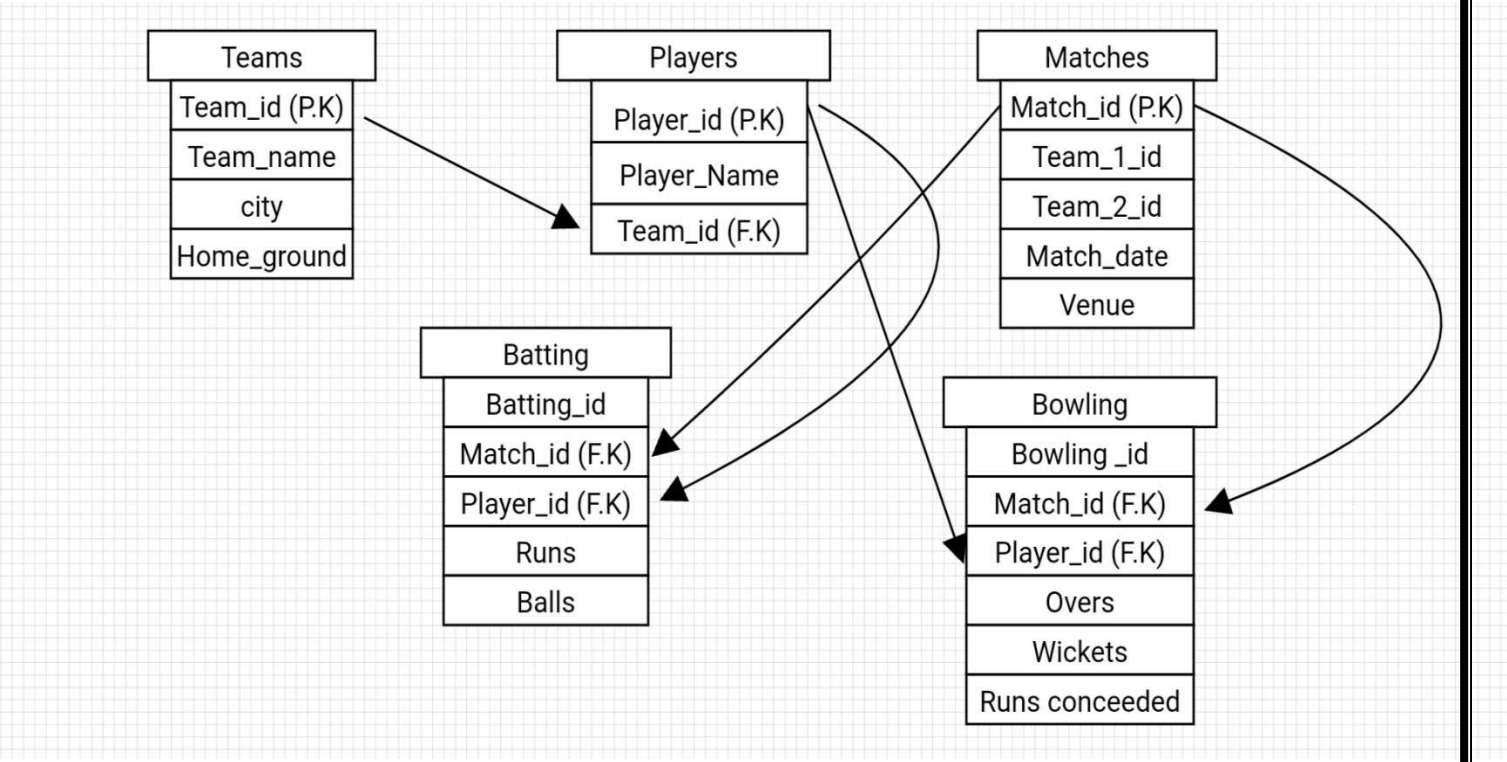
SQL PROJECT

We all know what SQL does.

- can execute queries against a database.
- can retrieve data from a database, can insert records in a database.
- can update records in a database, can delete records from a database.
- can create new databases.
- can create new tables in a database.
- create stored procedures in a database.
- can create views in a database.
- can set permissions on tables, procedures, and views.

To brush up through features of SQL let's take a scenario of IPL. The scenario is divided in 5 tables that are joined through constraints. (P.K- Primary Key & F.K - Foreign Key (the other constraints are Unique, Not Null, Default, Check).



We can mention the other constraints (Unique, Not Null, Default, Check) as per the requirements. The datatypes of the columns of the respective tables can be also mentioned according to the requirements.

Let's see how the queries are being written according to the requirements.

First of all, we start with creating a database and later on using the database required tables are created.

```
create database project;
use project;  -- create the database project and use it to create the tables.
```

Moving forward we start creating the required tables, mentioning the datatypes of the respective columns and then the insertion of the data in the tables containing respective columns takes place. In this way creation of tables and insertion of data is completed.

- Table name - teams

```
-- Tables present here -- teams, players, matches, batting, bowling

create table teams(team_id int primary key ,team_name text,city text,home_ground text);
insert into teams values(1,'RCB','banglore','chinaswamy'),(2,'MI','mumbai','wankhde'),(3,'KKR','kolkata','erden gardens'),
(4,'CSK','chennai','chepauk'),(5,'PBKS','mohali','PCA-stadium'),(6,'SRH','hyderabad','rajiv-gandhi stadium'),
(7,'DC','delhi','ferozshah-kotla'),(8,'LSG','lucknow','ekana sports stadium'),(9,'GT','gujrat','narendra modi stadium'),
(10,'RR','rajasthan','sawai mansingh stadium');
```

- Table name – players

```
create table players(player_id int primary key,player_name text,team_id int,foreign key(team_id) references teams(team_id));
insert into players values(1,'MS Dhoni',4),(2,'Rohit Sharma',2),(3,'Virat Kohli',1),(4,'shikhar dhawan',5),(5,'axar patel',7),
(6,'shardul thakur',4),(7,'shardul thakur',5),(8,'shardul thakur',7),(9,'shikhar dhawan',6),(10,'ravindra jadeja',4),
(11,'andre russel',3),(12,'ben stokes',4),(13,'andre russel',7),(14,'eoin morgan',3),(15,'anjinkya rahane',4),(16,'SKY',2),
(17,'tilak verma',2),(18,'quinton dekock',2),(19,'Jasprit Bumrah',2),(20,'Shreyas Iyer',3),(21,'Faf du Plessis',1),(22,'Faf du Plessis',4),
(23,'Rishabh Pant',7),(24,'Kane Williamson',9),(25,'Jos Buttler',10),(26,'kl rahul',8),(27,'mohsin khan',8),(28,'Hardik Pandya',9),
(29,'rashid khan',9),(30,'shubman gill',9),(31,'Mohammed Shami',9),(32,'quinton dekock',8),(33,'Ravi Bishnoi',8),(34,'Jason Holder',8),
(35,'Marcus Stoinis',8),(36,'Vijay Shankar',9),(37,'Kagiso Rabada',5),(38,'Trent Boult',10),(39,'Bhuvneshwar Kumar',6),(40,'Anrich Nortje',7),
(41,'jofra archer',2),(42,'dinesh karthik',1),(43,'Pat Cummins',3),(44,'Daniel Sams',2),(45,'Subhranshu Senapati',4),(46,'Dwayne Bravo',4),
(47,'Dwayne Bravo',2),(48,'Mohammed Siraj',1),(49,'Wanindu Hasaranga',1),(50,'R Ashwin',10),(51,'shardul thakur',3);
```

➢ Table name - matches

```
create table matches(match_id int primary key,team_1_id int,team_2_id int,match_date datetime,venue text,winner_team_id int);
insert into matches values(1,1,2,'2023-05-27 17:35:25','chinaswamy',1),(2,1,3,'2023-05-26 08:15:25','erden gardens',3),
(3,1,4,'2023-05-28 07:04:29','chepauk',1),(4,1,5,'2023-05-30 17:20:29','PCA-stadium',5),(5,1,6,'2023-06-01 00:35:24','rajiv-gandhi stadium',1),
(6,1,7,'2023-06-03 02:55:27','ferozshah-kotla',7),(7,1,8,'2023-06-05 12:40:03','ekana sports stadium',1),
(8,1,9,'2023-06-02 03:08:12','narendra modi stadium',9),(9,1,10,'2023-06-04 04:40:36','sawai mansingh stadium',1),
(10,2,3,'2023-06-06 22:30:33','wankhde',2),(11,2,4,'2023-06-07 02:07:13','chepauk',4),(12,2,5,'2023-06-09 05:30:23','PCA-stadium',2),
(13,2,6,'2023-06-11 21:05:43','rajiv-gandhi stadium',6),(14,2,7,'2023-06-08 01:05:33','ferozshah-kotla',2),
(15,2,8,'2023-06-10 09:23:12','ekana sports stadium',8),(16,2,9,'2023-06-12 10:30:53','narendra modi stadium',2),
(17,2,10,'2023-06-13 04:02:31','sawai mansingh stadium',10),(18,3,4,'2023-06-14 05:52:34','erden gardens',3),
(19,3,5,'2023-06-15 01:05:32','PCA-stadium',5),(20,3,6,'2023-06-16 03:08:42','rajiv-gandhi stadium',3),
(21,3,7,'2023-06-17 09:35:13','ferozshah-kotla',7),(22,3,8,'2023-06-18 08:15:53','ekana sports stadium',3),
(23,3,9,'2023-06-19 07:53:53','narendra modi stadium',9),(24,3,10,'2023-06-20 09:51:51','sawai mansingh stadium',3),
(25,4,5,'2023-06-21 02:05:52','chepauk',4),(26,4,6,'2023-06-22 01:04:31','rajiv-gandhi stadium',6),
(27,4,7,'2023-06-23 07:25:22','ferozshah-kotla',4),(28,4,8,'2023-06-24 06:35:24','ekana sports stadium',8),
(29,4,9,'2023-06-25 17:45:12','narendra modi stadium',4),(30,4,10,'2023-06-26 22:35:09','sawai mansingh stadium',10),
(31,5,6,'2023-06-27 04:20:39','PCA-stadium',5),(32,5,7,'2023-06-28 07:02:23','ferozshah-kotla',7),
(33,5,8,'2023-06-29 05:45:27','ekana sports stadium',5),(34,5,9,'2023-06-30 06:45:28','narendra modi stadium',9),
(35,5,10,'2023-07-01 21:35:10','sawai mansingh stadium',5),(36,6,7,'2023-07-02 07:03:25','rajiv-gandhi stadium',6),
(37,6,8,'2023-07-03 06:44:28','ekana sports stadium',8),(38,6,9,'2023-07-04 02:44:02','narendra modi stadium',6),
(39,6,10,'2023-07-05 20:34:11','sawai mansingh stadium',10),(40,7,8,'2023-07-06 06:45:20','rajiv-gandhi stadium',7),
(41,7,9,'2023-07-07 02:44:02','narendra modi stadium',9),(42,7,10,'2023-07-05 20:34:11','sawai mansingh stadium',7),
(43,8,9,'2023-07-08 02:45:02','ekana sports stadium',8),(44,8,10,'2023-07-09 20:30:15','sawai mansingh stadium',10),
(45,9,10,'2023-07-10 2:13:14','narendra modi stadium',9);
```

➢ Table name - batting

```
create table batting(batting_id int,match_id int,foreign key(match_id) references matches(match_id),
player_id int,foreign key(player_id) references players(player_id),runs int,balls int);
insert into batting values(1, 1, 1, 100, 35),(2, 2, 2, 115, 48),(3, 3, 3, 46, 25),(4, 4, 4, 46, 30),(5, 5, 5, 78, 22),(6, 6, 6, 33, 28),
(7, 7, 7, 72, 52),(8, 8, 8, 15, 18),(9, 9, 9, 60, 45),(10, 10, 10, 28, 25),(11, 11, 11, 88, 62),(12, 12, 12, 52, 41),
(13, 13, 13, 40, 35),(14, 14, 14, 25, 30),(15, 15, 15, 70, 55),(16, 16, 16, 12, 18),(17, 17, 17, 50, 40),(18, 18, 18, 95, 70),
(19, 19, 19, 20, 25),(20, 20, 20, 60, 50),(21, 21, 21, 35, 30),(22, 22, 22, 75, 60),(23, 23, 23, 48, 40),(24, 24, 24, 82, 65),
(25, 25, 25, 30, 32),(26, 26, 26, 55, 48),(27, 27, 27, 42, 38),(28, 28, 28, 68, 55),(29, 29, 29, 25, 20),(30, 30, 30, 90, 72),
(31, 31, 31, 15, 20),(32, 33, 32, 50, 42),(33, 33, 33, 75, 60),(34, 34, 34, 18, 25),(35, 35, 35, 62, 50),(36, 36, 36, 30, 28),
(37, 37, 37, 80, 65),(38, 38, 38, 22, 30),(39, 39, 39, 45, 40),(40, 40, 40, 58, 50),(41, 41, 41, 30, 25),(42, 42, 42, 42, 38),
(43, 43, 43, 90, 75),(44, 44, 44, 12, 18),(45, 45, 45, 55, 48),(47, 1, 26, 208, 35),(48, 2, 28, 215, 48),(49, 3, 17, null, 2),
(50, 4, 20, null, 3),(51, 5, 23, 309, 200),(52, 6, 25, 333, 208),(51, 10, 9, 60, 45),(52, 11, 9, 60, 45),(53, 12, 9, 60, 45),
(54, 16, 9, 60, 45),(55, 18, 9, 60, 45),(56,10, 9, 60, 45),(57, 16, 9, 61, 45),(58, 19, 9, 60, 45),(59, 21, 9, 60, 45),(60, 22, 9, 60, 45);
```

➢ Table name - bowling

```
create table bowling(bowling_id int,match_id int,foreign key(match_id) references matches(match_id),
player_id int,foreign key(player_id) references players(player_id),overs int,wickets int,runs_conceded int);
insert into bowling values(1,4,3,10,3,109),(2,2,3,10,4,123),(3,6,7,12,2,67),(4,8,9,14,3,68),(5,2,4,15,2,167),(6,5,6,18,4,69),
(7,2,4,12,5,60),(8,23,50,13,2,68),(9,13,49,10,3,67),(10,40,39,11,2,86),(11,2,2,14,2,67),(12,38,38,13,2,120),(13,40,40,20,2,141),
(14,34,34,6,2,70),(15,23,23,7,2,67),(16,41,41,10,4,62),(17,2,7,18,3,61),(18,2,2,7,2,56),(19,2,28,7,2,50),(20,34,39,16,6,93);
```

Now again we will have requirements from the data that are stored in the table. The requirements can be put forward in the form of questions. As we have taken the scenario of IPL the requirements will be there based on that and to fetch the data from the table according to the requirements, we will write queries.

1) List all the teams and their home grounds.

```
select team_name,home_ground from teams; -- Q1
```

2) List all the players along with their respective teams.

```
select p.player_name,t.team_name
from players p inner join teams t
on p.team_id = t.team_id;        -- Q2
```

3) List all the matches with their venues and dates.

```
select match_id,venue,match_date from matches; -- Q3
```

4) List the players who have scored more than 300 runs in a single match.

```
select p.player_name
from players p inner join batting b
on p.player_id = b.player_id
where runs>300;                  -- Q4
```

5) List the players who have taken 5 or more wickets in a single match.

```
select p.player_name
from players p inner join bowling bo
on p.player_id = bo.player_id
where wickets >= 5;              -- Q5
```

6) List the matches where a particular team scored more than 200 runs.

```
select m.match_id,m.venue,m.match_date,t.team_name
from matches m inner join batting b
on m.match_id = b.match_id
inner join players p
on p.player_id = b.player_id
inner join teams t
on t.team_id = p.team_id
where b.runs>200 ;          -- Q6
```

7) List the matches where a particular player scored a century.

```
select m.match_id,m.venue,m.match_date,p.player_name
from matches m inner join batting b
on m.match_id = b.match_id
inner join players p
on b.player_id = p.player_id
where b.runs >= 100;            --  Q7
```

8) List the players who have played for more than one team.

```
select p.player_name from players p
group by p.player_name
having count(distinct team_id)>1; -- Q8
```

9) List the players who have not scored any runs in a match.

```
select p.player_name
from players p inner join batting b
on p.player_id = b.player_id
where b.runs is null;    -- Q9
```

10) List the players who have both batted and bowled in a single match.

```
select p.player_name
from players p inner join batting b
on p.player_id = b.player_id
inner join bowling bl
on p.player_id = bl.player_id; -- Q10
```

11) List the players who have scored a century and took a wicket in a single match.

```
select distinct p.player_name
from players p inner join batting b
on p.player_id = b.player_id
inner join bowling bl
on p.player_id = bl.player_id
where b.runs>=100 and bl.wickets>=1; -- Q11
```

12) List the players and their batting averages.

```
select p.player_name,sum(b.runs)/count(distinct b.match_id) as batting_average
from players p inner join batting b
on p.player_id = b.player_id
group by p.player_name;          -- Q12
```

13) List the players who have played at least 10 matches and have batting average over 30.

```
select p.player_name                          -- Q13
from players p inner join batting b
on p.player_id = b.player_id where b.match_id in
(
(select match_id from batting group by match_id having
count(distinct player_id)>=10)
and
(select sum(b.runs)/count(distinct b.match_id) from batting b where
b.player_id = p.player_id)>=30
);
```

14) List the players who have taken a Hatrick.

```
select p.player_name                    -- Q14
from players p inner join bowling b
on p.player_id= b.player_id
where b.wickets >=3
group by p.player_name,b.match_id,b.bowling_id
having count(distinct b.bowling_id) = 3
order by b.bowling_id;
```

15) List the teams and their win, loss Ratios.

```
select t.team_name,
count(case when m.team_1_id=t.team_id and m.team_1_id=m.winner_team_id then 1 end) as wins,
count(case when m.team_2_id=t.team_id and m.team_2_id=m.winner_team_id then 1 end) as losses,
count(case when m.team_1_id=t.team_id or m.team_2_id=t.team_id then 1 end) as total_matches,
(count(case when m.team_1_id=t.team_id and m.team_1_id=m.winner_team_id then 1 end)
/count(case when m.team_2_id=t.team_id and m.team_2_id=m.winner_team_id then 1 end)) as win_loss_ratios
from teams t inner join matches m
on t.team_id=m.team_1_id
or t.team_id=m.team_2_id
group by t.team_name;                   -- Q15
```

Hence as per the requirements the queries are being written. This shows how efficiently we can fetch data from a table just by writing queries.

To get the raw files of this particular project you can go through the below link.

Link - https://github.com/Kai1817/MySQL-Project/blob/main/MYSQL-PROJECT.sql