VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

DATABASE MANAGEMENT SYSTEMS

TOPIC : Indian Premier League Team Management Model

FACULTY : Prof. Bimal Kumar Ray

GROUP MEMBERS :

1. Chaitanya Singh (19BIT0044)
2. Sourabh Karmakar (19BIT0048)
3. Madhur Patidar (19BIT0059)

REVIEW – 1

Mini World and Description

The Mini world we have selected for our project is the **Indian premier league.** The Indian Premier League (IPL) is a Twenty20 cricket league tournament held in India, contested during the months of April and May every year, where top players from all over the world take part. IPL is the most-attended cricket league in the world and ranks 6th amongst all sports leagues.

The IPL Management System allows the user to store team details and players details in a structured format. It keeps track of all the information about teams, players, injury record of players, team sponsors, venue of match, etc. These details must constantly be updated which tells us that the data must be easily accessible, and modifying it must never be an ordeal. The user will find it easy in this automated system rather than using the manual writing system. The system contains a database where all the information will be stored discretely and safely.

Data Requirements

The IPL has many teams participating, each with a **unique name.**

Each **team** has a **name**, represents a **state** (or city like Rajasthan Royals, Chennai Superkings, Mumbai Indians, Gujarat Lions, etc. all are having the names of either a state or a city), a **head coach**, a **captain**, and of course, a set of players. There may be more than one team representing the same state for a different city (like Pune Supergaints or Pune Warriors India and Mumbai Indians, both of Maharashtra).

Every team belongs to a Pool **'A'** or **'B'** and has won **number of titles**.
Each team can play only **one game per day.**

A team can either be a **host or a visitor** for a single game.
Every Team has **one or more sponsor,** which have a **unique name, period** (time period in years) and **fund** (Some Business Unicorns / corporate), but a sponsor can **invest** funds only in **one team** for certain period of years based on a contract.

A game is **played between two teams** (where one team hosts at home and the other team is a visitor) and has a **date** (such as May 11th, 2009) and a **venue** (stadium).

Each team is owned by an **owner**, and is assisted by a number of **support staffs** (like rehabilitation staff, massage staff, nutritionist staff, injury recovery staff, digital analyst staff, physical training staff, etc.).

Each owner has an **owner_id**, **name** and **profession** (e.g. Businessman for Mukesh Ambani, owner of Mumbai Indians, Actor for Shahrukh Khan, owner of Kolkata Knight Riders, etc.).
Every support staff is identified by a **staff_id**, **name** and his/her **field** (e.g. Physical Training, Injury Recovery, Massage, Rehabilitation, etc.).

Each **player belongs to only one** team.
Each player has a **name**, a **position** (Batsman, Bowler, All-rounder), **franchise_player ID** (ID provided by the particular franchise) and **skill level** (a rating given to every player i.e. a number in between 0 to 10).

A player has a log of **injury records including the description** (e.g. happened during practice session, happened during match while fielding in mid-on, happened due to a bouncer delivery during a match, etc.), **details** (hamstring injury, tear, bone injury, ligament injury, sprain, joint injury, etc.) and **date** of injury.
Each player has a log of **previous records**, containing the **player's id**, record of **total runs**, **no. of matches** played, **wickets taken** and batting/bowling **average**.

A player has to be the **captain** of each team.
Also, every player stays at a **hotel** which has details for each player, that are **name**, **room_no** and **room_type**.

Cardinality Relationship Type:

- A player can be a captain of only **one** team, and **one** team can be captained by only one player.
- A team can consist of **many** players, but each player belongs to only **one** team.
- A team can have **many** sponsors, but a sponsor can invest funds in only **one** team in a single edition/season.
- A team can have **many** number of visitors and can be a host for **many** number of teams.
- A player can have **many** number of injury records, but a particular injury record can correspond to **one** player only.
- A team can have only **one** owner, and an owner can own only **one** team.
- A team has **one** support staff and support staff can help only **one** particular team.
- A player can stay only at **one** hotel, but a single hotel can have **many** players staying in there.
- Each player has only **one** log of previous records, and each log of previous records corresponds to **one** player only.

Functional Requirements

REMOVAL OF DATA:-

1. Removal of the information of a player if the player leaves the team.
2. Removal of the information of a team if the team leaves the tournament.

3. Removal of the data of sponsor if their time period gets completed.
4. Removal of the data of players who have specific injury record.
5. Removal of the data of a support staff who assists a particular team.
6. Removal of the data of a hotel when a match is completed at a particular venue.
7. Removal of the previous records of a player if the player leaves the team.
8. Removal of the injury records of a player if the player leaves the team.

MODIFICATION OF DATA: -

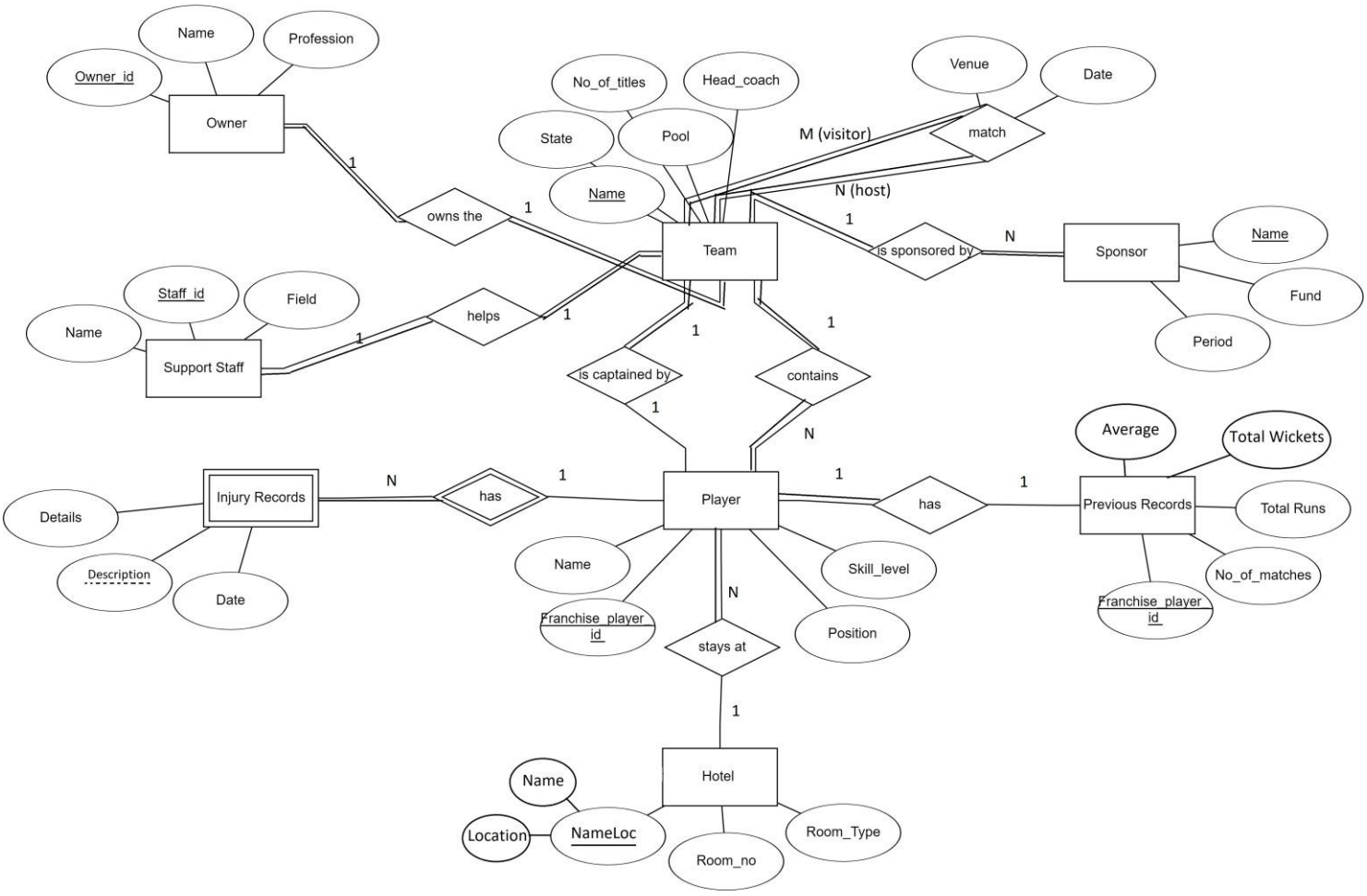
1. Update in team if new player is added.
2. Update in skill level if player changes his skill.
3. Update in match date.
4. Update in team if team name gets changed.
5. Update in player if its changes the franchise.
6. Update in hotel if venue of a match is changed.
7. Update in previous records if a new player is added.

DATA RETRIEVAL: -

1. Retrieve information of a particular player.
2. Retrieve information of a particular team.
3. Get details of injury for specific player.
4. Show number of teams participating in the tournament.

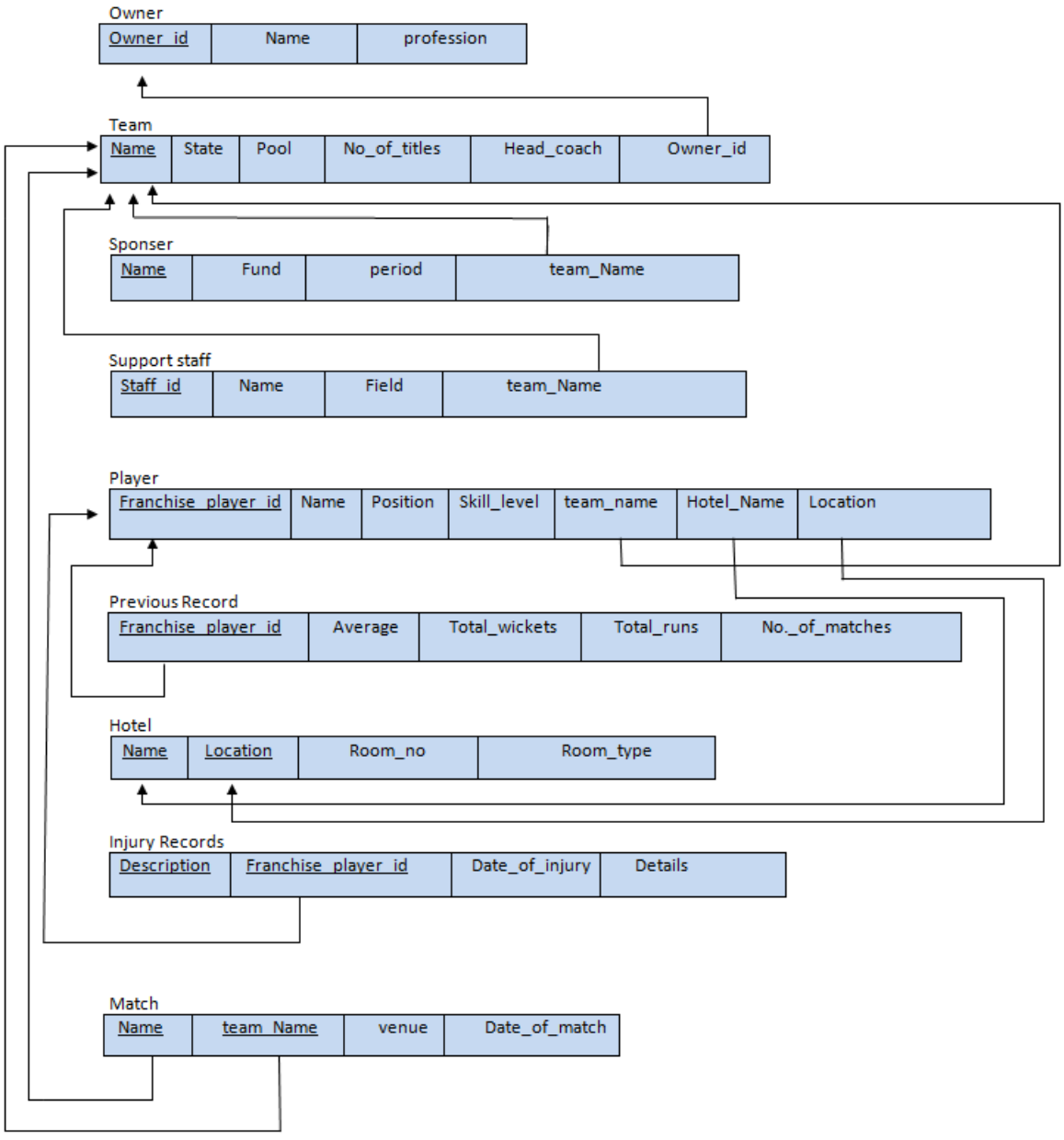
5. Show the details of a player with specific injury description.
6. Show number of sponsors for a team.
7. Show the players falling under a particular range of skill level rating.
8. Show the teams for a particular value of titles won.
9. Get the hotel details for a particular player.
10. Show previous records of a player of a particular team.
11. Show the field of a particular support staff assisting a particular team.

ER Diagram



REVIEW – 2

RELATIONAL DATABASE SCHEMA DIAGRAM



Creating Tables

Creating table owner

```
create table owner(owner_id char(5) constraint wn_pk primary key,
                   name varchar(20) not null,
                    profession varchar(20));
```

```
SQL> create table owner(owner_id char(5) constraint wn_pk primary key,
  2  name varchar(20) not null,
  3  profession varchar(20));

Table created.

SQL> desc owner
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 OWNER_ID                                  NOT NULL CHAR(5)
 NAME                                      NOT NULL VARCHAR2(20)
 PROFESSION                                         VARCHAR2(20)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'OWNER';

CONSTRAINT_NAME                C
------------------------------ -
SYS_C007665                    C
WN_PK                          P

SQL>
```

creating table hotel

```
create table hotel(name varchar(30),
                   location varchar(30),
                   room_no number(5),
                   room_type varchar(20),
constraint hotl_pk primary key(name, location));
```

```
SQL> create table hotel(name varchar(30),
  2  location varchar(30),
  3  room_no number(5),
  4  room_type varchar(20),
  5  constraint hotl_pk primary key(name, location));

Table created.

SQL> desc hotel
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                      NOT NULL VARCHAR2(30)
 LOCATION                                  NOT NULL VARCHAR2(30)
 ROOM_NO                                            NUMBER(5)
 ROOM_TYPE                                          VARCHAR2(20)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'HOTEL';

CONSTRAINT_NAME                C
------------------------------ -
HOTL_PK                        P

SQL>
```

creating table team

```
create table team(name varchar(30) constraint tm_pk primary key,
                  state varchar(15) not null,
                  no_of_titles integer not null,
                   head_coach varchar(20) unique not null,
                   owner_id constraint tm_fk1 references owner,
                   pool char(1) not null,
                   check(no_of_titles >= 0 and no_of_titles <13),
                    check(pool in ('A','B')));
```

```
SQL> create table team(name varchar(30) constraint tm_pk primary key,
  2  state varchar(15) not null,
  3  no_of_titles integer not null,
  4  head_coach varchar(20) unique not null,
  5  owner_id constraint tm_fk1 references owner,
  6  pool char(1) not null,
  7  check(no_of_titles >= 0 and no_of_titles <13),
  8  check(pool in ('A','B')));

Table created.

SQL> desc team
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                      NOT NULL VARCHAR2(30)
 STATE                                     NOT NULL VARCHAR2(15)
 NO_OF_TITLES                              NOT NULL NUMBER(38)
 HEAD_COACH                                NOT NULL VARCHAR2(20)
 OWNER_ID                                           CHAR(5)
 POOL                                      NOT NULL CHAR(1)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'TEAM';

CONSTRAINT_NAME               C
----------------------------- -
SYS_C007675                   U
TM_FK1                        R
SYS_C007668                   C
SYS_C007669                   C
SYS_C007670                   C
SYS_C007671                   C
SYS_C007672                   C
SYS_C007673                   C
TM_PK                         P

9 rows selected.

SQL>
```

creating table sponsor

```
create table sponsor(name varchar(30) constraint spn_pk primary key,
                     fund integer not null,
                     period integer not null,
                     team_name varchar (30),
                     foreign key(team_name) references team(name));
```

```
SQL> create table sponsor(name varchar(30) constraint spn_pk primary key,
  2  fund integer not null,
  3  period integer not null,
  4  team_name varchar (30),
  5  foreign key(team_name) references team(name));

Table created.

SQL>
SQL> desc sponsor
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                      NOT NULL VARCHAR2(30)
 FUND                                      NOT NULL NUMBER(38)
 PERIOD                                    NOT NULL NUMBER(38)
 TEAM_NAME                                          VARCHAR2(30)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'SPONSOR';

CONSTRAINT_NAME               C
------------------------------ -
SYS_C007677                   C
SYS_C007678                   C
SPN_PK                        P
SYS_C007680                   R

SQL>
```

creating table support_staff

```
create table support_staff(staff_id number(4) constraint stf_pk primary key,
                           name varchar(30) not null,
                           field char(15) not null,
                           team_name varchar(30),
                           foreign key(team_name) references team(name));
```

```
SQL> create table support_staff(staff_id number(4) constraint stf_pk primary key,
  2  name varchar(30) not null,
  3  field char(15) not null,
  4  team_name varchar(30),
  5  foreign key(team_name) references team(name));

Table created.

SQL> desc support_staff
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 STAFF_ID                                  NOT NULL NUMBER(4)
 NAME                                      NOT NULL VARCHAR2(30)
 FIELD                                     NOT NULL CHAR(15)
 TEAM_NAME                                          VARCHAR2(30)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'SUPPORT_STAFF';

CONSTRAINT_NAME               C
----------------------------- -
SYS_C007681                   C
SYS_C007682                   C
STF_PK                        P
SYS_C007684                   R
```

creating table player

```
create table player(franchise_player_id char(10) constraint plyr_pk primary key,
                    name varchar(40) not null,
                    position number(2),
                    hotel_name varchar(30),
                    location varchar(30),
                    skill_level number(3),
                    team_name varchar(30),
                    FOREIGN KEY (team_name) REFERENCES team (name),
                    FOREIGN KEY (hotel_name, location) REFERENCES hotel (name,
location));
```

```
SQL> create table player(franchise_player_id char(10) constraint plyr_pk primary key,
  2  name varchar(40) not null,
  3  position number(2),
  4  hotel_name varchar(30),
  5  location varchar(30),
  6  skill_level number(3),
  7  team_name varchar(30),
  8  FOREIGN KEY (team_name) REFERENCES team (name),
  9  FOREIGN KEY (hotel_name, location) REFERENCES hotel (name, location));

Table created.

SQL> desc player
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 FRANCHISE_PLAYER_ID                       NOT NULL CHAR(10)
 NAME                                      NOT NULL VARCHAR2(40)
 POSITION                                           NUMBER(2)
 HOTEL_NAME                                         VARCHAR2(30)
 LOCATION                                           VARCHAR2(30)
 SKILL_LEVEL                                        NUMBER(3)
 TEAM_NAME                                          VARCHAR2(30)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'PLAYER';

CONSTRAINT_NAME               C
------------------------------ -
SYS_C007685                   C
PLYR_PK                       P
SYS_C007687                   R
SYS_C007688                   R
```

Creating table previousRecords

```
create table previousRecords(franchise_player_id constraint prvRec_fk references
                                player,
                                average number(5,2),
                                total_wickets number(5),
                                total_runs number(10),
                                no_of_matches number(5),
                                constraint prvRec_pk primary
                                key(franchise_player_id));
```

```
SQL> create table previousRecords(franchise_player_id constraint prvRec_fk references player,
  2  average number(5,2),
  3  total_wickets number(5),
  4  total_runs number(10),
  5  no_of_matches number(5),
  6  constraint prvRec_pk primary key(franchise_player_id));

Table created.

SQL> desc previousRecords
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 FRANCHISE_PLAYER_ID                       NOT NULL CHAR(10)
 AVERAGE                                            NUMBER(5,2)
 TOTAL_WICKETS                                      NUMBER(5)
 TOTAL_RUNS                                         NUMBER(10)
 NO_OF_MATCHES                                      NUMBER(5)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'previousRecords';

no rows selected

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'PREVIOUSRECORDS';

CONSTRAINT_NAME               C
------------------------------ -
PRVREC_PK                     P
PRVREC_FK                     R
```

creating table injury_records

```
create table injury_records(franchise_player_id constraint inj_fk references player,
                            description varchar(20),
                            date_of_injury DATE,
                            details varchar(50),
                        constraint inj_pk primary key(description, franchise_player_id));
```

```
SQL> create table injury_records(franchise_player_id constraint inj_fk references player,
  2  description varchar(20),
  3  date_of_injury DATE,
  4  details varchar(50),
  5  constraint inj_pk primary key(description, franchise_player_id));

Table created.

SQL> desc injury_records
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 FRANCHISE_PLAYER_ID                       NOT NULL CHAR(10)
 DESCRIPTION                               NOT NULL VARCHAR2(20)
 DATE_OF_INJURY                                     DATE
 DETAILS                                            VARCHAR2(50)

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'INJURY_RECORDS';

CONSTRAINT_NAME               C
----------------------------- -
INJ_PK                        P
INJ_FK                        R
```

creating table Match

```
create table match(name constraint mt_fk1 references team,
                   team_name varchar(30),
                   foreign key(team_name) references team (name),
                    venue varchar(20),
                    date_of_match date,
                    constraint mt_pk primary key(name, team_name));
```

```
SQL> create table match(name constraint mt_fk1 references team,
  2  team_name varchar(30),
  3  foreign key(team_name) references team (name),
  4  venue varchar(20),
  5  date_of_match date,
  6  constraint mt_pk primary key(name, team_name));

Table created.

SQL> desc match
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                      NOT NULL VARCHAR2(30)
 TEAM_NAME                                 NOT NULL VARCHAR2(30)
 VENUE                                              VARCHAR2(20)
 DATE_OF_MATCH                                      DATE

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'MATCH';

CONSTRAINT_NAME               C
------------------------------ -
MT_PK                         P
MT_FK1                        R
SYS_C007695                   R

SQL>
```

DATA INSERTION

Table owner

```
SQL> INSERT INTO OWNER VALUES('10001', 'MUKESH AMBANI', 'BUSINESSMAN');

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO OWNER VALUES('10002', 'PREETI ZINTA', 'ACTOR');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from owner;

OWNER NAME                 PROFESSION
----- -------------------- --------------------
10001 MUKESH AMBANI        BUSINESSMAN
10002 PREETI ZINTA         ACTOR
```

Table hotel

```
SQL> INSERT INTO HOTEL VALUES('THE IVY PALACE', 'MUMBAI', 303, 'KING');

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO HOTEL VALUES('BABYLON INTERNATIONAL', 'PUNJAB', 107, 'SUITE');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from hotel;

NAME                           LOCATION                          ROOM_NO
------------------------------ ------------------------------ ----------
ROOM_TYPE
--------------------
THE IVY PALACE                 MUMBAI                               303
KING

BABYLON INTERNATIONAL          PUNJAB                               107
SUITE
```

Table Team

```
SQL> INSERT INTO TEAM VALUES('MUMBAI INDIANS', 'MAHARASHTRA', 5, 'MAHELA JAYAWARDENE', '10001', 'A');

1 row created.

SQL> commit;

Commit complete.

SQL> INSERT INTO TEAM VALUES('KINGS XI PUNJAB', 'PUNJAB', 0, 'ANIL KUMBLE', '10002', 'B');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from team;

NAME                           STATE          NO_OF_TITLES HEAD_COACH
------------------------------ -------------- ------------ --------------------
OWNER P
----- -
MUMBAI INDIANS                 MAHARASHTRA               5 MAHELA JAYAWARDENE
10001 A

KINGS XI PUNJAB                PUNJAB                    0 ANIL KUMBLE
10002 B
```

Table sponsor

```
SQL> insert into sponsor values('SAMSUNG', 80000000, 7, 'MUMBAI INDIANS');

1 row created.

SQL> commit;

Commit complete.

SQL> insert into sponsor values('EbixCash', 8900000, 5, 'KINGS XI PUNJAB');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from sponsor;

NAME                                FUND     PERIOD
------------------------------ ---------- ----------
TEAM_NAME
------------------------------
SAMSUNG                          80000000          7
MUMBAI INDIANS

EbixCash                          8900000          5
KINGS XI PUNJAB
```

Table support_staff

```
SQL> insert into support_staff values(2313, 'ROBIN SINGH', 'BATTING', 'MUMBAI INDIANS');

1 row created.

SQL> commit;

Commit complete.

SQL> insert into support_staff values(2124, 'WASIM JAFFER', 'BATTING', 'KINGS XI PUNJAB');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from support_staff;

  STAFF_ID NAME                           FIELD
---------- ------------------------------ ---------------
TEAM_NAME
------------------------------
      2313 ROBIN SINGH                    BATTING
MUMBAI INDIANS

      2124 WASIM JAFFER                   BATTING
KINGS XI PUNJAB
```

Table player

```
SQL> insert into player values('7271827181', 'ROHIT SHARMA', 1, 'THE IVY PALACE', 'MUMBAI', 9, 'MUMBAI INDIANS');

1 row created.

SQL> commit;

Commit complete.

SQL> insert into player values('6352617899', 'KL RAHUL', 1, 'BABYLON INTERNATIONAL', 'PUNJAB', 7, 'KINGS XI PUNJAB');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from player;

FRANCHISE_ NAME                                POSITION
---------- ---------------------------------- ----------
HOTEL_NAME                  LOCATION                       SKILL_LEVEL
--------------------------- ------------------------------ -----------
TEAM_NAME
------------------------------
7271827181 ROHIT SHARMA                                 1
THE IVY PALACE              MUMBAI                                9
MUMBAI INDIANS

6352617899 KL RAHUL                                     1
BABYLON INTERNATIONAL       PUNJAB                                7
KINGS XI PUNJAB
```

Table PreviousRecords

```
SQL> insert into previousRecords values('7271827181',32,15,5480,207);

1 row created.

SQL> commit;

Commit complete.

SQL> insert into previousRecords values('6352617899',47,0,2978,88);

1 row created.

SQL> commit;

Commit complete.

SQL> select * from previousRecords;

FRANCHISE_    AVERAGE TOTAL_WICKETS TOTAL_RUNS NO_OF_MATCHES
---------- ---------- ------------- ---------- -------------
7271827181         32            15       5480           207
6352617899         47             0       2978            88
```

Table injury_records

```
SQL> insert into injury_records values('7271827181','Hamstring injury',to_date('12-05-2020','dd-mm-yy'),'Mild muscle pull or strain');

1 row created.

SQL> commit;

Commit complete.

SQL> insert into injury_records values('6352617899','Appendicitis surgery',to_date('05-01-2021','dd-mm-yy'),'Pain in lower right abdomen');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from injury_records;

FRANCHISE_ DESCRIPTION         DATE_OF_I
---------- ------------------- ---------
DETAILS
--------------------------------------------------
7271827181 Hamstring injury     12-MAY-20
Mild muscle pull or strain

6352617899 Appendicitis surgery 05-JAN-21
Pain in lower right abdomen
```

Table Match

```
SQL> insert into match values('MUMBAI INDIANS','KINGS XI PUNJAB','Wankhede Stadium',to_date('20-04-2021','dd-mm-yy'));

1 row created.

SQL> commit;

Commit complete.

SQL> insert into match values('KINGS XI PUNJAB','KOLKATA KNIGHT RIDERS','Eden Gardens',to_date('23-04-2021','dd-mm-yy'));

1 row created.

SQL> commit;

Commit complete.

SQL> select * from match;

NAME                           TEAM_NAME
------------------------------ ------------------------------
VENUE               DATE_OF_M
------------------- ---------
MUMBAI INDIANS                 KINGS XI PUNJAB
Wankhede Stadium    20-APR-21

KINGS XI PUNJAB                KOLKATA KNIGHT RIDERS
Eden Gardens        23-APR-21
```

RENAMING CONSTRAINTS

TABLE TEAM

```
SQL> alter table team rename constraint SYS_C007675 to tm_un_nn;

Table altered.

SQL> alter table team rename constraint SYS_C007672 to tm_chk1;

Table altered.

SQL> alter table team rename constraint SYS_C007673 to tm_chk2;

Table altered.

SQL> alter table team rename constraint SYS_C007668 to tm_nn1;

Table altered.

SQL> alter table team rename constraint SYS_C007669 to tm_nn2;

Table altered.

SQL> alter table team rename constraint SYS_C007670 to tm_nn3;

Table altered.

SQL> alter table team rename constraint SYS_C007671 to tm_nn4;

Table altered.

SQL> commit;

Commit complete.
```

```
SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'TEAM';

CONSTRAINT_NAME                C
------------------------------ -
TM_UN_NN                       U
TM_FK1                         R
TM_NN1                         C
TM_NN2                         C
TM_NN3                         C
TM_NN4                         C
TM_CHK1                        C
TM_CHK2                        C
TM_PK                          P

9 rows selected.
```

TABLE SPONSOR

```
SQL> alter table sponsor rename constraint SYS_C007680 to spn_fk;

Table altered.

SQL> alter table sponsor rename constraint SYS_C007677 to spn_nn1;

Table altered.

SQL> alter table sponsor rename constraint SYS_C007678 to spn_nn2;

Table altered.

SQL> commit;

Commit complete.

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'SPONSOR';

CONSTRAINT_NAME               C
----------------------------- -
SPN_NN1                       C
SPN_NN2                       C
SPN_PK                        P
SPN_FK                        R
```

TABLE SUPPORT_STAFF

```
SQL> alter table support_staff rename constraint SYS_C007684 to stf_fk;

Table altered.

SQL> alter table support_staff rename constraint SYS_C007681 to stf_nn1;

Table altered.

SQL> alter table support_staff rename constraint SYS_C007682 to stf_nn2;

Table altered.

SQL> commit;

Commit complete.

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'SUPPORT_STAFF';

CONSTRAINT_NAME               C
----------------------------- -
STF_NN1                       C
STF_NN2                       C
STF_PK                        P
STF_FK                        R
```

TABLE PLAYER

```
SQL> alter table player rename constraint SYS_C007687 to plyr_fk1;

Table altered.

SQL> alter table player rename constraint SYS_C007688 to plyr_fk2;

Table altered.

SQL> alter table player rename constraint SYS_C007685 to plyr_nn1;

Table altered.

SQL> commit
  2  ;

Commit complete.

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'PLAYER';

CONSTRAINT_NAME                C
------------------------------ -
PLYR_NN1                       C
PLYR_PK                        P
PLYR_FK1                       R
PLYR_FK2                       R
```

TABLE MATCH

```
SQL> alter table match rename constraint SYS_C007695 to mt_fk2;

Table altered.

SQL> commit;

Commit complete.

SQL> select constraint_name, constraint_type from user_constraints
  2  where table_name = 'MATCH';

CONSTRAINT_NAME                C
------------------------------ -
MT_PK                          P
MT_FK1                         R
MT_FK2                         R
```

REVIEW – 3

SQL STATEMENTS FOR IMPLEMENTATION OF FUNCTIONAL REQUIREMENTS

select statements

1) Nvl and nullif

Display venue of a match and if it is null dislay ‘No data entered’ and if host name is same as visitor name then display null for the host name.

```
SQL> select nvl(venue,'NO DATA ENTERED') "Venue", nullif(name,team_name) from match;

Venue                NULLIF(NAME,TEAM_NAME)
-------------------- ------------------------------
Eden Gardens
Wankhede Stadium     MUMBAI INDIANS
Eden Gardens         KINGS XI PUNJAB
NO DATA ENTERED      KOLKATA KNIGHT RIDERS
```

2) Join query involving order by clause

Display the name of team, head coach, fund and period of a sponsor associated by that team using join query.

```
SQL> Select team.name, head_coach, fund, period from team inner join sponsor on team.name = sponsor.team_name order by team_name;

NAME                         HEAD_COACH                FUND     PERIOD
---------------------------- -------------------- ---------- ----------
KINGS XI PUNJAB              ANIL KUMBLE            8900000          5
MUMBAI INDIANS               MAHELA JAYAWARDENE    80000000          7
```

3) Uncorrelated query

Display no. of matches, average and total runs by Rohit Sharma

using uncorrelated query.

```
SQL> Select no_of_matches, average, total_runs from previousRecords where franchise_player_id
  2  in (Select franchise_player_id from player where name='ROHIT SHARMA');

NO_OF_MATCHES    AVERAGE TOTAL_RUNS
------------- ---------- ----------
          207         32       5480
```

4) Correlated query

Select the player with highest skill in every participating game along with his skill level

```
SQL> select name ,skill_level from player p where skill_level = (select max(skill_level)
  2  from player d where d.team_name=p.team_name);

NAME                                SKILL_LEVEL
----------------------------------- -----------
ROHIT SHARMA                                  9
KL RAHUL                                      7

SQL>
```

5) Set operator

Display venue in which playing teams won at least 2 titles.

```
SQL> Select venue from match minus
  2     Select venue from match, team where match.team_name = team.name and no_of_titles < 2;

VENUE
--------------------
Eden Gardens

SQL>
```

6) Group by

Display the total no.of.sponsors per team along with the team name.

```
SQL> Select count(Name) as No_of_sponsors, team_name from sponsor group by team_name;

NO_OF_SPONSORS TEAM_NAME
-------------- ------------------------------
             1 KINGS XI PUNJAB
             1 MUMBAI INDIANS

SQL>
```

7) having

Display the no of owners in those professions which have exactly 1 owner.

```
SQL> Select count(owner_id) as No_of_owners, profession from owner group by profession having count(owner_id)=1;

NO_OF_OWNERS PROFESSION
------------ --------------------
           1 BUSINESSMAN

SQL>
```

8) outer join

Perform a left outer join on players and injury_records of players based on their id and list the name, id, team name and injury description.

```
SQL> Select player.franchise_player_id, team_name , name , description from player left join injury_records on player.franchise_player_id = injury_records.franchise_player_id;

FRANCHISE_ TEAM_NAME
---------- ------------------------------
NAME                                 DESCRIPTION
------------------------------------ --------------------
6352617899 KINGS XI PUNJAB
KL RAHUL                             Appendicitis surgery

7271827181 MUMBAI INDIANS
ROHIT SHARMA                         Hamstring injury


SQL>
```

update statements

1) update in hotel for a given hotel name

```
SQL> select * from hotel;

NAME                           LOCATION                         ROOM_NO
------------------------------ ------------------------------ ----------
ROOM_TYPE
--------------------
THE IVY PALACE                 MUMBAI                              303
KING

BABYLON INTERNATIONAL          PUNJAB                              107
SUITE


SQL> UPDATE hotel
  2  SET room_no=420, room_type='SUPER DELUXE SUIT'
  3  WHERE name in (select name from hotel where name ='BABYLON INTERNATIONAL');

1 row updated.

SQL> commit;

Commit complete.

SQL> select * from hotel;

NAME                           LOCATION                         ROOM_NO
------------------------------ ------------------------------ ----------
ROOM_TYPE
--------------------
THE IVY PALACE                 MUMBAI                              303
KING

BABYLON INTERNATIONAL          PUNJAB                              420
SUPER DELUXE SUIT
```

2) update in sponsor for a particular team name

```
SQL> select * from sponsor;

NAME                              FUND     PERIOD
------------------------------ ---------- ----------
TEAM_NAME
------------------------------
SAMSUNG                         80000000          7
MUMBAI INDIANS

EbixCash                         8900000          5
KINGS XI PUNJAB


SQL> UPDATE sponsor
  2  SET name='VIDEOCON D2H', fund=10000000, period=2
  3  WHERE team_name in(select team_name from sponsor where team_name='MUMBAI INDIANS');

1 row updated.

SQL> commit;

Commit complete.

SQL> select * from sponsor;

NAME                              FUND     PERIOD
------------------------------ ---------- ----------
TEAM_NAME
------------------------------
VIDEOCON D2H                    10000000          2
MUMBAI INDIANS

EbixCash                         8900000          5
KINGS XI PUNJAB
```

3) update the details of support staff for a given team name

```
SQL> select * from support_staff;

  STAFF_ID NAME                           FIELD
---------- ------------------------------ ---------------
TEAM_NAME
------------------------------
      2313 ROBIN SINGH                    BATTING
MUMBAI INDIANS

      2124 WASIM JAFFER                   BATTING
KINGS XI PUNJAB


SQL> UPDATE support_staff
  2  SET name='SANJAY BANGAR', FIELD='BATTING'
  3  WHERE team_name in (select team_name from support_staff where team_name='MUMBAI INDIANS');

1 row updated.

SQL> commit;

Commit complete.

SQL> select * from support_staff;

  STAFF_ID NAME                           FIELD
---------- ------------------------------ ---------------
TEAM_NAME
------------------------------
      2313 SANJAY BANGAR                  BATTING
MUMBAI INDIANS

      2124 WASIM JAFFER                   BATTING
KINGS XI PUNJAB
```

4) update in match table.

```
SQL> select * from match;

NAME                           TEAM_NAME
------------------------------ ------------------------------
VENUE                DATE_OF_M
-------------------- ---------
MUMBAI INDIANS                 KINGS XI PUNJAB
Wankhede Stadium     20-APR-21

KINGS XI PUNJAB                KOLKATA KNIGHT RIDERS
Eden Gardens         20-APR-21


SQL> UPDATE match
  2  SET venue='MOHALI', date_of_match=to_date('25-04-21','dd-mm-yy')
  3  WHERE name in (select name from match where name='KINGS XI PUNJAB' and team_name='KOLKATA KNIGHT RIDERS');

1 row updated.

SQL> commit;

Commit complete.

SQL> select * from match;

NAME                           TEAM_NAME
------------------------------ ------------------------------
VENUE                DATE_OF_M
-------------------- ---------
MUMBAI INDIANS                 KINGS XI PUNJAB
Wankhede Stadium     20-APR-21

KINGS XI PUNJAB                KOLKATA KNIGHT RIDERS
MOHALI               25-APR-21
```

DELETE STATEMENTS

1)Delete the data from sponsor if name of sponsor company is SAMSUNG.

```
SQL> select * from sponsor;

NAME                              FUND     PERIOD
------------------------------ ---------- ----------
TEAM_NAME
------------------------------
EbixCash                         8900000          5
KINGS XI PUNJAB

SAMSUNG                         80000000          7
MUMBAI INDIANS


SQL> Delete from sponsor where name in(select name from sponsor where name='SAMSUNG');

1 row deleted.

SQL> commit;

Commit complete.

SQL> select * from sponsor;

NAME                              FUND     PERIOD
------------------------------ ---------- ----------
TEAM_NAME
------------------------------
EbixCash                         8900000          5
KINGS XI PUNJAB
```

2) Delete from support staff if the no_of_titles of team is zero.

```
SQL> select * from support_staff;

  STAFF_ID NAME                         FIELD
---------- ---------------------------- --------------
TEAM_NAME
------------------------------
      2313 ROBIN SINGH                  BATTING
MUMBAI INDIANS

      2124 WASIM JAFFER                 BATTING
KINGS XI PUNJAB


SQL> Delete from support_staff where team_name in(select name from team where no_of_titles =0);

1 row deleted.

SQL> select * from support_staff;

  STAFF_ID NAME                         FIELD
---------- ---------------------------- --------------
TEAM_NAME
------------------------------
      2313 ROBIN SINGH                  BATTING
MUMBAI INDIANS


SQL> commit;

Commit complete.
```

3)Delete from previous records if that skill level of player is greater than 8.

```
SQL> select * from previousRecords;

FRANCHISE_    AVERAGE TOTAL_WICKETS TOTAL_RUNS NO_OF_MATCHES
---------- ---------- ------------- ---------- -------------
7271827181         32            15       5480           207
6352617899         47             0       2978            88

SQL> Delete from previousRecords where franchise_player_id in(select franchise_player_id from player where skill_level>8);

1 row deleted.

SQL> select * from previousRecords;

FRANCHISE_    AVERAGE TOTAL_WICKETS TOTAL_RUNS NO_OF_MATCHES
---------- ---------- ------------- ---------- -------------
6352617899         47             0       2978            88

SQL> commit;

Commit complete.

SQL>
```

4)Delete from injury records if name of palyer has word RO.

```
SQL> select * from injury_records;

FRANCHISE_ DESCRIPTION         DATE_OF_I
---------- -------------------- ---------
DETAILS
--------------------------------------------------
7271827181 Hamstring injury     12-MAY-20
Mild muscle pull or strain

6352617899 Appendicitis surgery 05-JAN-21
Pain in lower right abdomen


SQL> Delete from injury_records where franchise_player_id in(select franchise_player_id from player where name like'%RO%');

1 row deleted.

SQL> select * from injury_records;

FRANCHISE_ DESCRIPTION         DATE_OF_I
---------- -------------------- ---------
DETAILS
--------------------------------------------------
6352617899 Appendicitis surgery 05-JAN-21
Pain in lower right abdomen


SQL> commit;

Commit complete.
```

PL/SQL FUNCTION INVOLVING CURSOR

1) The function to display the team name when number of titles is passed through it.

```
SQL> CREATE OR REPLACE FUNCTION GetName(x in number)
  2  RETURN varchar IS
  3  Name varchar(40);
  4  CURSOR c1
  5  IS
  6  select name from team where no_of_titles = x;
  7  counter integer := 0;
  8  BEGIN
  9    OPEN c1;
 10    FETCH c1 into Name;
 11    CLOSE c1;
 12    RETURN Name;
 13    END;
 14  /

Function created.

SQL>
```

EXECUTING FUNCTION:-

```
SQL> DECLARE
  2  getinfo varchar(40);
  3  no_of_titles number(2);
  4  BEGIN
  5  no_of_titles := 5;
  6  getinfo := GetName(no_of_titles);
  7  dbms_output.put_line(getinfo ||  ' has won ' || no_of_titles);
  8  END;
  9  /
MUMBAI INDIANS has won 5

PL/SQL procedure successfully completed.

SQL>
```

2) The function to display the runs scored by a particular player when name of the player is passed through it.

```
SQL> CREATE OR REPLACE FUNCTION Getruns(p_name in varchar)
  2  RETURN number IS
  3  runs number(5);
  4  CURSOR c1
  5  IS
  6  select total_runs from previousRecords,player where previousRecords.franchise_player_id=
  7  player.franchise_player_id and name =p_name;
  8  BEGIN
  9  OPEN c1;
 10  FETCH c1 into runs;
 11  CLOSE c1;
 12  RETURN runs;
 13  END;
 14  /

Function created.
```

EXECUTING FUNCTION:-

```
SQL> DECLARE
  2   getinfo number(5);
  3   player_name varchar(30);
  4   BEGIN
  5   player_name := 'ROHIT SHARMA';
  6   getinfo := Getruns(player_name);
  7   dbms_output.put_line(player_name ||  ' has scored total ' || getinfo || ' runs ');
  8   END;
  9   /
ROHIT SHARMA has scored total 5480 runs

PL/SQL procedure successfully completed.
```

PL/SQL PROCEDURE INVOLVING CURSOR

1) Display the details of support staff associated with the team whose whose owner's name is given.

```
SQL> CREATE OR REPLACE PROCEDURE get_staff_details(owner_name IN varchar2) as
  2  staff_name varchar2(30);
  3  staff_field varchar2(15);
  4  t_name varchar2(20);
  5  CURSOR c1
  6  IS
  7  Select support_staff.name, field, team.name from support_staff, owner, team
  8  where owner_name = owner.name and owner.owner_id = team.owner_id and
  9  team.name = support_staff.team_name;
 10  BEGIN
 11      open c1;
 12      fetch c1 into staff_name, staff_field, t_name;
 13      dbms_output.put_line('Support Staff Details: ');
 14      dbms_output.put_line('Name: ' || staff_name);
 15      dbms_output.put_line('Field: ' || staff_field);
 16      dbms_output.put_line('Team Name: ' || t_name);
 17      close c1;
 18  END;
 19  /

Procedure created.
```

Executing procedure:

```
SQL> execute get_staff_details('PREETI ZINTA');
Support Staff Details:
Name: WASIM JAFFER
Field: BATTING
Team Name: KINGS XI PUNJAB

PL/SQL procedure successfully completed.
```

2) Display the name and skill level for a player associted with a team when player name is given.

```
SQL> CREATE OR REPLACE PROCEDURE get_player_skill(player_name IN varchar2) as
  2  pl_name varchar2(40);
  3  skl_lvl number(3);
  4  CURSOR crsr IS select name, skill_level from player where player_name = name;
  5  BEGIN
  6  open crsr;
  7  fetch crsr into pl_name, skl_lvl;
  8  dbms_output.put_line('Skill of player:');
  9  dbms_output.put_line('Player Name: ' || pl_name);
 10  dbms_output.put_line('Player skill level: ' || skl_lvl);
 11  close crsr;
 12  END;
 13  /

Procedure created.
```

EXECUTING PROCEDURE:

```
SQL> execute get_player_skill('ROHIT SHARMA');
Skill of player:
Player Name: ROHIT SHARMA
Player skill level: 9

PL/SQL procedure successfully completed.
```