. DCI Num -Re: RCLST;

BEGIN

Geb Even Player ID's For Tournament (tournament id, even,
Player -id's).

FOR IN 1 --even -player -ids + COUNT loop.
DBMS = output . put -LINE ("even player ID: Even -Player
ID")
END loop
END;

VEL TECHI - CSE	
EX NO.	1
PERFORMANCE	5
RESULT / MARKS	3
VIVA VOCE	5
RECORD	4
TO DO	1

Result:- Thus, the Trigger, views and Exceptions experiment
was successfully completed results are verified.