Dept of Computer Engg. Sub : DBMS Practical & Oral Exam – April – 2025

Example no 1a : Player Relation (T1)

Pid Number(4)	Pname Varchar2(20)	Role Varchar2(20)	Age Number(2)	Runs Number(10)	Wicket Number(10)	Tid Number(3)
1001	Sachin	Batsman	25	10000	45	101
1002	Jaywardhne	Captain	29	5000		102
1003	Kapil	Bowler	30	2000	470	103
1004	Clark	Batsman	27	3000	120	104
1005	Gautam	Batsman	28	20000		101
1006	Rahul	Batsman	22	15000		101
1007	Vas	Bowler	28	3000	480	102
1008	Flintop	Bowler	24	6200	190	103
1009	Hussy	Batsman	29	2900		104
1010	Murlidharan	Bowler	31	1800	500	102
1011	Laxman	Batsman	21	5200		101
1112	Dhoni	Wicketkeeper	24	4500		101

Country (T2)

Tid Number(3)	Cname Varchar2(15)
101	India
102	Shrilanka
103	England
104	Australia
105	South Africa

Write SQL statement with Output for the following:

1. **Create Player and country table with above mentioned schema.(eg : player & ur Rno eg player59)**

- **COUNTRY TABLE:**

```
CREATE TABLE Country(
   -> Tid INT(3) PRIMARY KEY,
   -> Cname Varchar(15)
   -> );
```

Player Table:

```
CREATE TABLE Player(
   -> Pid INT(4) PRIMARY KEY,
   -> Pname VARCHAR(20),
   -> Role VARCHAR(20),
   -> Age INT(2),
   -> Runs INT(10),
   -> Wicket INT(10),
   -> Tid INT(3),
   -> FOREIGN KEY (Tid) REFERENCES Country(Tid)
   -> );
```

2. **Insert all tuples in both the tables.**

- **COUNTRY:**

```
INSERT INTO Country VALUES (101, 'India');
INSERT INTO Country VALUES (102, 'Srilanka');
INSERT INTO Country VALUES (103, 'England');
INSERT INTO Country VALUES (104, 'Australia');
INSERT INTO Country VALUES (105, 'South Africa');
```

PLAYER:

```
INSERT INTO Player VALUES(1001, 'Sachin', 'Batsman', 25, 10000, 45, 101);
INSERT INTO Player VALUES(1002, 'Jaywardhne', 'Captain', 29, 5000, NULL, 102);
INSERT INTO Player VALUES(1003, 'Kapil', 'Bowler', 30, 2000, 470, 103);
INSERT INTO Player VALUES(1004, 'Clark', 'Batsman', 27, 3000, 120, 104);
INSERT INTO Player VALUES(1005, 'Gautam', 'Batsman', 28, 20000, NULL, 101);
INSERT INTO Player VALUES(1006, 'Rahul', 'Batsman', 22, 15000, NULL, 101);
INSERT INTO Player VALUES(1007, 'Vas', 'Bowler', 28, 3000, 480, 102);
INSERT INTO Player VALUES(1008, 'Flintop', 'Bowler', 24, 6200, 190, 103);
INSERT INTO Player VALUES(1009, 'Hussy', 'Batsman', 29, 2900, NULL, 104);
INSERT INTO Player VALUES(1010, 'Murlidharan', 'Bowler', 31, 1800, 5000, 102);
INSERT INTO Player VALUES(1011, 'Laxman', 'Batsman', 21, 5200, NULL, 101);
INSERT INTO Player VALUES(1112, 'Dhoni', 'Wicketkeeper', 24, 4500, NULL, 101);
```

3. **Add new attribute No of Matches (NOM) in player table with number (4).**
   - ALTER TABLE Player ADD NOM INT(4);

4. **Modify the NOM attribute value by 75 to all Players.**
   - UPDATE Player SET NOM = 75;

5. **List the Player names, who are not taking wickets.**
   - SELECT Pname FROM Player WHERE Wicket is NULL;

6. **Display names of players in lower and upper case.**
   - SELECT LOWER(Pname), UPPER(Pname) FROM Player;

7. **List the name, runs, and strike rate (SR) of all players. (SR is run/NOM)**
   - SELECT Pname, Runs, (Runs/NOM) as Strikerate FROM Player;

8. **List the Different roles available.**
   - SELECT DISTINCT Role FROM Player;

9. **List the player's details not belonging to the teamid 101, 102, 104.**
   - SELECT * FROM Player WHERE Tid NOT IN (101,102,104);

10. **List the Player Name and runs, whose runs are in the range of 10000 and 20000.**
    - SELECT Pname, Runs FROM Player
      -> WHERE Runs BETWEEN 10000 and 20000;

11. **List the total runs, highest runs and average runs of players teamid wise for the tid 101 and display only those rows having average runs grater than 1500 and arrange the result in Descending order of the total runs.**
    - SELECT Tid,
      -> SUM(Runs) AS TotalRuns,
      -> MAX(Runs) AS HighestRun,
      -> AVG(Runs) AS AugRun FROM Player

```
-> WHERE Tid = 101
-> GROUP BY Tid HAVING AVG(Runs) > 1500
-> ORDER BY TotalRuns DESC;
```

12. Display the list of Players in each country. Display the country information even if no player is belong to that country.

- SELECT Country.Tid, Country.Cname, Player.Pname
  -> FROM Country
  -> LEFT JOIN Player ON Country.Tid = Player.Tid;

13. Display the names of player having runs same as that of player VAS.

- SELECT Pname FROM PLAYER
  -> WHERE Runs = (SELECT Runs FROM Player WHERE Pname = 'VAS');

14. Write PL SQL program to find total player in the player table.

```
DECLARE
   total_players NUMBER;
BEGIN
   SELECT COUNT(*) INTO total_players FROM Player;
   DBMS_OUTPUT.PUT_LINE('Total Players: ' || total_players);
END;
```

15. Display the different roles present in Tid 101 and 102.

- SELECT DISTINCT Role FROM Player
  -> WHERE Tid IN (101,102);