Olympic Games Database Management Project

Problem Statement

In this project, I have designed a database management system to store information about the Olympic games.

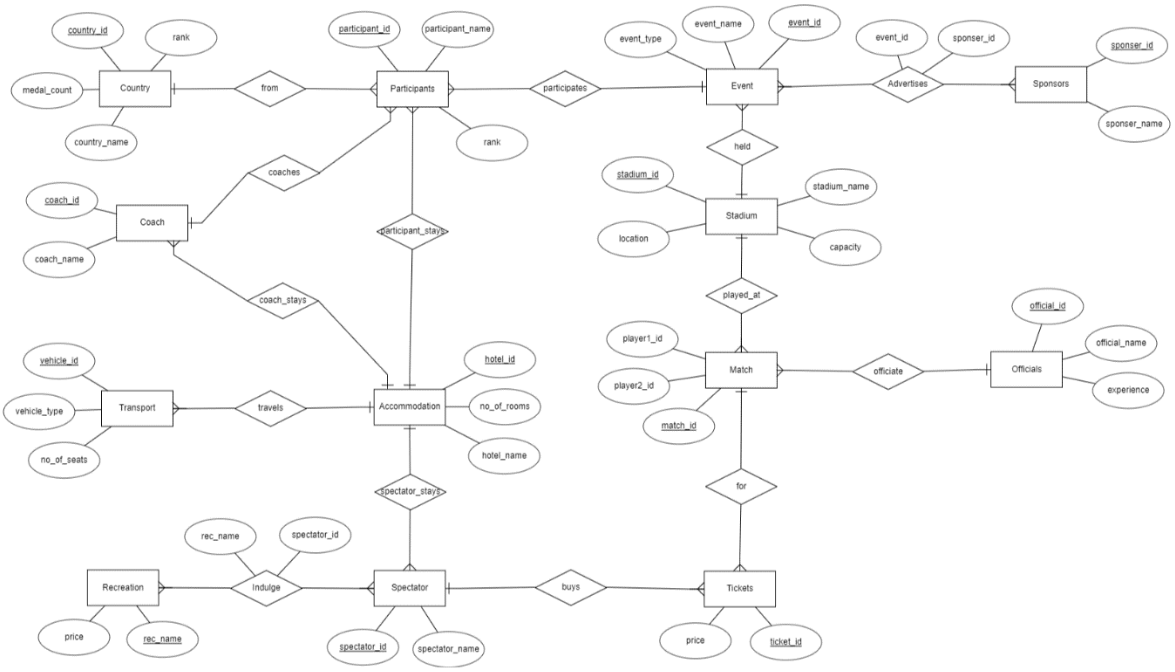
The database will contain details of the Olympic teams of each country, participating athletes and their support staff, sporting venues and spectating audiences' details. It also provides categorization of tickets and events, details of accommodation and transport for participants, ranking based on sport/country, match fixtures, sponsors, and other recreational facilities for both participants and audience.

This database was created with the objective of assisting the International Olympic Committee in successfully organizing the Olympic games. It aims to provide an efficient and secure environment for the organizing committee to store, access, manage and analyze the essential information for the smooth conduct of the Olympics.

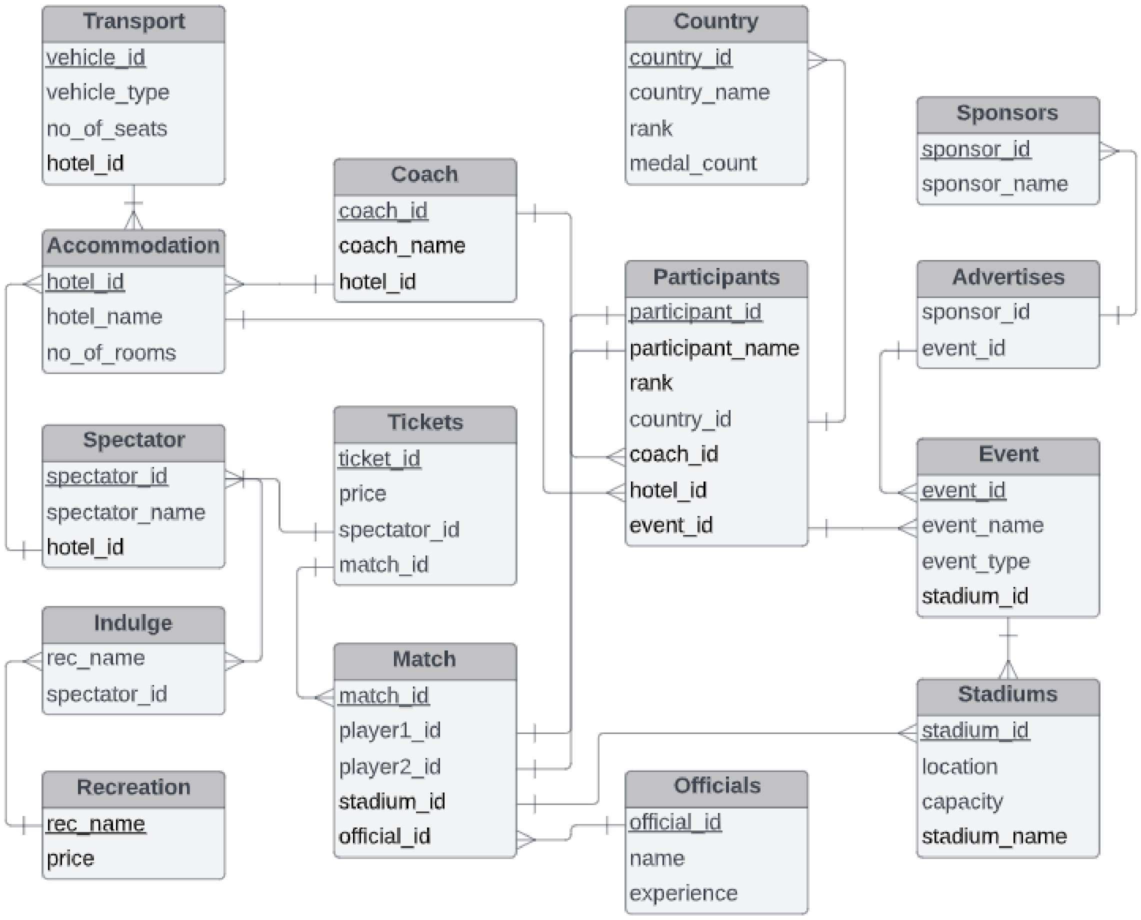
ER Model Assumptions

a) Only two players can play in a match.
b) Each player can only have one coach.
c) Each vehicle can only operate for one accommodation.
d) Each match can only have one official.
e) Each event can only take place in one stadium.

ER Diagram



Relational Schema:



Tables

Transport

Attribute	Data Type	Constraints
vehicle_id	NUMBER	PRIMARY KEY
vehicle_type	VARCHAR(30)	
no_of_seats	NUMBER	
hotel_id	NUMBER	FOREIGN KEY

Country

Attribute	Data Type	Constraints
country_id	NUMBER	PRIMARY KEY
country_name	VARCHAR(30)	
rank	NUMBER	
medal_count	NUMBER	

Sponsors

Attribute	Data Type	Constraints
sponsor_id	NUMBER	PRIMARY KEY
sponsor_name	VARCHAR(30)	

Coach

Attribute	Data Type	Constraints
coach_id	NUMBER	PRIMARY KEY
coach_name	VARCHAR(30)	
hotel_id	NUMBER	FOREIGN KEY

Accommodation

Attribute	Data Type	Constraints
hotel_id	NUMBER	PRIMARY KEY
hotel_name	VARCHAR(30)	
no_of_rooms	NUMBER	

Participants

Attribute	Data Type	Constraints
participant_id	NUMBER	PRIMARY KEY
participant_name	VARCHAR(30)	
rank	NUMBER	
country_id	NUMBER	FOREIGN KEY
coach_id	NUMBER	FOREIGN KEY
hotel_id	NUMBER	FOREIGN KEY
event_id	NUMBER	FOREIGN KEY

Advertises

Attribute	Data Type	Constraints
sponsor_id	NUMBER	FOREIGN KEY
event_id	NUMBER	FOREIGN KEY

Spectator

Attribute	Data Type	Constraints
spectator_id	NUMBER	PRIMARY KEY
spectator_name	VARCHAR(30)	
hotel_id	NUMBER	FOREIGN KEY

Tickets

Attribute	Data Type	Constraints
ticket_id	NUMBER	PRIMARY KEY
price	FLOAT	
spectator_id	NUMBER	FOREIGN KEY
match_id	NUMBER	FOREIGN KEY

Event

Attribute	Data Type	Constraints
event_id	NUMBER	PRIMARY KEY
event_name	VARCHAR(30)	
event_type	VARCHAR(30)	
stadium_id	NUMBER	FOREIGN KEY

Indulge

Attribute	Data Type	Constraints
rec_name	VARCHAR(30)	FOREIGN KEY
spectator_id	NUMBER	FOREIGN KEY

Recreation

Attribute	Data Type	Constraints
rec_name	VARCHAR(30)	PRIMARY KEY,
price	FLOAT	

Match

Attribute	Data Type	Constraints
match_id	NUMBER	PRIMARY KEY
player1_id	NUMBER	FOREIGN KEY
player2_id	NUMBER	FOREIGN KEY
stadium_id	NUMBER	FOREIGN KEY
official_id	NUMBER	FOREIGN KEY

Officials

Attribute	Data Type	Constraints
official_id	NUMBER	PRIMARY KEY
name	VARCHAR(30)	
experience	NUMBER	

Stadiums

Attribute	Data Type	Constraints
stadium_id	NUMBER	PRIMARY KEY
location	VARCHAR(30)	
capacity	NUMBER	
stadium_name	VARCHAR(30)	

Functional Dependency and Primary Key:

Sponsors:

sponsor_id -> {sponsor_id, sponsor_name}
Since all the fields depend on sponsor_id, $(\text{sponsor\_id})^{+}$ -> R
Hence, sponsor_id is Primary Key of Sponsors table

Accomodation:

hotel_id -> {hotel_id, hotel_name, no_of_rooms}
Since all the fields depend on hotel_id, (hotel_id)+ -> R
Hence, hotel_id is Primary Key of Accomodation table

Transport:

vehicle_id -> {vehicle_id, vehicle_type, no_of_seats, hotel_id}
Since all the fields depend on vehicle_id, (vehicle_id)+ -> R
Hence, vehicle_id is Primary Key of Transport table

Coach:

coach_id -> {coach_id, coach_name, hotel_id}
Since all the fields depend on coach_id, (coach_id)+ -> R
Hence, coach_id is Primary Key of Coach table

Stadiums:

stadium_id -> {stadium_id, location, capacity, stadium_name}
Since all the fields depend on stadium_id, (stadium_id)+ -> R
Hence, stadium_id is Primary Key of Stadiums table

Country:

country_id -> {country_id, country_name, rank, medal_count}
Since all the fields depend on stadium_id, (country_id)+ -> R
Hence, country_id is Primary Key of Country table

Officials:

official_id -> {official_id, name, experience}
Since all the fields depend on official_id, (official_id)+ -> R
Hence, official_id is Primary Key of Officials table

Spectators:

spectator_id -> {spectator_id, spectator_name, hotel_id}
Since all the fields depend on spectator_id, (spectator_id)+ -> R
Hence, spectator_id is Primary Key of Spectators table

Recreation:

rec_name -> {rec_name, price}
Since all the fields depend on rec_name, (rec_name)+ -> R
Hence, rec_name is Primary Key of Recreation table

Event:

event_id -> {event_id, event_name, event_type, stadium_id}
Since all the fields depend on event_id, (event_id)+ -> R
Hence, event_id is Primary Key of Event table

Participants:

participant_id -> {participant_id, participant_name, rank, country_id, coach_id, hotel_id, event_id}
Since all the fields depend on participant_id, (participant_id)+ -> R
Hence, participant_id is Primary Key of Participants table

Match:

match_id -> {match_id, player1_id, player2_id, stadium_id, official_id}
Since all the fields depend on match_id, (match_id)+ -> R
Hence, match_id is Primary Key of Match table

Tickets:

ticket_id -> {ticket_id, price, spectator_id, match_id}
Since all the fields depend on ticket_id, (ticket_id)+ -> R
Hence, ticket_id is Primary Key of Tickets table

Indulge:

{rec_name, spectator_id} -> {rec_name, spectator_id}
Since all the fields depend on {rec_name, spectator_id}, {rec_name, spectator_id}+ -> R
Hence, {rec_name, spectator_id} is Primary Key of Indulge table

Advertises:

{sponsor_id, event_id} -> {sponsor_id, event_id}
Since all the fields depend on {sponsor_id, event_id}, {sponsor_id, event_id}+ -> R
Hence, {sponsor_id, event_id} is Primary Key of Advertises table

Normalisation:

Sponsors:

Primary key: sponsor_id
All attributes depend on the sponsor_id, hence the table is 2NF.
All attributes depend directly on sponsor_id, hence the table is in 3NF.
All determinants sponsor_id are candidate keys, hence the table is in BCNF.

Accomodation:

Primary key: hotel_id
All attributes depend on the hotel_id, hence the table is 2NF.
All attributes depend directly on hotel_id, hence the table is in 3NF.
All determinants hotel_id are candidate keys, hence the table is in BCNF.

Transport:

Primary key: vehicle_id
All attributes depend on the vehicle_id, hence the table is 2NF.
All attributes depend directly on vehicle_id, hence the table is in 3NF.
All determinants vehicle_id are candidate keys, hence the table is in BCNF.

Coach:

Primary key: coach_id
All attributes depend on the coach_id, hence the table is 2NF.
All attributes depend directly on coach_id, hence the table is in 3NF.
All determinants coach_id are candidate keys, hence the table is in BCNF.

Stadiums:

Primary key: stadium_id
All attributes depend on the stadium_id, hence the table is 2NF.
All attributes depend directly on stadium_id, hence the table is in 3NF.
All determinants stadium_id are candidate keys, hence the table is in BCNF.

Country:

Primary key: country_id
All attributes depend on the country_id, hence the table is 2NF.
All attributes depend directly on country_id, hence the table is in 3NF.
All determinants country_id are candidate keys, hence the table is in BCNF.

Officials:

Primary key: official_id
All attributes depend on the official_id, hence the table is 2NF.
All attributes depend directly on official_id, hence the table is in 3NF.
All determinants official_id are candidate keys, hence the table is in BCNF.

Spectators:

Primary key: spectator_id
All attributes depend on the spectator_id, hence the table is 2NF.
All attributes depend directly on spectator_id, hence the table is in 3NF.
All determinants spectator_id are candidate keys, hence the table is in BCNF.

Recreation:

Primary key: rec_name
All attributes depend on the rec_name, hence the table is 2NF.
All attributes depend directly on rec_name, hence the table is in 3NF.
All determinants rec_name are candidate keys, hence the table is in BCNF.

Event:

Primary key: event_id
All attributes depend on the event_id, hence the table is 2NF.
All attributes depend directly on event_id, hence the table is in 3NF.
All determinants event_id are candidate keys, hence the table is in BCNF.

Participants:

Primary key: participant_id
All attributes depend on the participant_id, hence the table is 2NF.
All attributes depend directly on participant_id, hence the table is in 3NF.
All determinants participant_id are candidate keys, hence the table is in BCNF.

Match:

Primary key: match_id
All attributes depend on the match_id, hence the table is 2NF.
All attributes depend directly on match_id, hence the table is in 3NF.
All determinants match_id are candidate keys, hence the table is in BCNF.

Tickets:

Primary key: ticket_id
All attributes depend on the ticket_id, hence the table is 2NF.
All attributes depend directly on ticket_id , hence the table is in 3NF.
All determinants ticket_id are candidate keys, hence the table is in BCNF.

Indulge:

Primary key: {rec_name, spectator_id}
All attributes depend on the {rec_name, spectator_id}, hence the table is 2NF.
All attributes depend directly on {rec_name, spectator_id}, hence the table is in 3NF.
All determinants {rec_name, spectator_id} are candidate keys, hence the table is in BCNF.

Advertises:

Primary key: {sponsor_id, event_id}
All attributes depend on the {sponsor_id, event_id}, hence the table is 2NF.
All attributes depend directly on {sponsor_id, event_id}, hence the table is in 3NF.
All determinants {sponsor_id, event_id} are candidate keys, hence the table is in BCNF.

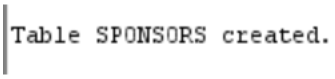
SQL Code

Creation of Tables

1. **Sponsors**

```
CREATE TABLE Sponsors (
  sponsor_id NUMBER PRIMARY KEY,
  sponsor_name VARCHAR(30)
);
```

Output:

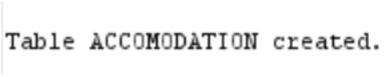


```
Table SPONSORS created.
```

2. **Accomodation**

```
CREATE TABLE Accomodation (
  hotel_id NUMBER PRIMARY KEY,
  hotel_name VARCHAR(30),
  no_of_rooms NUMBER
);
```

Output:



```
Table ACCOMODATION created.
```

3. Transport

```
CREATE TABLE Transport (
  vehicle_id NUMBER PRIMARY KEY,
  vehicle_type VARCHAR(30),
  no_of_seats NUMBER,
  hotel_id NUMBER,
  FOREIGN KEY(hotel_id) REFERENCES Accomodation
);
```

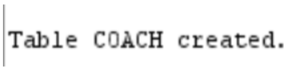
Output:

```
Table TRANSPORT created.
```

4. Coach

```
CREATE TABLE Coach (
  coach_id NUMBER PRIMARY KEY,
  coach_name VARCHAR(30),
  hotel_id NUMBER,
  FOREIGN KEY(hotel_id) REFERENCES Accomodation
);
```

Output:

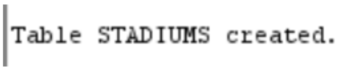


```
Table COACH created.
```

5. Stadiums

```
CREATE TABLE Stadiums (
  stadium_id NUMBER PRIMARY KEY,
  location VARCHAR(30),
  capacity NUMBER,
  stadium_name VARCHAR(30)
);
```

Output:

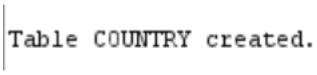


```
Table STADIUMS created.
```

6. Country

```
CREATE TABLE Country (
  country_id NUMBER PRIMARY KEY,
  country_name VARCHAR(30),
  rank NUMBER,
  medal_count NUMBER
);
```

Output:

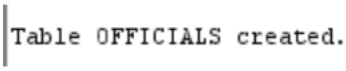


```
Table COUNTRY created.
```

7. Officials

```
CREATE TABLE Officials (
  official_id NUMBER PRIMARY KEY,
  name VARCHAR(30),
  experience NUMBER
);
```

Output:



```
Table OFFICIALS created.
```

8. **Spectator**

```
CREATE TABLE Spectator (
  spectator_id NUMBER PRIMARY KEY,
  spectator_name VARCHAR(30),
  hotel_id NUMBER,
  FOREIGN KEY(hotel_id) REFERENCES Accomodation
);
```

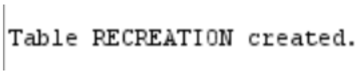
Output:

```
Table SPECTATOR created.
```

9. **Recreation**

```
CREATE TABLE Recreation (
  rec_name VARCHAR(30) PRIMARY KEY,
  price FLOAT
);
```

Output:



```
Table RECREATION created.
```

10. Event

```
CREATE TABLE Event (
  event_id NUMBER PRIMARY KEY,
  event_name VARCHAR(30),
  event_type VARCHAR(30),
  stadium_id NUMBER,
  FOREIGN KEY(stadium_id) REFERENCES Stadiums
);
```

Output:

```
Table EVENT created.
```

11. Participants

```
CREATE TABLE Participants (
  participant_id NUMBER PRIMARY KEY,
  participant_name VARCHAR(30),
  rank NUMBER,
  country_id NUMBER,
  coach_id NUMBER,
  hotel_id NUMBER,
  event_id NUMBER,
  FOREIGN KEY(country_id) REFERENCES Country,
  FOREIGN KEY(coach_id) REFERENCES Coach,
  FOREIGN KEY(hotel_id) REFERENCES Accomodation,
  FOREIGN KEY(event_id) REFERENCES Event
);
```

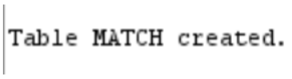
Output:

```
Table PARTICIPANTS created.
```

12.Match

```
CREATE TABLE Match (
  match_id NUMBER PRIMARY KEY,
  player1_id NUMBER,
  player2_id NUMBER,
  stadium_id NUMBER,
  official_id NUMBER,
  FOREIGN KEY(player1_id) REFERENCES Participants,
  FOREIGN KEY(player2_id) REFERENCES Participants,
  FOREIGN KEY(stadium_id) REFERENCES Stadiums,
  FOREIGN KEY(official_id) REFERENCES Officials
);
```

Output:

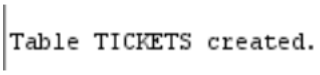


```
Table MATCH created.
```

13. Tickets

```
CREATE TABLE Tickets (
  ticket_id NUMBER PRIMARY KEY,
  price FLOAT,
  spectator_id NUMBER,
  match_id NUMBER,
  FOREIGN KEY(spectator_id) REFERENCES Spectator,
  FOREIGN KEY(match_id) REFERENCES Match
);
```

Output:

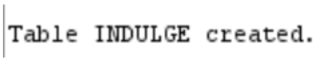


```
Table TICKETS created.
```

14. Indulge

```
CREATE TABLE Indulge (
  rec_name VARCHAR(30),
  spectator_id NUMBER,
  FOREIGN KEY(rec_name) REFERENCES Recreation,
  FOREIGN KEY(spectator_id) REFERENCES Spectator
);
```

Output:

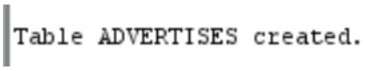


```
Table INDULGE created.
```

15. Advertises

```
CREATE TABLE Advertises (
  sponsor_id NUMBER,
  event_id NUMBER,
  FOREIGN KEY(sponsor_id) REFERENCES Sponsors,
  FOREIGN KEY(event_id) REFERENCES Event
);
```

Output:



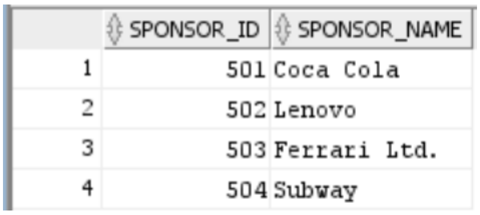
```
Table ADVERTISES created.
```

Insertion of Values into Tables

1. **Sponsors**

```
INSERT INTO Sponsors VALUES(501, 'Coca Cola');
INSERT INTO Sponsors VALUES(502, 'Lenovo');
INSERT INTO Sponsors VALUES(503, 'Ferrari Ltd.');
INSERT INTO Sponsors VALUES(504, 'Subway');
SELECT * FROM Sponsors;
```

Output:

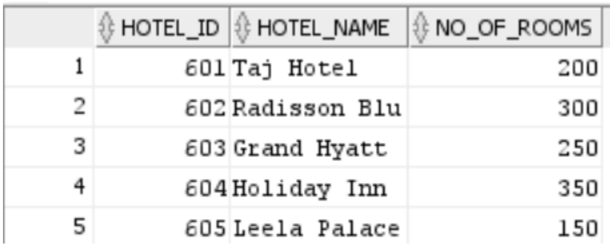


	SPONSOR_ID	SPONSOR_NAME
1	501	Coca Cola
2	502	Lenovo
3	503	Ferrari Ltd.
4	504	Subway

2. **Accomodation**

```
INSERT INTO Accomodation VALUES(601, 'Taj Hotel', 200);
INSERT INTO Accomodation VALUES(602, 'Radisson Blu', 300);
INSERT INTO Accomodation VALUES(603, 'Grand Hyatt', 250);
INSERT INTO Accomodation VALUES(604, 'Holiday Inn', 350);
INSERT INTO Accomodation VALUES(605, 'Leela Palace', 150);
SELECT * FROM Accomodation;
```

Output:

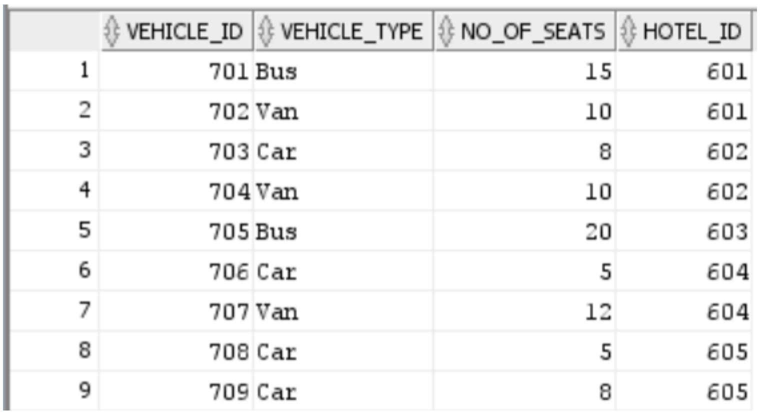


	HOTEL_ID	HOTEL_NAME	NO_OF_ROOMS
1	601	Taj Hotel	200
2	602	Radisson Blu	300
3	603	Grand Hyatt	250
4	604	Holiday Inn	350
5	605	Leela Palace	150

3. Transport

```
INSERT INTO Transport VALUES(701, 'Bus', 15, 601);
INSERT INTO Transport VALUES(702, 'Van', 10, 601);
INSERT INTO Transport VALUES(703, 'Car', 8, 602);
INSERT INTO Transport VALUES(704, 'Van', 10, 602);
INSERT INTO Transport VALUES(705, 'Bus', 20, 603);
INSERT INTO Transport VALUES(706, 'Car', 5, 604);
INSERT INTO Transport VALUES(707, 'Van', 12, 604);
INSERT INTO Transport VALUES(708, 'Car', 5, 605);
INSERT INTO Transport VALUES(709, 'Car', 8, 605);
SELECT * FROM Transport;
```

Output:

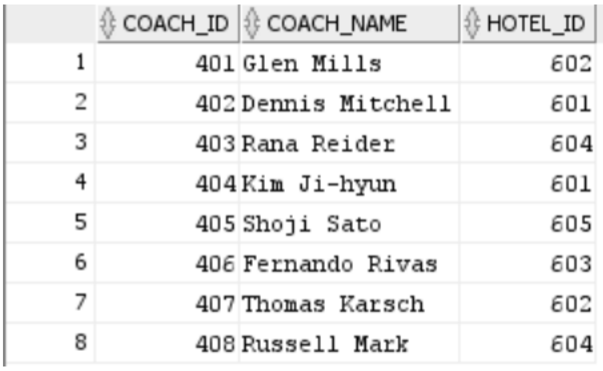


	VEHICLE_ID	VEHICLE_TYPE	NO_OF_SEATS	HOTEL_ID
1	701	Bus	15	601
2	702	Van	10	601
3	703	Car	8	602
4	704	Van	10	602
5	705	Bus	20	603
6	706	Car	5	604
7	707	Van	12	604
8	708	Car	5	605
9	709	Car	8	605

4. Coach

```
INSERT INTO Coach VALUES(401, 'Glen Mills', 602);
INSERT INTO Coach VALUES(402, 'Dennis Mitchell', 601);
INSERT INTO Coach VALUES(403, 'Rana Reider', 604);
INSERT INTO Coach VALUES(404, 'Kim Ji-hyun', 601);
INSERT INTO Coach VALUES(405, 'Shoji Sato', 605);
INSERT INTO Coach VALUES(406, 'Fernando Rivas', 603);
INSERT INTO Coach VALUES(407, 'Thomas Karsch', 602);
INSERT INTO Coach VALUES(408, 'Russell Mark', 604);
SELECT * FROM Coach;
```

Output:

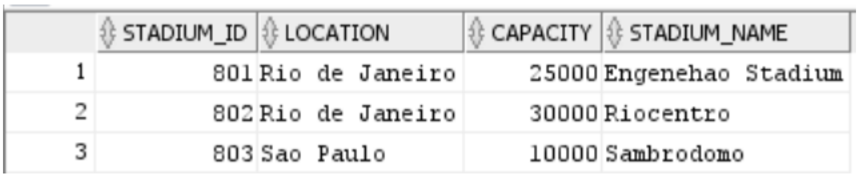


	COACH_ID	COACH_NAME	HOTEL_ID
1	401	Glen Mills	602
2	402	Dennis Mitchell	601
3	403	Rana Reider	604
4	404	Kim Ji-hyun	601
5	405	Shoji Sato	605
6	406	Fernando Rivas	603
7	407	Thomas Karsch	602
8	408	Russell Mark	604

5. **Stadiums**

INSERT INTO Stadiums VALUES(801, 'Rio de Janeiro', 25000, 'Engenehao Stadium');
INSERT INTO Stadiums VALUES(802, 'Rio de Janeiro', 30000, 'Riocentro');
INSERT INTO Stadiums VALUES(803, 'Sao Paulo', 10000, 'Sambrodomo');
SELECT * FROM Stadiums;

Output:

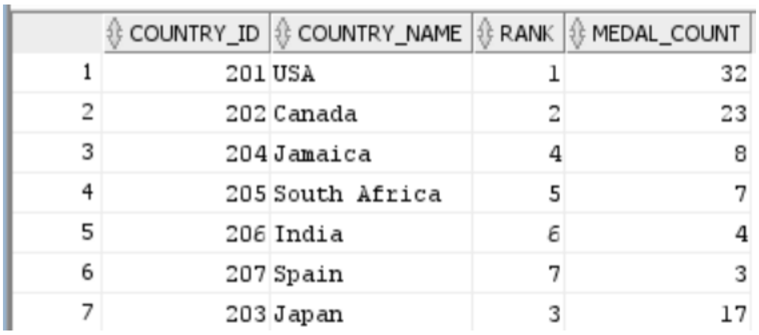


	STADIUM_ID	LOCATION	CAPACITY	STADIUM_NAME
1	801	Rio de Janeiro	25000	Engenehao Stadium
2	802	Rio de Janeiro	30000	Riocentro
3	803	Sao Paulo	10000	Sambrodomo

6. **Country**

INSERT INTO Country values(201, 'USA', 1, 32);
INSERT INTO Country values(202, 'Canada', 2, 23);
INSERT INTO Country values(203, 'Japan', 3, 17);
INSERT INTO Country values(204, 'Jamaica', 4, 8);
INSERT INTO Country values(205, 'South Africa', 5, 7);
INSERT INTO Country values(206, 'India', 6, 4);
INSERT INTO Country values(207, 'Spain', 7, 3);
SELECT * FROM Country;

Output:

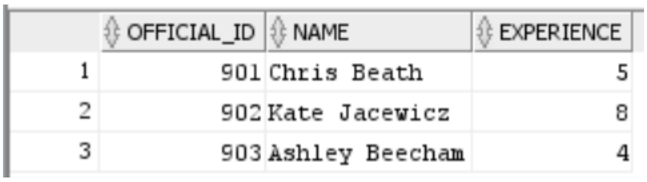


	COUNTRY_ID	COUNTRY_NAME	RANK	MEDAL_COUNT
1	201	USA	1	32
2	202	Canada	2	23
3	204	Jamaica	4	8
4	205	South Africa	5	7
5	206	India	6	4
6	207	Spain	7	3
7	203	Japan	3	17

7. **Officials**

```
INSERT INTO Officials VALUES(901, 'Chris Beath', 5);
INSERT INTO Officials VALUES(902, 'Kate Jacewicz', 8);
INSERT INTO Officials VALUES(903, 'Ashley Beecham', 4);
SELECT * FROM Officials;
```

Output:

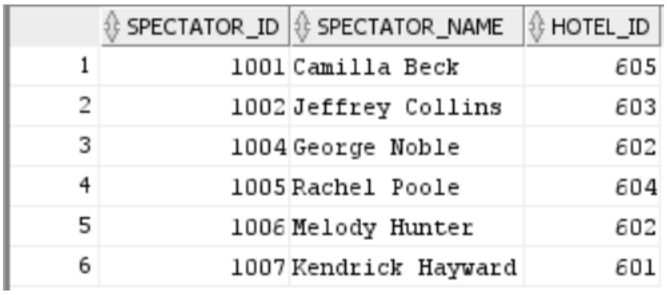


	OFFICIAL_ID	NAME	EXPERIENCE
1	901	Chris Beath	5
2	902	Kate Jacewicz	8
3	903	Ashley Beecham	4

8. **Spectator**

```
INSERT INTO Spectator VALUES(1001, 'Camilla Beck', 605);
INSERT INTO Spectator VALUES(1002, 'Jeffrey Collins', 603);
INSERT INTO Spectator VALUES(1004, 'George Noble', 602);
INSERT INTO Spectator VALUES(1005, 'Rachel Poole', 604);
INSERT INTO Spectator VALUES(1006, 'Melody Hunter', 602);
INSERT INTO Spectator VALUES(1007, 'Kendrick Hayward', 601);
SELECT * FROM Spectator;
```

Output:

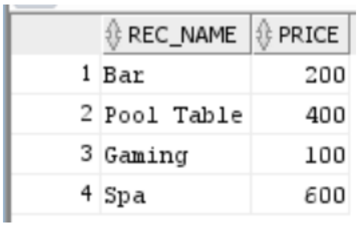


	SPECTATOR_ID	SPECTATOR_NAME	HOTEL_ID
1	1001	Camilla Beck	605
2	1002	Jeffrey Collins	603
3	1004	George Noble	602
4	1005	Rachel Poole	604
5	1006	Melody Hunter	602
6	1007	Kendrick Hayward	601

9. Recreation

```
INSERT INTO Recreation VALUES('Bar', 200);
INSERT INTO Recreation VALUES('Pool Table', 400);
INSERT INTO Recreation VALUES('Gaming', 100);
INSERT INTO Recreation VALUES('Spa', 600);
SELECT * FROM Recreation;
```

Output:

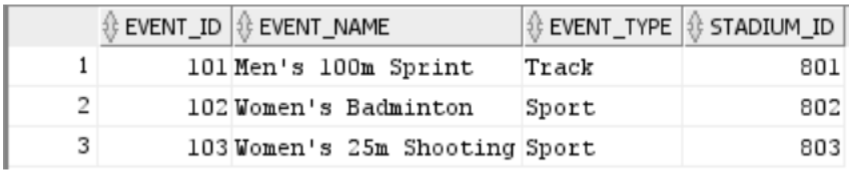


	REC_NAME	PRICE
1	Bar	200
2	Pool Table	400
3	Gaming	100
4	Spa	600

10. Event

```
INSERT INTO Event VALUES(101, 'Men''s 100m Sprint', 'Track', 801);
INSERT INTO Event VALUES(102, 'Women''s Badminton', 'Sport', 802);
INSERT INTO Event VALUES(103, 'Women''s 25m Shooting', 'Sport', 803);
SELECT * FROM Event;
```

Output:

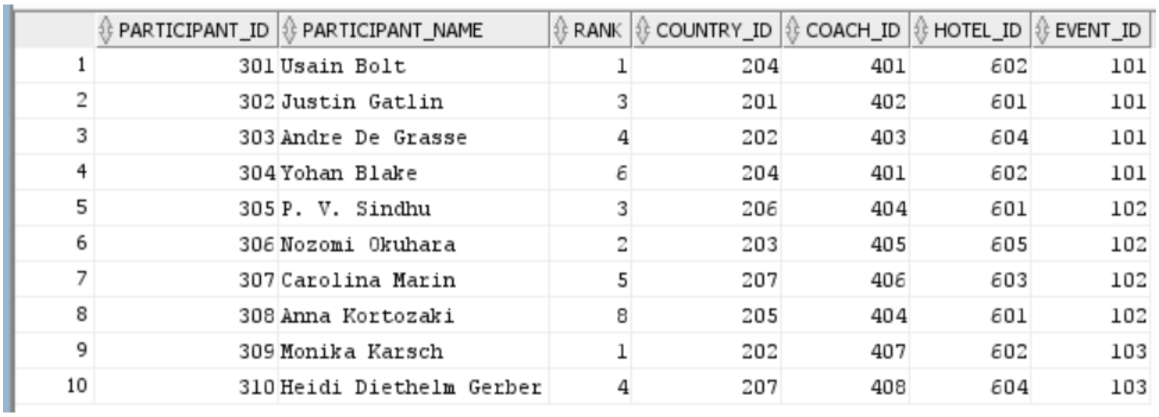


	EVENT_ID	EVENT_NAME	EVENT_TYPE	STADIUM_ID
1	101	Men's 100m Sprint	Track	801
2	102	Women's Badminton	Sport	802
3	103	Women's 25m Shooting	Sport	803

11. Participants

```
INSERT INTO Participants VALUES(301, 'Usain Bolt', 1, 204, 401, 602, 101);
INSERT INTO Participants VALUES(302, 'Justin Gatlin', 3, 201, 402, 601, 101);
INSERT INTO Participants VALUES(303, 'Andre De Grasse', 4, 202, 403, 604, 101);
INSERT INTO Participants VALUES(304, 'Yohan Blake', 6, 204, 401, 602, 101);
INSERT INTO Participants VALUES(305, 'P. V. Sindhu', 3, 206, 404, 601, 102);
INSERT INTO Participants VALUES(306, 'Nozomi Okuhara', 2, 203, 405, 605, 102);
INSERT INTO Participants VALUES(307, 'Carolina Marin', 5, 207, 406, 603, 102);
INSERT INTO Participants VALUES(308, 'Anna Kortozaki', 8, 205, 404, 601, 102);
INSERT INTO Participants VALUES(309, 'Monika Karsch', 1, 202, 407, 602, 103);
INSERT INTO Participants VALUES(310, 'Heidi Diethelm Gerber', 4, 207, 408, 604, 103);
SELECT * FROM Participants;
```

Output:

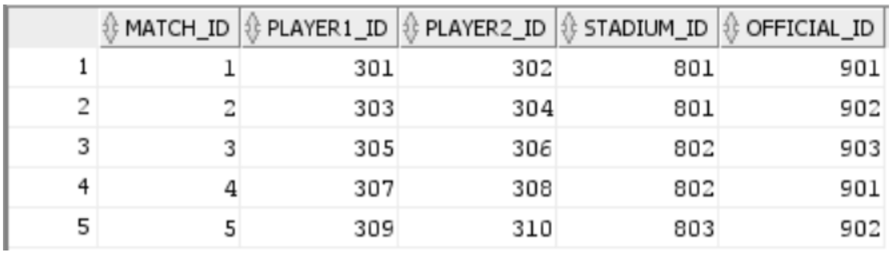


	PARTICIPANT_ID	PARTICIPANT_NAME	RANK	COUNTRY_ID	COACH_ID	HOTEL_ID	EVENT_ID
1	301	Usain Bolt	1	204	401	602	101
2	302	Justin Gatlin	3	201	402	601	101
3	303	Andre De Grasse	4	202	403	604	101
4	304	Yohan Blake	6	204	401	602	101
5	305	P. V. Sindhu	3	206	404	601	102
6	306	Nozomi Okuhara	2	203	405	605	102
7	307	Carolina Marin	5	207	406	603	102
8	308	Anna Kortozaki	8	205	404	601	102
9	309	Monika Karsch	1	202	407	602	103
10	310	Heidi Diethelm Gerber	4	207	408	604	103

12. Match

```
INSERT INTO Match VALUES(1, 301, 302, 801, 901);
INSERT INTO Match VALUES(2, 303, 304, 801, 902);
INSERT INTO Match VALUES(3, 305, 306, 802, 903);
INSERT INTO Match VALUES(4, 307, 308, 802, 901);
INSERT INTO Match VALUES(5, 309, 310, 803, 902);
SELECT * FROM Match;
```

Output:

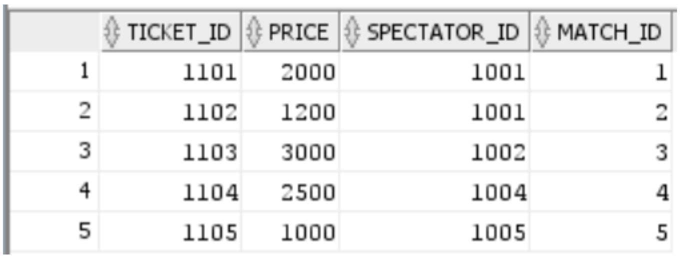


	MATCH_ID	PLAYER1_ID	PLAYER2_ID	STADIUM_ID	OFFICIAL_ID
1	1	301	302	801	901
2	2	303	304	801	902
3	3	305	306	802	903
4	4	307	308	802	901
5	5	309	310	803	902

13. Tickets

```
INSERT INTO Tickets VALUES(1101, 2000, 1001, 1);
INSERT INTO Tickets VALUES(1102, 1200, 1001, 2);
INSERT INTO Tickets VALUES(1103, 3000, 1002, 3);
INSERT INTO Tickets VALUES(1104, 2500, 1004, 4);
INSERT INTO Tickets VALUES(1105, 1000, 1005, 5);
SELECT * FROM Tickets;
```

Output:

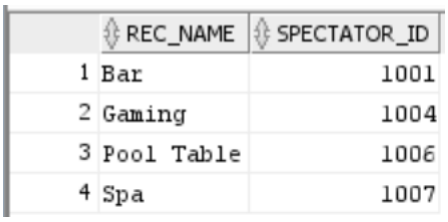


	TICKET_ID	PRICE	SPECTATOR_ID	MATCH_ID
1	1101	2000	1001	1
2	1102	1200	1001	2
3	1103	3000	1002	3
4	1104	2500	1004	4
5	1105	1000	1005	5

14. Indulge

```
INSERT INTO Indulge VALUES('Bar', 1001);
INSERT INTO Indulge VALUES('Gaming', 1004);
INSERT INTO Indulge VALUES('Pool Table', 1006);
INSERT INTO Indulge VALUES('Spa', 1007);
SELECT * FROM Indulge;
```

Output:

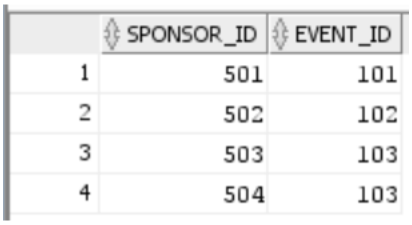


	REC_NAME	SPECTATOR_ID
1	Bar	1001
2	Gaming	1004
3	Pool Table	1006
4	Spa	1007

15. Advertises

```
INSERT INTO Advertises VALUES(501, 101);
INSERT INTO Advertises VALUES(502, 102);
INSERT INTO Advertises VALUES(503, 103);
INSERT INTO Advertises VALUES(504, 103);
SELECT * FROM Advertises;
```

Output:



	SPONSOR_ID	EVENT_ID
1	501	101
2	502	102
3	503	103
4	504	103