



DBMS PROJECT ON:
**“CRICKET DATABASE MANAGEMENT
SYSTEM”**

Section-K21RG

Submitted by

Kaicherla Sai Kumar

REG NO: 12109828

COURSE CODE: INT306

Under the guidance of

Madhuri Mam

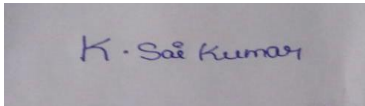
“Bachelor of technology in Computer science”

“School of Computer Science and Engineering”

Lovely professional University

Phagwara, Punjab.

DECLARATION

A rectangular box containing a handwritten signature in blue ink that reads "K. Sai Kumar".

(Signature of student)

Name of Student:

Kaicherla sai kumar

Registration no:12109828

Date:17-11-2022

ACKNOWLEDMENT

I would like to express my special thanks of gratitude to my teacher **Madhuri mam** as well as our university Lovely Professional University who gave me the golden opportunity to do this wonderful **Database Management System Project** on the topic **Cricket Management System** which also helped me in doing a lot of researches and I came to know about so many new new thing.

I am really thankful to mam.

Table of contents:

1. Introduction
2. Technology learnt
3. Profile of problem
4. Existing system
5. Problem analysis
 - Product definition
 - Feasibility study
6. Software requirement analysis
7. Design
 - Information of entities
 - Relationship between entities
 - Tables
 - ER diagram
 - Normalization
8. Implementation
 - SQL
 - PLSQL
9. Learning outcomes
10. Conclusion

Introduction:

This project is based on cricket league database system. It provides various information about the teams participating in a league. This database contains details of all teams, players, coach, owner, captain, umpire, matches etc. A cricket league is a group of cricket teams that compete each other in a cricket season. There are many formats in this game like ODI, T20I, Test, T10, The Hundred etc. Many cricket teams try to participate in the tournament, but the team captain can track out the league details through the portal. If any of the organization conducts the cricket league, the details such as stadium, umpire name and opposition team details that are stored in the database can be viewed. This project provides a team building needs and how to improve the team by creating a database of the players because we know that many players make a team. So, we keep data about every player and their performance which helps us to know how the player plays. We also create multiple variations in the team so that in any case of problems we can face and easily solve it. Many players make a team so it is important to know about every player of the team. This project provides every player performance.

Technology learnt:

Backend Software: ORACLE 11G EXPRESS EDITION

Oracle Database 11 Express Edition (Oracle Database XE) is a free version of the world's most capable relational database. With Oracle Database 11g XE, you use an intuitive, browser-based interface, to:

- Administer the database
- Create table, views, and other database object Import, export, and view table data
- Run queries and SQL script.
- Generate reports

Query languages: SQL, PLSQL

SQL:

SQL is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS). Originally based upon relational algebra and tuple relational calculus. SQL consists of a Data Definition Language (DDL), Data Manipulation Language (DML) and

Data Control Language. The scope of SQL includes data insertion, query, update and deletion of data, schema creation and modification, and data access control. SQL stores each data item in its own fields. In SQL, the fields relating to a particular person, thing or event are bundled together to form a complete unit of data, called a record (it can also be referred to as row or an occurrence). Each record is made up of a number of fields. No two fields in a record can have the same field name.

PLSQL:

PL/SQL stands for “Procedural Language extensions to the Structured Query Language”. SQL is a popular language for both querying and updating data in the relational database management systems (RDBMS). PL/SQL adds many procedural constructs to SQL language to overcome some limitations of SQL. Besides, PL/SQL provides a more comprehensive programming language solution for building mission-critical applications on Oracle Databases.

Normalization is used to minimize data redundancy and removing anomalies

Profile of the problem:

1. Removal of old data:

- If a player gets injured during the World Cup and is unable to play further, then their data needs to be removed from the database.
- If any match gets cancelled due to unforeseen circumstances, then the particular match details should be removed.
- If any team gets disqualified, then their data needs to be removed from the database.

2. Modification of data:

- After every match, the statistics of every player should be updated.
- Due to unfavourable weather conditions, a match might get delayed. Hence, the match timings need to be changed.
- After every match the existing ranks of every team should be modified.

3. Retrieval of data:

1. View details of every match like

- i. Name of the first team
- ii. Name of the second team

- iii. Umpire of the match
- iv. Winner of the match
- v. Loser of the match
- vi. man of the match

2.View details of every team like

- i. Name of the team
- ii. Captain of the team
- iii. Wicket keeper of the team
- iv. Coach of the team
- v. Owner of the team
- vi. Number of batters in the team
- vii. Number of bowlers in the team
- viii. Number of wins
- ix. Number of loses
- x. Total number of players in the team

3.View scoreboard of every match like

- i. Rank of each team
- ii. Name of the team
- iii. Total runs scored by every player
- iv. Total wickets taken by each bowler
- v. Points scored by team after every match

4.View the captain of each team

- i. Name of the captain
- ii. His experience of captaincy
- iii. Total years he served as a captain

Existing system:

The existing database contains all the data related to league on basis of it we need to normalize the tables and find the solutions to the problem. We need to normalize the tables and remove redundancy and anomalies from all the tables. The existing system has details like

Team-this table contains all the details related to team like name,id,home state,number of players,captain,coach and owner

Player-This table consists of details like player id,player name,team he represents,total runs he scored in the league ,strike rate,batting average,no of fours,no of sixes,no of centuries, no of halfcenturies,type of bowling action ,economy of the bowler and bowling average

Matches-This table consists details like venue of the match,winner of match,loser of the match,umpire details of match,date and time of match,name of teams participating in the match

Wicket keeper-This table consists of details like name of the wicket keeper,team he represent and total runs

Captain-This table consists of details like captain id,team he represents, name, experience of the player

Coach-This table consists of details like coach id,team he coached and name of the coach

Owner-This table consists of details like name of the owner ,team he owns and owner id

Umpire-This table consists of details like name of the umpire,matches he umpired,no of matches he umpired and the country he belongs to

Problem analysis:

Product definition:

It is my opinion that a new system could be beneficial to manage the data of cricket statistics of a league provided this new suggestion is feasible. Currently, the system being used for these statistics is a paper based system, in which the statistics are written down on paper and kept as documents within the club. To access these statistics, cricketers are currently having to physically go down to the club and ask to view the statistics. A photocopy of the data can then be taken.

Feasibility study:

Feasibility analysis evaluates all key factors pertinent to a project, including the economic, technological, and legal aspects and project time frame here is only one copy of the entire statistics for the cricketers at the club. This means that if the data is lost or damaged, there is no real way of recollecting the entire set of data, apart from relying on memories, and also on the possible photocopies that may have been made. Also, it is difficult for the information to be obtained by those who wish to see it. If the person who wishes to see it isn't a cricketer from that club, they may not know where the club is located in order

to view the statistics. Current cricketers also may not be willing to travel to the club if it is a long way away, especially if they just wish to glance at a few statistics, and aren't that worried about seeing all of them.

Software requirement analysis:

Firstly, there is only one copy of the entire statistics for the cricketers at the league. This means that if the data is lost or damaged, there is no real way of recollecting the entire set of data, apart from relying on memories, and also on the possible photocopies that may have been made. Also, it is difficult for the information to be obtained by those who wish to see it. If the person who wishes to see it isn't a cricketer from that league, they may not know where the clubs is located in order to view the statistics. Current cricketers also may not be willing to travel to the club if it is a long way away, especially if they just wish to glance at a few statistics, and aren't that worried about seeing all of them.

This newly suggested web based system would have many benefits when compared with the old system, and so will be much better for the cricket league to implement. The many benefits include things such as the accessibility of the system, the decreased amount of physical storage space, and the ability to sort to find statistics grade by grade.

DESIGN:

Information of entities:

In total I have ten entities and information of each entity is mentioned below:

1.Team:

Attributes:

team_id,team_rank,team_name,home_state,no_of_wins,no_of_loses,no_of_draws,net_run_rate,points,no_of_batters,no_of_bowlers.

Entity team has team_id as primary key as it is unique to determine the total team other than that name of team,rank,wins,loses,draws,nrr,points total number of players will be stored in database under team entity

2.Player:

Attributes:

Player_id,team_id,player_name,team_name,no_of_seasons,no_of_matches,no_of_runs,highest_score,strike_rate,batting_average,no_of_fours,no_of_sixes,no_of_halfcenturies,no_of_centuries,no_of_wickets,type_of_bowler,economy,bowling_average

Player entity has all the statistics of a player who participated in the league .player_id is determined as the primary key of this table all the runs, wickets, matches, highestscore, strikerate,average,fours,sixes,halfcenturies,centuries,economy are stored in the database under player entity

3.wicket_keeper:

Attributes:

Team_id,wk_name,no_of_runs

Wicket keeper entity has all the statistics of the wicket keeper.team id as a primary key references to the team_id attribute in team entity,name of the wicket keeper and number of runs are stored in the database under wicket_keeper entity

4.Captain:

Attributes:

Captain_id,captain_name,team_id,player_id,years_of_captaincy,no_of_wins

Captain entity has captain_id as primary key as it uniquely defines the captain.team_id is referred from team entity,player id,years of captaincy is about experience the player have and no of wins is the wins he bought to team as a captain all these attributes are stored in the database under captain entity

5.coach:

Attributes:

Coach_id,team_id,coach_name

Coach entity has coach_id as primary key as it uniquely defines the coach in the table,team_id is referred from team entity and name of the coach is stored in the database under coach entity

6.owner:

Attributes:

Owner_id,owner_name,team_id

Owner entity has owner_id as primary key and team_id is referred from team entity owner name is stored in the database under owner entity

7.umpire:

Attributes:

Umpire_id,umpire_name,no_of_matches,country

Umpire entity has umpire_id as primary key .no of matches is the matches he hoisted and country is the place he belongs to along with these umpire name is stored in the database under umpire entity

8.matches

Attributes:

Match_id,match_date,match_time,team_1_name,team_2_name,winner,loser,stadium,umpire_id

Matches entity has match_id as primary key.team1 name,team2 name are the team names referred from team entity,umpire_id is referred from umpire entity winner,loser,match time,date,stadium are stored in the database under matches entity

9.Umpired_by

Attributes:

Match_id,umpire_id

Umpired_by entity determined the relationship between match and umpire it is referred from umpire entity here match_id is a primary key as a match has one umpire it is referred from matches entity

10.plays:

Attributes:

team_id,match_id

Plays entity determines the relation between match and teams here none is a primary key as a match has two teams played eachother.team_id refers team entity and match_id refers matches entity

Relationship between entities:

1.team and player:

Relationship:'has'

Type:one to many

One team can have many players

2.team and wicket_keeper:

Relationship:'has'

Type:one to one

One team can have only one wicker keeper

3.team and captain:

Relationship:'leadered by'

Type:one to one

One team can have only one captain

4.team and coach, team and owner:

Relationship:'has'

Type:one to one

One team can have only one coach and owner

5.team and matches:

Relationship:'plays'

Type:one to many

One team will have many matches to play

6.matches and umpire:

Relationship:'umpired by'

Type:one to one

One match will be umpired by only one umpire

7.matches and plays:

Relationship:'have'

Type:many to many

Many matches will have many plays

TABLES:

1.Team

create table team

(

team_id number(10) primary key,

team_rank number(3),

team_name varchar(100) not null,

home_state varchar(20) not null,

no_of_wins number(3),

no_of_loses number(3),

```

no_of_draws number(3),
net_run_rate varchar(5),
points number(2),
no_of_batters number(2),
no_of_bowlers number(2)
);

insert into team values ('10111821',9,'Chennai super kings','chennai',4,10,0,0.203,8,8,4);
insert into team values ('13151719',10,'Mumbai indians','mumbai',4,10,0,0.506,8,7,5);
insert into team values ('12145435',7,'Kolkata knight riders','kolkata',6,8,0,0.416,12,8,4);
insert into team values ('08123432',2,'Rajasthan Royals','Jaipur',9,5,0,0.298,18,7,4);
insert into team values ('09162637',8,'Sunrisers Hyderabad','Hyderabad',6,8,0,0.379,12,7,4);
insert into team values ('10121639',4,'Royal Challengers Bangalore','Bangalore',8,6,0,0.253,16,8,3);
insert into team values ('19201292',5,'Delhi capitals','Delhi',7,7,0,0.204,14,7,4);
insert into team values ('12335433',6,'Punjab Kings','Chandigarh',7,7,0,0.322,14,7,4);
insert into team values ('22322243',1,'Gujarat Titans','Ahmedabad',10,4,0,0.316,20,7,4);
insert into team values ('23444563',3,'Lucknow super gaints','Lucknow',9,5,0,.251,18,8,3);

select * from team;

```

TEAM_ID	TEAM_RANK	TEAM_NAME	HOME_STATE	NO_OF_WINS	NO_OF_LOSES	NO_OF_DRAWS	NET_RUN_RATE	POINTS	NO_OF_BATTERS	NO_OF_BOWLERS
10111821	9	Chennai super kings	chennai	4	10	0	.203	8	8	4
13151719	10	Mumbai indians	mumbai	4	10	0	.506	8	7	5
12145435	7	Kolkata knight riders	kolkata	6	8	0	.416	12	8	4
8123432	2	Rajasthan Royals	Jaipur	9	5	0	.298	18	7	4
9162637	8	Sunrisers Hyderabad	Hyderabad	6	8	0	.379	12	7	4
10121639	4	Royal Challengers Bangalore	Bangalore	8	6	0	.253	16	8	3
19201292	5	Delhi capitals	Delhi	7	7	0	.204	14	7	4
22322243	1	Gujarat Titans	Ahmedabad	10	4	0	.316	20	7	4
23444563	3	Lucknow super gaints	Lucknow	9	5	0	.251	18	8	3
12335433	6	Punjab Kings	Chandigarh	7	7	0	.322	14	7	4

2.player

create table player

(

player_id varchar(30) primary key,

team_id number(10),

```

player_name varchar(20),
team_name varchar(100) not null,
no_of_seasons number(2),
number_of_matches number(3),
no_of_runs number(10),
highest_score number(10),
strike_rate number(5),
batting_average number(3),
no_of_fours number(3),
no_of_sixes number(3),
no_of_halfcenturies number(2),
no_of_centuries number(2),
no_of_wickets number(2),
type_of_bowler varchar(30),
economy number(3),
bowling_average number(10)
);

insert into player values(18,10121639,'Virat Kohli','Royal challengers
bangalore',15,223,6624,113,129.15,36.2,578,218,44,5,4,'Right arm medium',8.8,92.0);

insert into player values(45,13151719,'Rohit sharma','Mumbai
indians',15,227,5879,109,129.89,30.3,519,240,40,1,15,'Right arm offbreak',8.02,30.2);

insert into player values(7,10111821,'Ms Dhoni','Chennai super
kings',15,234,4978,84,135.2,39.2,346,229,24,0,0,'Right arm medium',0,0);

insert into player values(33,22322243,'Hardik Pandya','Gujarat
Titans',8,107,1963,91,147.59,30.2,146,110,8,0,50,'Right arm medium fast',8.75,30.7);

insert into player values(1,23444563,'KL Rahul','Lucknow super
gaints',10,109,3889,132,136.22,48.01,327,164,31,4,0,' ',0,0);

insert into player values(31,09162637,'David Warner','Sunrisers
Hyderabad',12,162,5881,126,140.69,42.01,578,215,55,4,0,'Right arm legbreak',0,0);

insert into player values(41,12145435,'Shreyas iyer','Kolkata knight
riders',8,101,2776,96,125.38,31.55,237,99,19,0,0,'Right arm legbreak',0,0);

insert into player values(17,19201292,'Rishabh pant','Delhi
capitals',8,98,2838,128,147.97,34.61,260,129,15,1,0,' ',0,0)

```

insert into player values(42,12335433,'Shikhar dhawan','Punjab Kings',15,208,6243,106,126.33,34.88,701,136,47,2,4,'Right arm offbreak',8.25,16.5);

insert into player values(9,08123432,'Sanju samson','Rajasthan Royals',9,138,3526,119,135.72,29.14,279,158,17,3,0,'',0,0);

select * from player;

PLAYER_ID	TEAM_ID	PLAYER_NAME	TEAM_NAME	NO_OF_SEASONS	NUMBER_OF_MATCHES	NO_OF_RUNS	HIGHEST_SCORE	STRIKE_RATE	BATTING_AVERAGE	NO_OF_FOURS	NO_OF_SIXES	NO_OF_HALF CENTURIES	NO_OF_CENTURIES	NO_OF_WICKETS	TYPE_OF_BOWLER	ECONOMY	BALLING_AVERAGE
18	10121839	Virat Kohli	Royal Challengers Bangalore	15	223	6824	113	129	38	578	218	44	5	4	Right arm medium	9	9
45	13151719	Rohit sharma	Mumbai indians	15	227	5879	109	130	30	519	240	40	1	15	Right arm offbreak	8	3
7	10111821	Ms Dhoni	Chennai super kings	15	234	4978	84	135	39	348	229	24	0	0	Right arm medium	0	0
33	22322243	Hardik Pandya	Gujarat Titans	8	107	1963	91	148	30	148	110	8	0	50	Right arm medium fast	9	3
1	23444583	KL Rahul	Lucknow super gaints	10	109	3889	132	138	48	327	184	31	4	0		0	0
31	9102837	David Warner	Sunrisers Hyderabad	12	182	5881	128	141	42	578	215	55	4	0	Right arm legbreak	0	0
41	12145435	Shreyas Iyer	Kolkata knight riders	8	101	2776	98	125	32	237	99	19	0	0	Right arm legbreak	0	0
17	18201282	Rishabh pant	Delhi capitals	8	98	2838	128	148	35	280	129	15	1	0		0	0
42	12335433	Shikhar dhawan	Punjab Kings	15	208	6243	106	128	35	701	136	47	2	4	Right arm offbreak	8	1
9	8123432	Sanju samson	Rajasthan Royals	9	138	3526	119	138	29	279	158	17	3	0		0	0

3.matches:

create table matches(

match_id varchar(20) primary key,

match_date varchar(20),

match_time varchar(20),

team_1_name varchar(30),

team_2_name varchar(30),

winner varchar(30),

loser varchar(30),

stadium varchar(30),

umpire_id varchar(30)

);

insert into matches values(8382,230322,'7pm','Mumbai indians','Delhi capitals','Delhi capitals','Mumbai indians','wankhade','u93');

insert into matches values(8748,240322,'3pm','Royal Challengers Bangalore','Gujarat Titans','Gujarat Titans','Royal Challengers Bangalore','bangalore','u62');

insert into matches values(8294,250322,'7pm','Punjab Kings','Lucknow super gaints','Lucknow super gaints','Punjab Kings','lucknow','u22');

insert into matches values(8029,250322,'7pm','Rajasthan Royals','Chennai super kings','Chennai super kings','Rajasthan Royals','jaipur','u93');

insert into matches values(8064,260322,'3pm','Sunrisers Hyderabad','Lucknow super gaints','Sunrisers Hyderabad','Lucknow super gaints','hyderabad','u22');

insert into matches values(8938,270322,'7pm','Rajasthan Royals','Royal Challengers Bangalore','Royal Challengers Bangalore','Rajasthan Royals','bangalore','u65');

insert into matches values(8637,280322,'3pm','Punjab Kings','Chennai super kings','Chennai super kings','Punjab Kings','chennai','u72');

insert into matches values(8039,290322,'7pm','Rajasthan Royals','Kolkata knight riders','Rajasthan Royals','Kolkata knight riders','kolkata','u62');

insert into matches values(8749,300322,'3pm','Delhi capitals','Sunrisers Hyderabad','Delhi capitals','Sunrisers Hyderabad','delhi','u65');

insert into matches values(8673,310322,'7pm','Mumbai indians','Lucknow super gaints','Lucknow super gaints','Mumbai indians','wankhade','u72');

MATCH_ID	MATCH_DATE	MATCH_TIME	TEAM_1_NAME	TEAM_2_NAME	WINNER	LOSER	STADIUM	UMPIRE_ID
8382	230322	7pm	Mumbai indians	Delhi capitals	Delhi capitals	Mumbai indians	wankhade	u93
8748	240322	3pm	Royal Challengers Bangalore	Gujarat Titans	Gujarat Titans	Royal Challengers Bangalore	bangalore	u62
8294	250322	7pm	Punjab Kings	Lucknow super gaints	Lucknow super gaints	Punjab Kings	lucknow	u22
8029	250322	7pm	Rajasthan Royals	Chennai super kings	Chennai super kings	Rajasthan Royals	jaipur	u93
8064	260322	3pm	Sunrisers Hyderabad	Lucknow super gaints	Sunrisers Hyderabad	Lucknow super gaints	hyderabad	u22
8938	270322	7pm	Rajasthan Royals	Royal Challengers Bangalore	Royal Challengers Bangalore	Rajasthan Royals	bangalore	u65
8637	280322	3pm	Punjab Kings	Chennai super kings	Chennai super kings	Punjab Kings	chennai	u72
8039	290322	7pm	Rajasthan Royals	Kolkata knight riders	Rajasthan Royals	Kolkata knight riders	kolkata	u62
8749	300322	3pm	Delhi capitals	Sunrisers Hyderabad	Delhi capitals	Sunrisers Hyderabad	delhi	u65
8673	310322	7pm	Mumbai indians	Lucknow super gaints	Lucknow super gaints	Mumbai indians	wankhade	u72

4.Wicket keeper:

create table WICKET_KEEPER(

team_id number(10) primary key,

wk_name varchar(30),

no_of_runs number(10)

);

insert into wicket_keeper values(10111821,'Ms Dhoni',4978);

insert into wicket_keeper values(23444563,'KL Rahul',3889);

insert into wicket_keeper values(19201292,'Rishabh pant',2838);

insert into wicket_keeper values(08123432,'Sanju samson',3526);

insert into wicket_keeper values(10121639,'AB de villiers',5162);

insert into wicket_keeper values(13151719,'Ishan Kishan',1870);

insert into wicket_keeper values(12145435,'Dinesh Karthik',4376);


```

insert into wicket_keeper values(12335433,'Jonny Bairstow',2938);
insert into wicket_keeper values(09162637,'Nichollos pooran',938);
insert into wicket_keeper values(22322243,'Mathew wade',1870);
select * from wicket_keeper;

```

TEAM_ID	WK_NAME	NO_OF_RUNS
10111821	Ms Dhoni	4978
23444563	KL Rahul	3889
19201292	Rishabh pant	2838
8123432	Sanju samson	3526
10121639	AB de villiers	5162
13151719	Ishan Kishan	1870
12145435	Dinesh Karthik	4376
9162637	Nichollos pooran	938
12335433	Jonny Bairstow	2938
22322243	Mathew wade	1870

5.Captain:

```

create table CAPTAIN(
captain_id varchar(30) primary key,
captain_name varchar(30),
team_id number(10),
player_id varchar(30),
year_of_captaincy number(2),
no_of_wins number(4)
);
insert into captain values('cap7','Ms Dhoni',10111821,7,15,121);
insert into captain values('cap18','Virat Kohli',10121639,18,11,64);
insert into captain values('cap45','Rohit sharma',13151719,45,12,75);
insert into captain values('cap32','Kane williamson',09162637,32,4,35);
insert into captain values('cap50','Mayank agarwal',12335433,50,1,7);
insert into captain values('cap9','Sanju samson',08123432,9,3,43);
insert into captain values('cap17','Rishabh pant',19201292,17,3,32);
insert into captain values('cap21','Dinesh Karthik','12145435',21,7,72);
insert into captain values('cap1','KL Rahul',23444563,1,6,52);
insert into captain values('cap33','Hardik Pandya',22322243,33,1,10);
select * from captain;

```

CAPTAIN_ID	CAPTAIN_NAME	TEAM_ID	PLAYER_ID	YEAR_OF_CAPTAINCY	NO_OF_WINS
cap7	Ms Dhoni	10111821	7	15	121
cap18	Virat Kohli	10121639	18	11	64
cap32	Kane williamson	9162637	32	4	35
cap50	Mayank agarwal	12335433	50	1	7
cap9	Sanju samson	8123432	9	3	43
cap21	Dinesh Karthik	12145435	21	7	72
cap1	KL Rahul	23444563	1	6	52
cap33	Hardik Pandya	22322243	33	1	10

6.Coach:

```

create table COACH(
coach_id varchar(30) primary key,
team_id number(10),
coach_name varchar(30)
);

insert into coach values ('c10',10111821,'Stephen Fleming');
insert into coach values ('c12',13151719,'Sachin Tendulkar');
insert into coach values ('c22',12145435,'Brendon McCullum');
insert into coach values ('c32',10121639,'Mike Hesson');
insert into coach values ('c28',19201292,'Ricky Ponting');
insert into coach values ('c93',09162637,' Tom Moody');
insert into coach values ('c26',12335433,'Anil Kumble');
insert into coach values ('c29',08123432,'Kumar Sangakkara');
insert into coach values ('c02',23444563,' Gautam Gambhir');
insert into coach values ('c03',22322243,'Ashish Nehra');

select * from coach;

```

COACH_ID	TEAM_ID	COACH_NAME
c10	10111821	Stephen Fleming
c12	13151719	Sachin Tendulkar
c22	12145435	Brendon McCullum
c32	10121639	Mike Hesson
c28	19201292	Ricky Ponting
c93	9162637	Tom Moody
c26	12335433	Anil Kumble
c29	8123432	Kumar Sangakkara
c02	23444563	Gautam Gambhir
c03	22322243	Ashish Nehra

7.Umpire:

```
create table UMPIRE(
umpire_id varchar(30) primary key,
umpire_name varchar(30),
no_of_matches number(4),
country varchar(20)
);
insert into umpire values('u93','anil chaudhary',34,'india');
insert into umpire values('u22','paul reifell',93,'australia');
insert into umpire values('u62','chris gaffaney',28,'england');
insert into umpire values('u72','kumar dharamsena',63,'sri lanka');
insert into umpire values('u65','nitin menon',74,'india');
select * from umpire;
```

UMPIRE_ID	UMPIRE_NAME	NO_OF_MATCHES	COUNTRY
u93	anil chaudhary	34	india
u22	paul reifell	93	australia
u62	chris gaffaney	28	england
u72	kumar dharamsena	63	sri lanka
u65	nitin menon	74	india

8.Umpired by:

```
create table UMPIRED_BY(
match_id varchar(20) primary key,
umpire_id varchar(30)
```

```
);

insert into umpired_by values(8382,'u93');
insert into umpired_by values(8748,'u62');
insert into umpired_by values(8294,'u22');
insert into umpired_by values(8029,'u93');
insert into umpired_by values(8064,'u22');
insert into umpired_by values(8938,'u72');
insert into umpired_by values(8637,'u93');
insert into umpired_by values(8039,'u62');
insert into umpired_by values(8749,'u65');
insert into umpired_by values(8673,'u72');
select * from umpired_by;
```

MATCH_ID	UMPIRE_ID
8382	u93
8748	u62
8294	u22
8029	u93
8064	u22
8938	u65
8637	u72
8039	u62
8749	u65
8673	u72

9.plays:

```
create table PLAYS(
team_id number(10),
match_id varchar(20)
);

insert into plays values(13151719,8382)
insert into plays values(19201292,8382)
insert into plays values(10121639,8748)
insert into plays values(22322243,8748)
insert into plays values(12335433,8294)
```

```

insert into plays values(23444563,8294)
insert into plays values(08123432,8029)
insert into plays values(10111821,8029)
insert into plays values(09162637,8064)
insert into plays values(23444563,8064)
insert into plays values(08123432,8938)
insert into plays values(10121639,8938)
insert into plays values(12335433,8637)
insert into plays values(10111821,8637)
insert into plays values(08123432,8039)
insert into plays values(12145435,8039)
insert into plays values(19201292,8749)
insert into plays values(09162637,8749)
insert into plays values(13151719,8673)
insert into plays values(23444563,8673)
select * from plays;

```

TEAM_ID	MATCH_ID
13151719	8382
19201292	8382
10121639	8748
22322243	8748
12335433	8294
23444563	8294
8123432	8029
10111821	8029
9162637	8064
23444563	8064
8123432	8938
10121639	8938
12335433	8637
10111821	8637
8123432	8039
12145435	8039
19201292	8749
9162637	8749
13151719	8673
23444563	8673

10.owner:

```
create table
owner(
owner_id
number(10),
owner_name
varchar(100),
team_id
number(10)
);
```

```
insert into owner values('32','JSW Group and GMR Group',19201292);
```

```
insert into owner values('20','Manoj Badale',8123432);
```

```
insert into owner values('73','Vijay Mallya',10121639);
```

```
insert into owner values('54','Shah Rukh Khan',12145435);
```

```
insert into owner values('73','Kalanithi Maran',09162637);
```

```
insert into owner values('82','N Srinivasan',10111821);
```

```
insert into owner values('27','Preity Zinta',12335433);
```

```
insert into owner values('94','Mukesh Ambani',13151719);
```

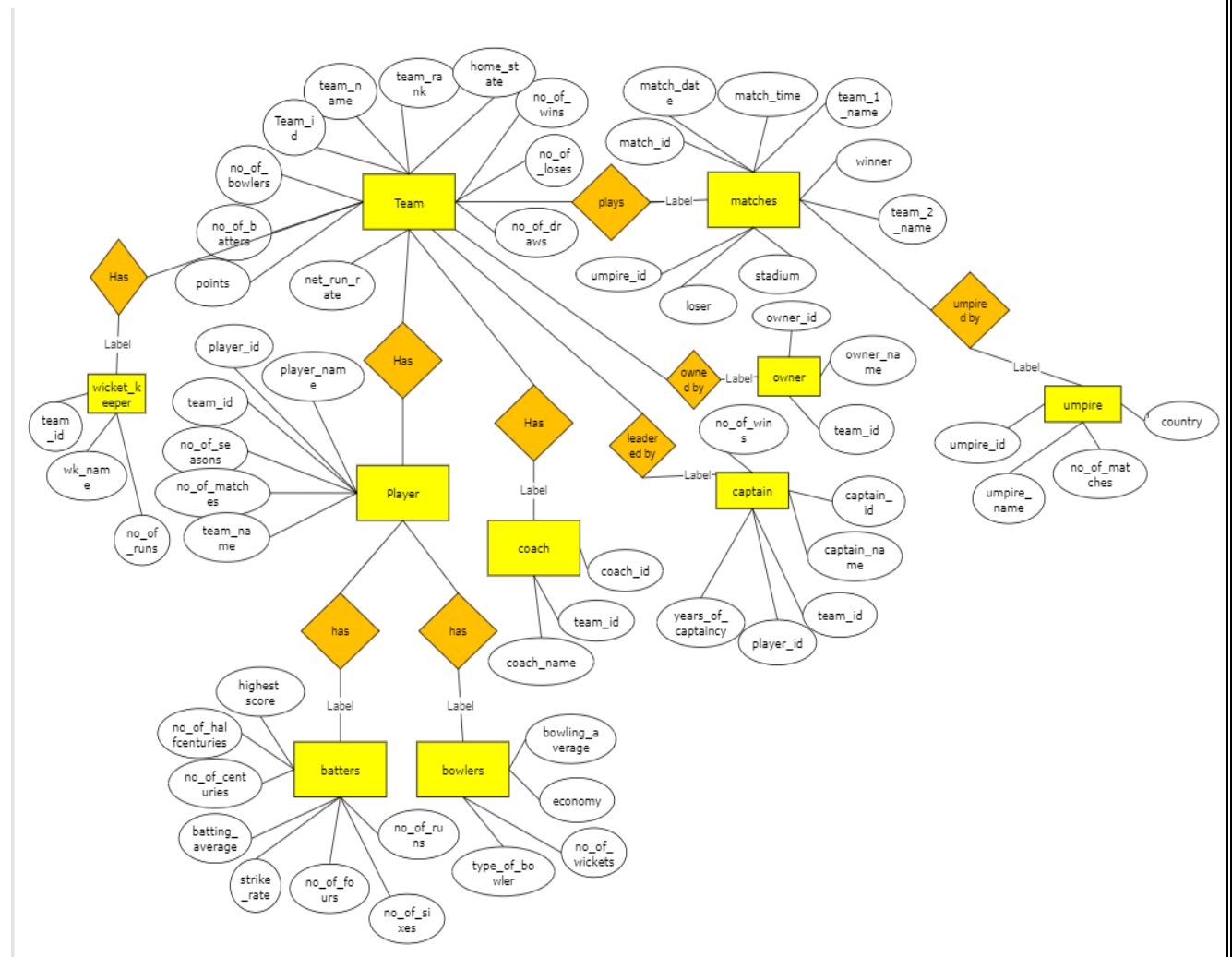
```
insert into owner values('71','Sanjeev Goenka',23444563);
```

```
insert into owner values('57','CVC Capital Partners',22322243);
```

```
select * from owner;
```

OWNER_ID	OWNER_NAME	TEAM_ID
32	JSW Group and GMR Group	19201292
20	Manoj Badale	8123432
73	Vijay Mallya	10121639
54	Shah Rukh Khan	12145435
73	Kalanithi Maran	9162637
82	N Srinivasan	10111821
27	Preity Zinta	12335433
94	Mukesh Ambani	13151719
71	Sanjeev Goenka	23444563
57	CVC Capital Partners	22322243

ER-DIAGRAM:



Normalization of Tables:

1.player table

Attributes:

Player_id,team_id,player_name,team_name,no_of_seasons,no_of_matches,no_of_runs,highest_score,strike_rate,batting_average,no_of_fours,no_of_sixes,no_of_halfcenturies,no_of_centuries,no_of_wickets,type_of_bowler,economy,bowling_average

Functional dependencies:

Player_id→player_name,team_name,no_of_seasons,no_of_matches,no_of_runs,highest_score,strike_rate,batting_average,no_of_fours,no_of_sixes,no_of_halfcenturies,no_of_centuries,no_of_wickets,type_of_bowler,economy,bowling_average

Candidate keys → player_id, team_id

The table is in first normal form as no attribute have multi values all are atomic in nature

Primary key → player_id

This table is in it's highest normal form i.e BCNF as player_id(LHS) is the super key of the table

2.team table

Attributes:

team_id, team_rank, team_name, home_state, no_of_wins, no_of_loses, no_of_draws, net_run_rate, points, no_of_batters, no_of_bowlers.

Functional dependency:

Team_id → team_rank, team_name, home_state, no_of_wins, no_of_loses, no_of_draws, net_run_rate, points, no_of_batters, no_of_bowlers.

Candidate key → team_rank

Foreignkey → team_id

This table is in it's highest normal form i.e BCNF as team_id(LHS) is the super key of the table

3.matches

Attributes:

Match_id, match_date, match_time, team_1_name, team_2_name, winner, loser, stadium, umpire_id

Functional dependency:

Match_id → match_date, match_time, team_1_name, team_2_name, winner, loser, stadium, umpire_id

Match_date → team_1_name, team_2_name, winner, loser, stadium, umpire_id

Primary key → match_id

Candidate key → match_date

Foreign key → umpire_id

This table is in it's highest normal form i.e BCNF as match_id is the super key of the table.

Tables wicket_keeper,umpire,captain,coach,umpired_by,owner,plays are all normalized as they have the foreign keys referenced to above 3 tables

Implementation:

SQL:

Update :

Suppose a player has scored a century(105 runs) in the match happened today so update the details of the player in the table player

Query:

update player set

no_of_runs=no_of_runs+105,no_of_centuries=no_of_centuries+1 where player_id=18;

select * from player

PLAYER_ID	TEAM_ID	PLAYER_NAME	TEAM_NAME	NO_OF_SEASONS	NUMBER_OF_MATCHES	NO_OF_RUNS	HIGHEST_SCORE	STRIKE_RATE	BATTING_AVERAGE	NO_OF_FOURS	NO_OF_SIXES	NO_OF_HALFCENTURIES	NO_OF_CENTURIES	NO_OF_WICKETS	TYPE_OF_BOWLER	ECONOMY	BOWLING_AVERAGE
18	10121839	Virat Kohli	Royal Challengers Bangalore	15	223	6834	113	129	38	578	218	44	8	4	Right arm medium	9	92
45	13151719	Rohit sharma	Mumbai indians	15	227	5879	109	130	30	519	240	40	1	15	Right arm offbreak	8	30
7	10111821	Ms Dhoni	Chennai super kings	15	234	4978	84	135	39	348	229	24	0	0	Right arm medium	0	0
33	22322243	Hardik Pandya	Gujarat Titans	8	107	1983	91	148	30	148	110	8	0	50	Right arm medium fast	9	31
1	23444680	KL Rahul	Lucknow super giants	10	109	3889	132	138	48	327	184	31	4	0		0	0
31	9182837	David Warner	Sunrisers Hyderabad	12	182	5881	128	141	42	578	215	55	4	0	Right arm legbreak	0	0
41	12148435	Shreyas iyer	Kolkata knight riders	8	101	2778	98	125	32	237	99	19	0	0	Right arm legbreak	0	0
17	19201292	Rishabh pant	Delhi capitals	8	98	2838	128	148	35	280	129	15	1	0		0	0
42	12335433	Shikhar dhawan	Punjab Kings	15	208	6243	108	128	35	701	138	47	2	4	Right arm offbreak	8	17
9	8123432	Sanju samson	Rajasthan Royals	9	138	3528	119	138	29	279	158	17	3	0		0	0

And:

Selecting players whose runs are over 5000 and matches greater than 150

Query:

select * from player where no_of_runs>5000 and number_of_matches>150;

These are the highest runs scored by a batter in the league

PLAYER_ID	TEAM_ID	PLAYER_NAME	TEAM_NAME	NO_OF_SEASONS	NUMBER_OF_MATCHES	NO_OF_RUNS	HIGHEST_SCORE	STRIKE_RATE	BATTING_AVERAGE	NO_OF_FOURS	NO_OF_SIXES	NO_OF_HALFCENTURIES	NO_OF_CENTURIES	NO_OF_WICKETS	TYPE_OF_BOWLER	ECONOMY	BOWLING_AVERAGE
18	10121839	Virat Kohli	Royal Challengers Bangalore	15	223	6834	113	129	38	578	218	44	8	4	Right arm medium	9	92
45	13151719	Rohit sharma	Mumbai indians	15	227	5879	109	130	30	519	240	40	1	15	Right arm offbreak	8	30
31	9182837	David Warner	Sunrisers Hyderabad	12	182	5881	128	141	42	578	215	55	4	0	Right arm legbreak	0	0
42	12335433	Shikhar dhawan	Punjab Kings	15	208	6243	108	128	35	701	138	47	2	4	Right arm offbreak	8	17

Order by:

Creating the points table of the league

Query:

```
select * from team order by points desc;
```

From this query we can determine the points table of the league

TEAM_ID	TEAM_RANK	TEAM_NAME	HOME_STATE	NO_OF_WINS	NO_OF_LOSES	NO_OF_DRAWS	NET_RUN_RATE	POINTS	NO_OF_BATTERS	NO_OF_BOWLERS	TOTAL_MATCHES
22322243	1	Gujarat Titans	Ahmedabad	10	4	0	.316	20	7	4	14
23444563	3	Lucknow super gaints	Lucknow	9	5	0	.251	18	8	3	14
8123432	2	Rajasthan Royals	Jaipur	9	5	0	.298	18	7	4	14
10121639	4	Royal Challengers Bangalore	Bangalore	8	6	0	.253	16	8	3	14
19201292	5	Delhi capitals	Delhi	7	7	0	.204	14	7	4	14
12335433	6	Punjab Kings	Chandigarh	7	7	0	.322	14	7	4	14
12145435	7	Kolkata knight riders	kolkata	6	8	0	.416	12	8	4	14
9162637	8	Sunrisers Hyderabad	Hyderabad	6	8	0	.379	12	7	4	14
10111821	9	Chennai super kings	chennai	4	10	0	.203	8	8	4	14
13151719	10	Mumbai indians	mumbai	4	10	0	.506	8	7	5	14

Set operations:

1.view of the umpire who have not umpired matches in bangalore

Query:

```
select umpire_name from umpire minus select umpire_name from umpire
where umpire_id in(select umpire_id from matches where stadium='bangalore');
```

UMPIRE_NAME
anil chaudhary
kumar dharamsena
paul reifell

2. View of name of coach who has coached a player with total_runs greater than 500;

Query:

```
select distinct coach_name from coach where team_id in(select team_id from
player where no_of_runs>5000 );
```

COACH_NAME
Sachin Tendulkar
Mike Hesson
Tom Moody
Anil Kumble

3. Display name of wicket keeper who is also the captain of his team.

Query:

```
select wk_name from WICKET_KEEPER where team_id in(select team_id
from captain where captain_name=wk_name);
```

WK_NAME
Ms Dhoni
Sanju samson
Dinesh Karthik
KL Rahul

Delete:

if some match is cancelled due to some unforeseen situation we need to remove the match details from the database

match between Punjab kings and lucknow super gaints is cancelled

query:

```
delete from matches where match_id=8294;
```

MATCH_ID	MATCH_DATE	MATCH_TIME	TEAM_1_NAME	TEAM_2_NAME	WINNER	LOSER	STADIUM	UMPIRE_ID
8382	230322	7pm	Mumbai indians	Delhi capitals	Delhi capitals	Mumbai indians	wankhade	u93
8748	240322	3pm	Royal Challengers Bangalore	Gujarat Titans	Gujarat Titans	Royal Challengers Bangalore	bangalore	u62
8029	250322	7pm	Rajasthan Royals	Chennai super kings	Chennai super kings	Rajasthan Royals	jaipur	u93
8064	260322	3pm	Sunrisers Hyderabad	Lucknow super gaints	Sunrisers Hyderabad	Lucknow super gaints	hyderabad	u22
8938	270322	7pm	Rajasthan Royals	Royal Challengers Bangalore	Royal Challengers Bangalore	Rajasthan Royals	bangalore	u65
8637	280322	3pm	Punjab Kings	Chennai super kings	Chennai super kings	Punjab Kings	chennai	u72
8039	290322	7pm	Rajasthan Royals	Kolkata knight riders	Rajasthan Royals	Kolkata knight riders	kolkata	u62
8749	300322	3pm	Delhi capitals	Sunrisers Hyderabad	Delhi capitals	Sunrisers Hyderabad	delhi	u65
8673	310322	7pm	Mumbai indians	Lucknow super gaints	Lucknow super gaints	Mumbai indians	wankhade	u72

In this table match with id=8294 details are deleted

PLSQL:

Functions:

if we need to book tickets for the upcoming match and need to display the stadium name on entering match

query:

```

declare
matchid matches.match_id%type:=:matchid;
std matches.stadium%type;
function get_stdname(matchid in matches.match_id%type,std out matches.stadium%type)
return varchar is
begin
select stadium into std from matches where match_id=matchid;
return std;
end;
begin
dbms_output.put_line('stadium: '||get_stdname(matchid,std));
end;

match_id=8382

output:
stadium: wankade

```

Exception handling:

When we need to have the details of a particular player by mentioning his player id

Query:

```

declare
playername player.player_name%type;
playerid player.player_id%type:=:playerid;
noofseasons player.no_of_seasons%type;
matches player.number_of_matches%type;
runs player.no_of_runs%type;
highestscore player.highest_score%type;
begin
select player_name,no_of_seasons,number_of_matches,no_of_runs,highest_score into
playerid,noofseasons,matches,runs,highestscore from player where playerid=player_id;
dbms_output.put_line('playerid='||playerid);
dbms_output.put_line('no of seasons='||noofseasons);
dbms_output.put_line('matches='||matches);
dbms_output.put_line('runs='||runs);
dbms_output.put_line('high score='||highestscore);
exception

```

```

when no_data_found then

dbms_output.put_line('enter the correct player name');

end;

player_id=18

output:

playerid=Virat Kohli
no of seasons=15
matches=223
runs=6834
high score=113
if entered wrong id

player_id=84

output:

enter the correct player name

```

Triggers:

1.when a team is eliminated from the league due to poor performance we need to update the table

Query:

```

create table elimination

(
team_id varchar(10) primary key,
home_state varchar(15),
rank number(2),
team_name varchar(15),
no_of_loses number(2)
);

create or replace trigger trig2
after delete on team
referencing new as new old as old
for each row
begin
insert into elimination values (:OLD.team_id ,:OLD.home_state,:OLD.team_rank
,:OLD.team_name,:OLD.no_of_loses);

end;

delete from team where team_id=:team_id;

```

select * from elimination

TEAM_ID	HOME_STATE	RANK	TEAM_NAME	NO_OF_LOSES
12335433	Chandigarh	6	Punjab Kings	7

2.due to some unforeseen practice a team was banned for an year.after an year the team came back to board of cricket in india in order to change the team_id because of some reasons

Query:

Create or replace trigger updateid after update on team for each row

begin

update player set team_id=:new.team_id where

team_id=:old.team_id;

update coach set team_id=:new.team_id

where team_id=:old.team_id;

update captain set team_id=:new.team_id where

team_id=:old.team_id;

update plays set team_id=:new.team_id where

team_id=:old.team_id;

update wicket_keeper set team_id=:new.team_id where

team_id=:old.team_id;

end;

after creating the trigger we now update the team_id from the whole database

update team set team_id='1234' where team_id='12145435';

select * from coach;

COACH_ID	TEAM_ID	COACH_NAME	CAPTAIN_ID	CAPTAIN_NAME	TEAM_ID	PLAYER_ID	YEAR_OF_CAPTAINCY	NO_OF_WINS
c10	10111821	Stephen Fleming	cap7	Ms Dhoni	10111821	7	15	121
c12	13151719	Sachin Tendulkar	cap18	Virat Kohli	10121639	18	11	64
c22	1234	Brendon McCullum	cap32	Kane williamson	9162637	32	4	35
c32	10121639	Mike Hesson	cap50	Mayank agarwal	12335433	50	1	7
c28	19201292	Ricky Ponting	cap9	Sanju samson	8123432	9	3	43
c93	9162637	Tom Moody	cap21	Dinesh Karthik	1234	21	7	72
c26	12335433	Anil Kumble	cap1	KL Rahul	23444563	1	6	52
c29	8123432	Kumar Sangakkara	cap33	Hardik Pandya	22322243	33	1	10
c02	23444563	Gautam Gambhir						
c03	22322243	Ashish Nehra						

We can see that team_id is updated in both coach and captain table

Cursors:

1.If we need to predict the players performance for the upcoming match by fetching the players details using procedures

Query:

declare

cursor details is

select * from player;

det details%rowtype;

team_id number(10);

begin

open details;

loop

fetch details into det;

exit when details% notfound;

team_id:=:team_id;

if(det.team_id=team_id) then

dbms_output.put_line('player name: '||det.player_name);

dbms_output.put_line('no of total runs: '||det.no_of_runs);

dbms_output.put_line('Batting average: '||det.batting_average);

dbms_output.put_line('economy: '||det.economy);

else

```

dbms_output.put_line('Team id did not match');

end if;

end loop;

end;

team_id= 10111821

output:

player name:Ms Dhoni
no of total runs:4978
Batting average: 39
economy: 0

```

2.When we want to print the statistics of all players

Query:

```

declare
playerid player.player_id%type;
playername player.player_name%type;
runs player.no_of_runs%type;
cursor c_player is select player_id,player_name,no_of_runs from player;
begin
open c_player;
loop fetch c_player into playerid,playername,runs;
exit when c_player%notfound;
dbms_output.put_line(playerid||' '||playername||' '||runs);
end loop;
close c_player;
end;

output:
18 Virat Kohli 6834
45 Rohit sharma 5879
7 Ms Dhoni 4978
33 Hardik Pandya 1963
1 KL Rahul 3889
31 David Warner 5881
41 Shreyas iyer 2776
17 Rishabh pant 2838
42 Shikhar dhawan 6243
9 Sanju samson 3526

```

Learning outcomes:

- I have learnt to create tables on this database
- Editing,updating and deleting the records in the table which results in proper management of database
- I have understood the relational schema of the database

- Managing the information of players
- I have learnt to remove errors from plsql
- I have learnt to monitor all the statistics of all players

Conclusion:

Cricket database management helps user to view all the information in the database by using sql and plsql. these query languages helped a lot to find the details in every aspect of database. By allowing users to simply access their data at their own homes presents a very attractive solution to the problem that we currently face in the paper based system. This system would reduce the chances of data loss through physical damage, as the data could be easily saved onto a disk, and the data itself can be easily organized as there is no risk of losing a sheet of paper anymore as it is all on a computer. The data requirements are easily understandable using an ER model as it utilizes clear diagrams also it can be converted into relational schema. plsql has high productivity to programmers as it can query, transform and also update the data in the database. Also the database I created is portable that anybody can use this dataset