PROJECT SQL DATA ANALYSIS

The main aim of the project was to analyze the data of 2 tables and write down queries to give the required set of data**.**

- The first table given was named as Matches and it contains 816 rows and 16 columns namely **:** id, city, date, player_of_match,venue, neutral_venue ,team1, team2, toss_winner, toss_decision, winner, result, result_margin, eliminator , method, umpire1, umpire2

- The second table given was named as deliveries and it contains 193468 rows and columns namely : match_id, inning, over , ball, batsman, non_striker, bowler, batsman_runs extra_runs, total_runs, is_wicket, dismissal_kind, player_dismissed, fielder, extras_type, batting_team, bowling_team .

- Based upon the data we had to analyze ball by ball data of ipl matches from the year 2008 to year 2020.

- The set of Question were:

1. Create table named Matches
2. Create table named deliveries
3. Import data from csv for matches
4. Import data from csv for deliveries
5. Select top 20 rows of matches table
6. Select top 20 rows of deliveries table
7. Show all matches played on 2013/05/02
8. Show all matches where margin of victory >100
9. Show all matches where result was a tie.
10. Show distinct cities where matches were played.

11. Create a view delivries_02 with all columns of deliveries and an extra column names as ball_result (0=dot,runs>=4: boundry else other)
12. Count number of dot balls from deliveries_02
13. Count number of dot balls or boundry from deliveries_02
14. Show number of boundries hit by each batting team.
15. Show number of dot balls by each bowling team.
16. Count dismisaals according to dismissal kind.
17. Select top 5 bowlers who were hitted the most number of runs.
18. Create a table names as deliveries_03 with all columns of deliveries_02 and match date, venue of matches table;
19. Select runs hit in different venues in desc order.
20. Show year wise total runs hit in eden gardens.
21. Rising pune supergiants is misspleed as rising pune supergiant in few entries show original and corrected data.
22. Create table deliveries_04 with all columns of deliveries_03 and a column ball_id as(match_id-inings-over-ball)
23. Count distinct ball ids from deliveries_04.
24. Create table deliveries_05 s.t it shows row numbers(if ball id occoured first time it shows 1 if it is second time it shows 2).
25. Show all repeated rows in deliveries_05.

SOLUTION

Task 1

```
CREATE TABLE matches ( match_id int,

city varchar, date,

player_of_match varchar, venue varchar, neutral_venue int,

team1 varchar, team2 varchar, toss_winner varchar,

toss_decision varchar, winner varchar, result_mode varchar,
result_margin int, eliminator varchar, method_dl varchar, umpire1
varchar, umpire2 varchar

);
```

Task 2

```
drop table deliveries; CREATE TABLE deliveries ( match_id int,

inning int, over int, ball int,
batsman varchar, non_striker varchar, bowler varchar, batsman_runs int, extra_runs int, total_runs int, wicket_ball int, dismissal_kind varchar,

player_dismissed varchar, fielder varchar, extras_type varchar, batting_team varchar, bowling_team varchar

);
```

Task 3

```
copy matches from 'C:\Program Files\PostgreSQL\13\data\IPL\IPL_matches.csv' CSV header ;
```

Task 4

```
copy deliveries from 'C:\Program Files\PostgreSQL\13\data\IPL\IPL_Ball.csv' CSV header ;
```

Task 5

```
select * from deliveries limit 20;
```

Task 6

```
select * from matches limit 20;
```

Task 7

```
select * from matches where date = '02-05-2013';
```

Task 8

```
select * from matches where result_mode = 'runs' and result_margin > 100;
```

Task 9

```
select * from matches where result_mode ='tie' order by date desc;
```

Task 10

```
select count (distinct city) from matches;
```

Task 11

```
create table deliveries_v02 as select *,
CASE WHEN total_runs >= 4 then 'boundary' WHEN total_runs = 0 THEN 'dot'
```

```
else 'other' END as ball_result

FROM deliveries;
```

Task 12

```
select ball_result, count (*) from deliveries_v02 group by ball_result;
```

Task 13

```
select batting_team, count(*) from deliveries_v02 where ball_result =
'boundary' group by batting_team order by count desc;
```

Task 14

```
select bowling_team, count(*) from deliveries_v02 where ball_result =
'dot' group by bowling_team order by count desc;
```

Task 15

```
select dismissal_kind, count (*) from deliveries where dismissal_kind <>
'NA' group by dismissal_kind order by count desc;
```

Task 16

```
select bowler, sum(extra_runs) as total_extra_runs from deliveries
group by bowler order by total_extra_runs desc limit 5;
```

Task 17

```
create table deliveries_v03 AS SELECT a.*, b.venue, b.match_date from deliveries_v02 as a

left join (select max(venue) as venue, max(date) as match_date, match_id from matches group by match_id) as b

on a.match_id = b.match_id;
```

Task 18

```
select venue, sum(total_runs) as runs from deliveries_v03 group by venue order by runs desc;
```

Task 19

```
select extract(year from match_date) as IPL_year, sum(total_runs) as runs from deliveries_v03 where venue = 'Eden Gardens' group by IPL_year order by runs desc;
```

Task 20

```
select distinct team1 from matches;
```

```
create table matches_corrected as select *, replace(team1, 'Rising Pune Supergiants', 'Rising Pune Supergiant') as team1_corr

, replace(team2, 'Rising Pune Supergiants', 'Rising Pune Supergiant') as team2_corr from matches;

select distinct team1_corr from matches_corrected;
```

Task 21

```
create table deliveries_v04 as select concat(match_id,'-',inning,'-',over,'-',ball) as ball_id, * from deliveries_v03;
```

Task 22

```
select * from deliveries_v04 limit 20;

select count(distinct ball_id) from deliveries_v04; select count(*) from deliveries_v04;
```

Task 23

```
drop table deliveries_v05;
```

```
create table deliveries_v05 as select *, row_number() over (partition by
ball_id) as r_num from deliveries_v04;
```

Task 24

```
select count(*) from deliveries_v05; select sum(r_num) from
deliveries_v05;
```

```
select * from deliveries_v05 order by r_num limit 20; select * from
deliveries_v05 WHERE r_num=2;
```

Task 25

```
SELECT * FROM deliveries_v05 WHERE ball_id in (select BALL_ID from
deliveries_v05 WHERE r_num=2);
```