NEWGIZA UNIVERSITY
SCHOOL of
INFORMATION TECHNOLOGY

Sports League Management System

This **Final Project** is built to fulfill the requirements of

CSAI 410 - Advanced Database Systems

Fall 2024

Submitted to : Dr. Zeinab Abd El Haliem

Name	***ID***
Mai Waheed	***202200556***
Lujain Ahmad	***202201738***
Zeina Ayman	***202200351***

System Requirements

1.1 Purpose and Functionality

The primary goal of this database system is to serve as a comprehensive platform for managing and analyzing data related to a sports league. It organizes information about teams, players, and matches across multiple sports, enabling efficient tracking of player statistics, team performance, and match results. It enables efficient storage and querying of information related to team details, match schedules, and player performance, facilitating the analysis of trends and league operations. Its purpose is to empower stakeholders with precise, real-time insights, transforming raw data into actionable strategies that enhance team performance, improve player development, and drive fan engagement.

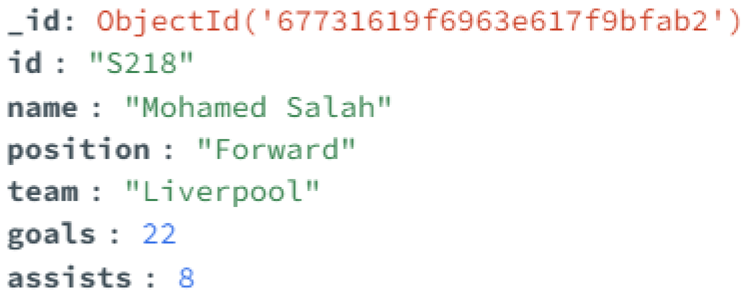
1.2 Domain or Industry

The domain of this database system is the sports and entertainment industry. It is tailored to serve organizations managing professional and amateur sports leagues, including clubs, teams, coaches, and analysts across multiple sports such as soccer, basketball, tennis, and swimming. Additionally, it caters to sports analytics companies, event organizers, and fan engagement platforms by providing comprehensive tools for performance tracking, league management, and audience interaction analysis.

1.3 Collections and Documents

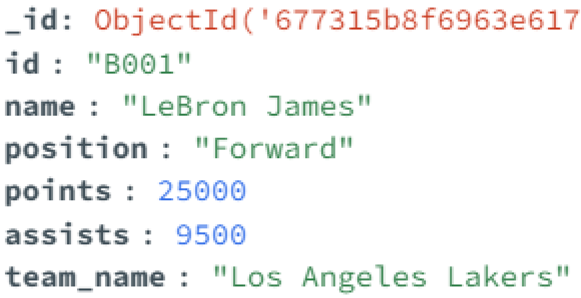
The database consists of the following collections:

1. **SoccerPlayers:**
   Description: Includes data about football players, specifying their names, roles (e.g., Goalkeeper, Defender, Midfielder, Forward), total goals, assists, and the teams they represent.



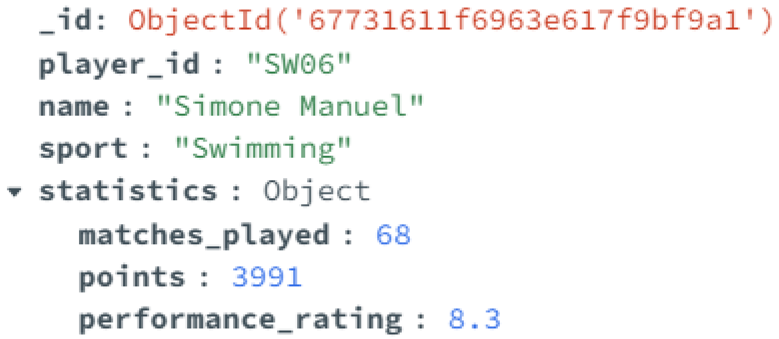
```
_id: ObjectId('67731619f6963e617f9bfab2')
id : "S218"
name : "Mohamed Salah"
position : "Forward"
team : "Liverpool"
goals : 22
assists : 8
```

2. **BasketBallPlayers**
   Description: Contains profiles of basketball players, including their names, positions (e.g., Forward, Center, Guard), total points scored, assists made, and the teams they play for.



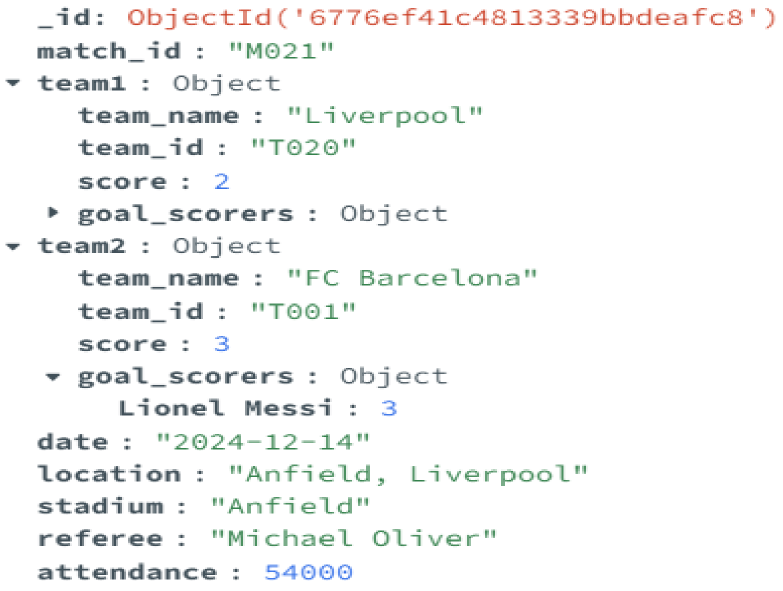
```
_id: ObjectId('677315b8f6963e617f9bf965')
id : "B001"
name : "LeBron James"
position : "Forward"
points : 25000
assists : 9500
team_name : "Los Angeles Lakers"
```

3. **SinglePlayers:**
   Description: Focuses on individual athletes from sports like swimming and tennis, offering details such as matches played, total points, and performance ratings.



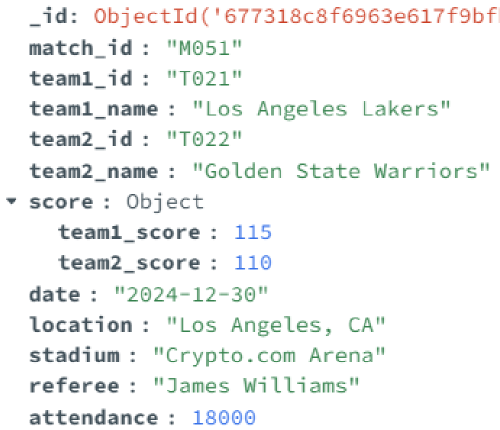
```
  _id: ObjectId('67731611f6963e617f9bf9a1')
  player_id : "SW06"
  name : "Simone Manuel"
  sport : "Swimming"
▾ statistics : Object
    matches_played : 68
    points : 3991
    performance_rating : 8.3
```

4. **SoccerMatches:**
   Description: Logs of football matches, showing the competing teams, final scores, dates, stadiums, referees, and audience size.



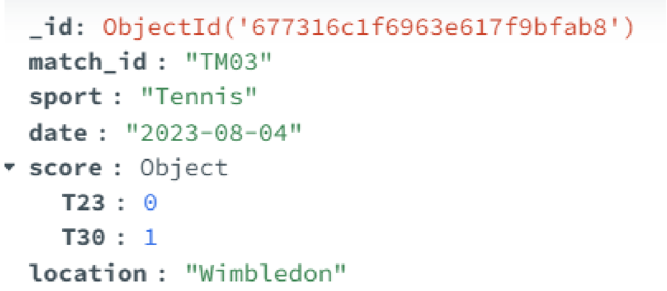
```
  _id: ObjectId('6776ef41c4813339bbdeafc8')
  match_id : "M021"
▾ team1 : Object
    team_name : "Liverpool"
    team_id : "T020"
    score : 2
  ▸ goal_scorers : Object
▾ team2 : Object
    team_name : "FC Barcelona"
    team_id : "T001"
    score : 3
  ▾ goal_scorers : Object
      Lionel Messi : 3
  date : "2024-12-14"
  location : "Anfield, Liverpool"
  stadium : "Anfield"
  referee : "Michael Oliver"
  attendance : 54000
```

5. **BasketBallMatches:**
   Description: Records games played between basketball teams, providing information on team names, scores, match dates, venues, and attendance.



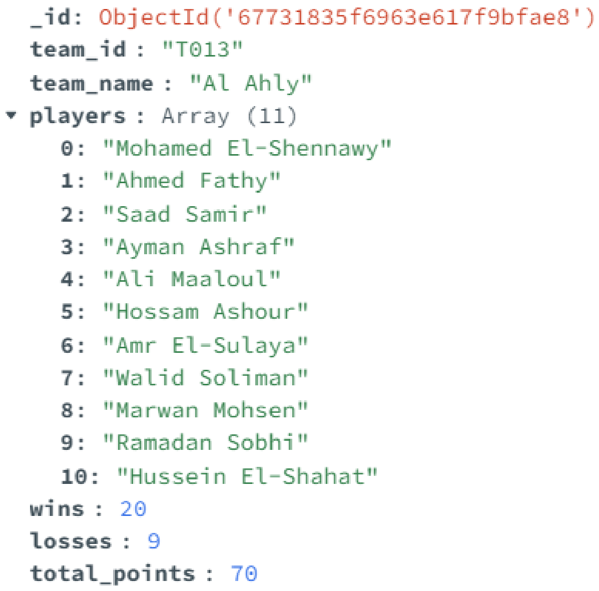
```
  _id: ObjectId('677318c8f6963e617f9bfb00')
  match_id : "M051"
  team1_id : "T021"
  team1_name : "Los Angeles Lakers"
  team2_id : "T022"
  team2_name : "Golden State Warriors"
▾ score : Object
    team1_score : 115
    team2_score : 110
  date : "2024-12-30"
  location : "Los Angeles, CA"
  stadium : "Crypto.com Arena"
  referee : "James Williams"
  attendance : 18000
```

6. **SingleMatches:**
   Description: Details results of individual sports matches, including tennis and swimming, along with scores, participants, dates, and locations.



```
  _id: ObjectId('677316c1f6963e617f9bfab8')
  match_id : "TM03"
  sport : "Tennis"
  date : "2023-08-04"
▾ score : Object
    T23 : 0
    T30 : 1
  location : "Wimbledon"
```

7. **Teams:**
   Description: Lists sports teams along with their player rosters, total wins, losses, and accumulated points.



```
  _id: ObjectId('67731835f6963e617f9bfae8')
  team_id : "T013"
  team_name : "Al Ahly"
▾ players : Array (11)
    0: "Mohamed El-Shennawy"
    1: "Ahmed Fathy"
    2: "Saad Samir"
    3: "Ayman Ashraf"
    4: "Ali Maaloul"
    5: "Hossam Ashour"
    6: "Amr El-Sulaya"
    7: "Walid Soliman"
    8: "Marwan Mohsen"
    9: "Ramadan Sobhi"
    10: "Hussein El-Shahat"
  wins : 20
  losses : 9
  total_points : 70
```

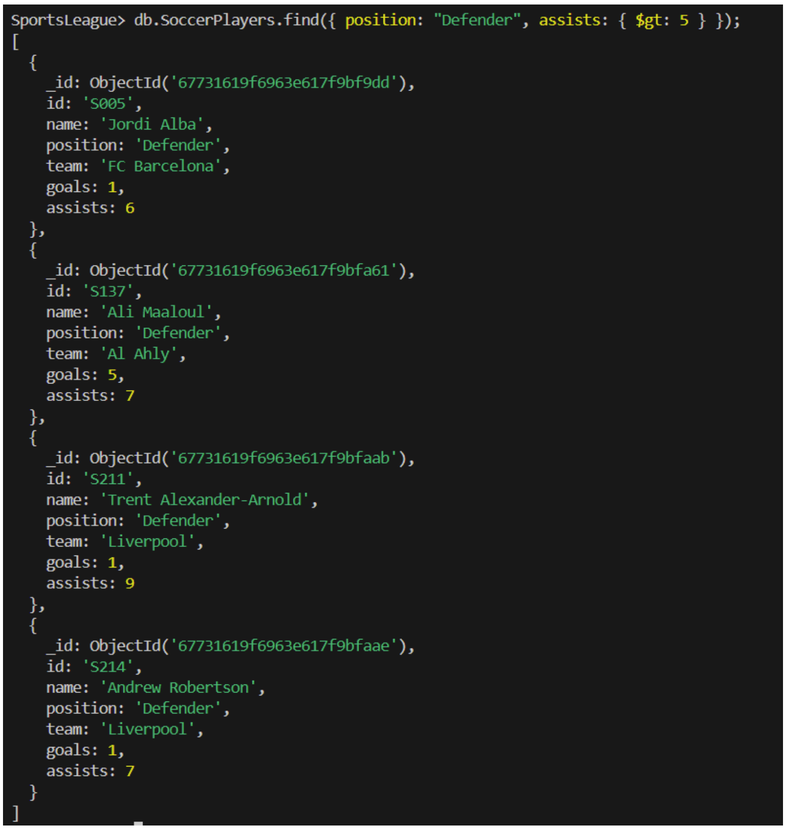
System Implementation

2.1 Tools Selection

We used MongoDB for entering and managing our data.

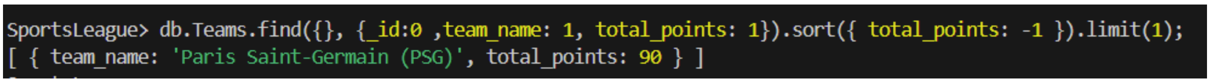
2.2 Queries

1. Find all soccer Defenders with more than 5 assists:



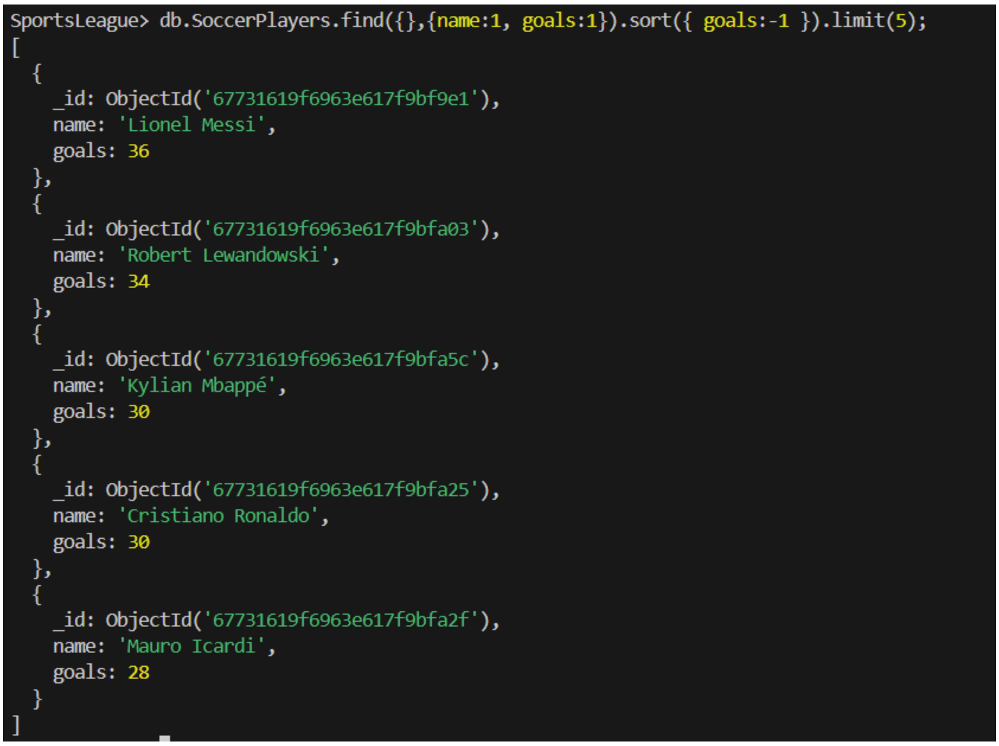
```
SportsLeague> db.SoccerPlayers.find({ position: "Defender", assists: { $gt: 5 } });
[
  {
    _id: ObjectId('67731619f6963e617f9bf9dd'),
    id: 'S005',
    name: 'Jordi Alba',
    position: 'Defender',
    team: 'FC Barcelona',
    goals: 1,
    assists: 6
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa61'),
    id: 'S137',
    name: 'Ali Maaloul',
    position: 'Defender',
    team: 'Al Ahly',
    goals: 5,
    assists: 7
  },
  {
    _id: ObjectId('67731619f6963e617f9bfaab'),
    id: 'S211',
    name: 'Trent Alexander-Arnold',
    position: 'Defender',
    team: 'Liverpool',
    goals: 1,
    assists: 9
  },
  {
    _id: ObjectId('67731619f6963e617f9bfaae'),
    id: 'S214',
    name: 'Andrew Robertson',
    position: 'Defender',
    team: 'Liverpool',
    goals: 1,
    assists: 7
  }
]
```

2. Find the team with the highest total points in the league:



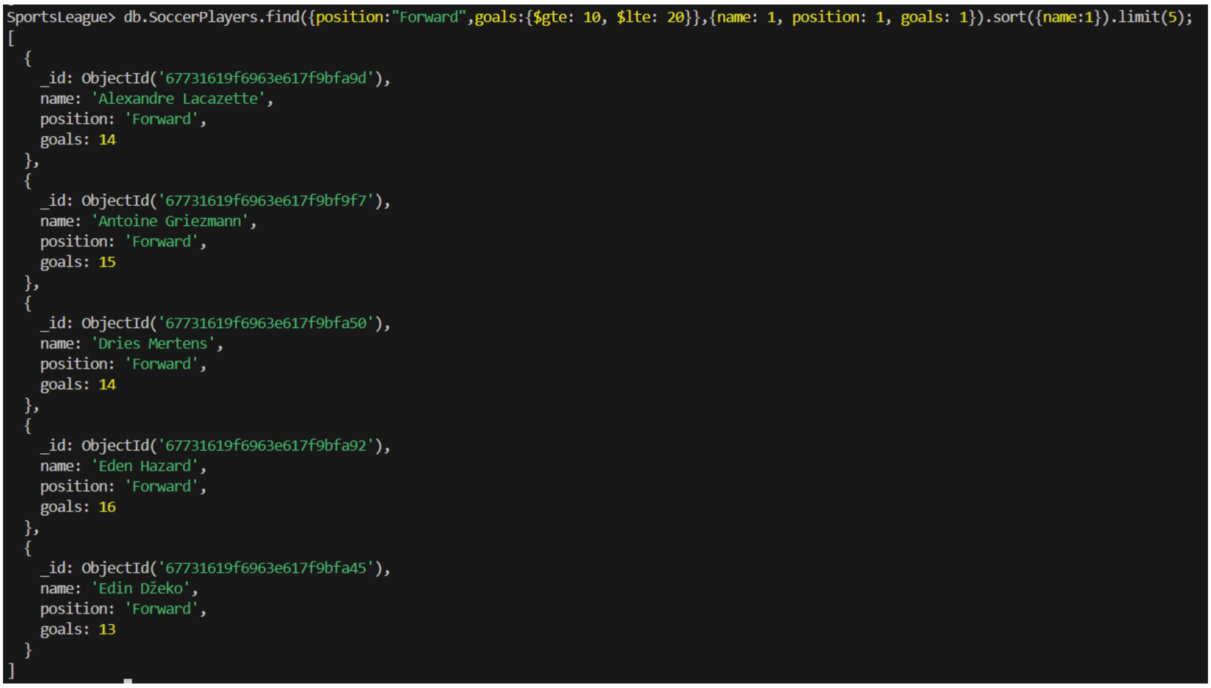
```
SportsLeague> db.Teams.find({}, {_id:0 ,team_name: 1, total_points: 1}).sort({ total_points: -1 }).limit(1);
[ { team_name: 'Paris Saint-Germain (PSG)', total_points: 90 } ]
```

3. Find top 5 goal scorers across all football players:



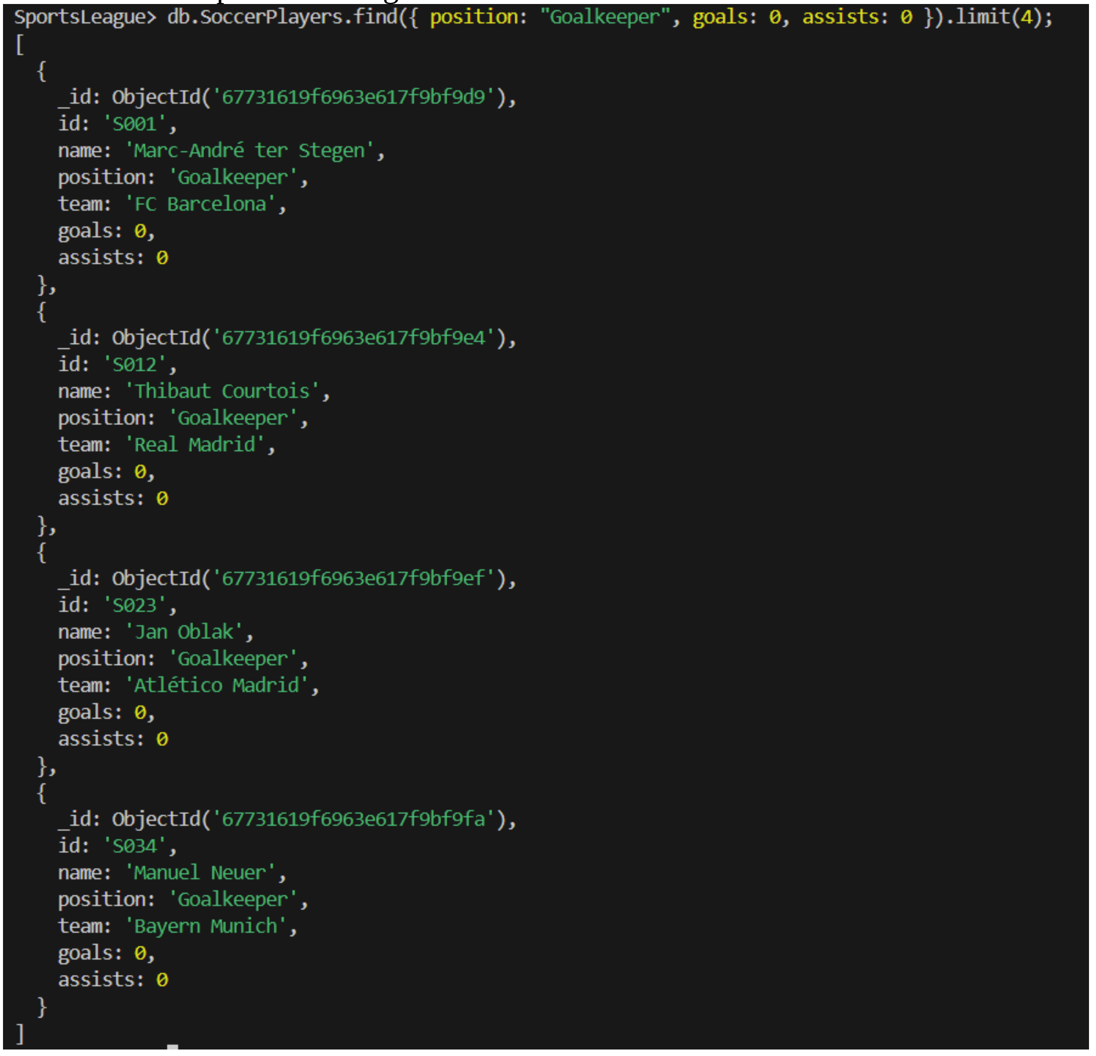
```
SportsLeague> db.SoccerPlayers.find({},{name:1, goals:1}).sort({ goals:-1 }).limit(5);
[
  {
    _id: ObjectId('67731619f6963e617f9bf9e1'),
    name: 'Lionel Messi',
    goals: 36
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa03'),
    name: 'Robert Lewandowski',
    goals: 34
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa5c'),
    name: 'Kylian Mbappé',
    goals: 30
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa25'),
    name: 'Cristiano Ronaldo',
    goals: 30
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa2f'),
    name: 'Mauro Icardi',
    goals: 28
  }
]
```

4. Find football players who are forwards and have scored between 10 and 20 goals, sorted alphabetically by name:



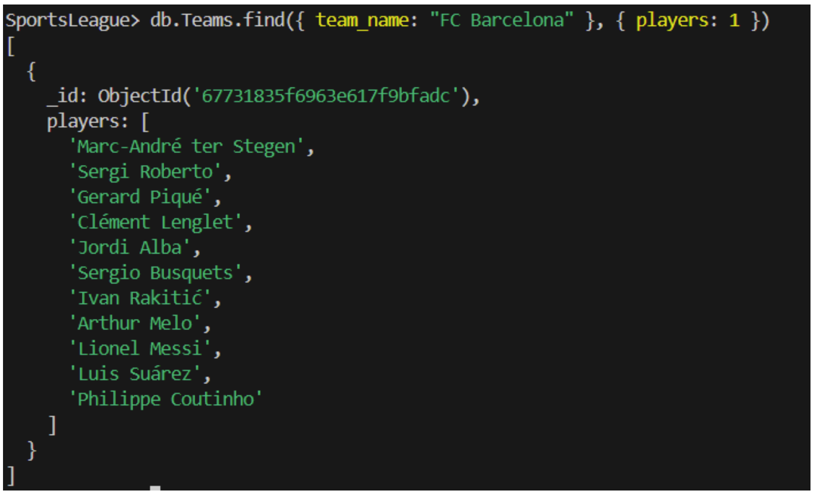
```
SportsLeague> db.SoccerPlayers.find({position:"Forward",goals:{$gte: 10, $lte: 20}},{name: 1, position: 1, goals: 1}).sort({name:1}).limit(5);
[
  {
    _id: ObjectId('67731619f6963e617f9bfa9d'),
    name: 'Alexandre Lacazette',
    position: 'Forward',
    goals: 14
  },
  {
    _id: ObjectId('67731619f6963e617f9bf9f7'),
    name: 'Antoine Griezmann',
    position: 'Forward',
    goals: 15
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa50'),
    name: 'Dries Mertens',
    position: 'Forward',
    goals: 14
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa92'),
    name: 'Eden Hazard',
    position: 'Forward',
    goals: 16
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa45'),
    name: 'Edin Džeko',
    position: 'Forward',
    goals: 13
  }
]
```

5. Find all Goalkeepers with no goals and no assists:



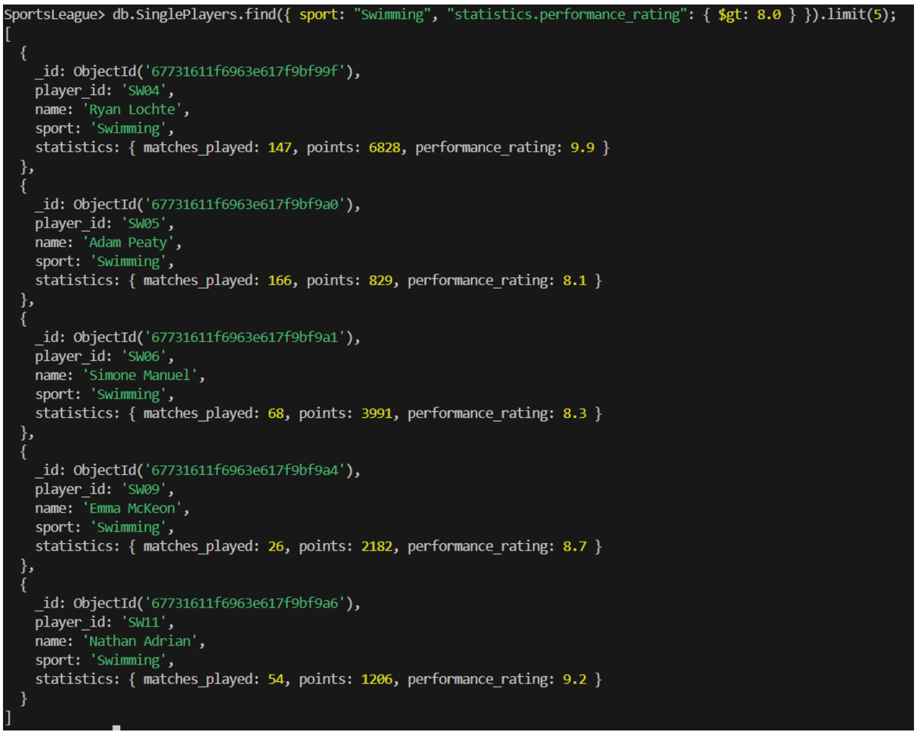
```
SportsLeague> db.SoccerPlayers.find({ position: "Goalkeeper", goals: 0, assists: 0 }).limit(4);
[
  {
    _id: ObjectId('67731619f6963e617f9bf9d9'),
    id: 'S001',
    name: 'Marc-André ter Stegen',
    position: 'Goalkeeper',
    team: 'FC Barcelona',
    goals: 0,
    assists: 0
  },
  {
    _id: ObjectId('67731619f6963e617f9bf9e4'),
    id: 'S012',
    name: 'Thibaut Courtois',
    position: 'Goalkeeper',
    team: 'Real Madrid',
    goals: 0,
    assists: 0
  },
  {
    _id: ObjectId('67731619f6963e617f9bf9ef'),
    id: 'S023',
    name: 'Jan Oblak',
    position: 'Goalkeeper',
    team: 'Atlético Madrid',
    goals: 0,
    assists: 0
  },
  {
    _id: ObjectId('67731619f6963e617f9bf9fa'),
    id: 'S034',
    name: 'Manuel Neuer',
    position: 'Goalkeeper',
    team: 'Bayern Munich',
    goals: 0,
    assists: 0
  }
]
```

6. Retrieve all players from a specific team:



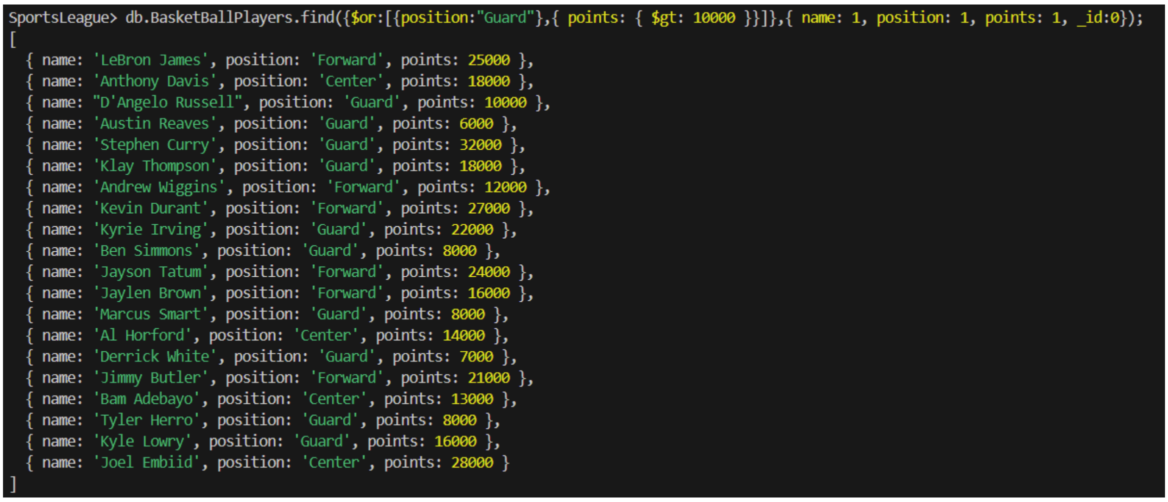
```
SportsLeague> db.Teams.find({ team_name: "FC Barcelona" }, { players: 1 })
[
  {
    _id: ObjectId('67731835f6963e617f9bfadc'),
    players: [
      'Marc-André ter Stegen',
      'Sergi Roberto',
      'Gerard Piqué',
      'Clément Lenglet',
      'Jordi Alba',
      'Sergio Busquets',
      'Ivan Rakitić',
      'Arthur Melo',
      'Lionel Messi',
      'Luis Suárez',
      'Philippe Coutinho'
    ]
  }
]
```

7. Find swimmers with a performance rating above 8.0:



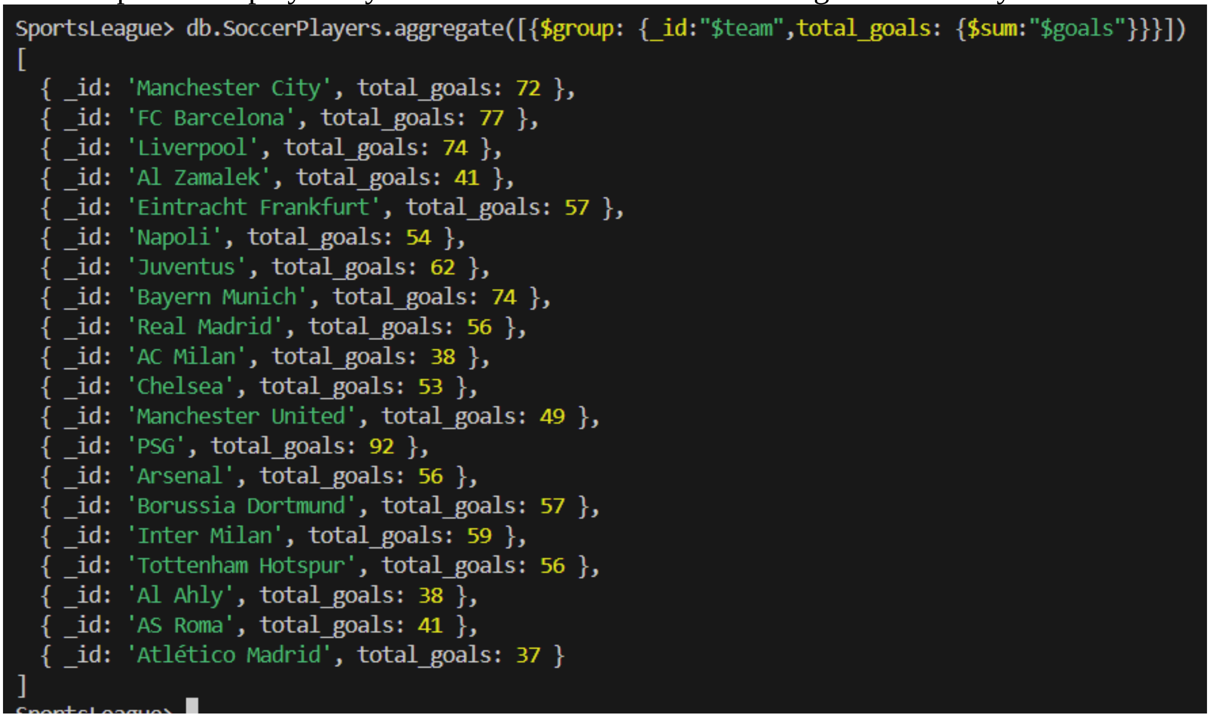
```
SportsLeague> db.SinglePlayers.find({ sport: "Swimming", "statistics.performance_rating": { $gt: 8.0 } }).limit(5);
[
  {
    _id: ObjectId('67731611f6963e617f9bf99f'),
    player_id: 'SW04',
    name: 'Ryan Lochte',
    sport: 'Swimming',
    statistics: { matches_played: 147, points: 6828, performance_rating: 9.9 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9a0'),
    player_id: 'SW05',
    name: 'Adam Peaty',
    sport: 'Swimming',
    statistics: { matches_played: 166, points: 829, performance_rating: 8.1 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9a1'),
    player_id: 'SW06',
    name: 'Simone Manuel',
    sport: 'Swimming',
    statistics: { matches_played: 68, points: 3991, performance_rating: 8.3 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9a4'),
    player_id: 'SW09',
    name: 'Emma McKeon',
    sport: 'Swimming',
    statistics: { matches_played: 26, points: 2182, performance_rating: 8.7 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9a6'),
    player_id: 'SW11',
    name: 'Nathan Adrian',
    sport: 'Swimming',
    statistics: { matches_played: 54, points: 1206, performance_rating: 9.2 }
  }
]
```

8. Retrieve basketball players who are either guards or have scored more than 10,000 points:



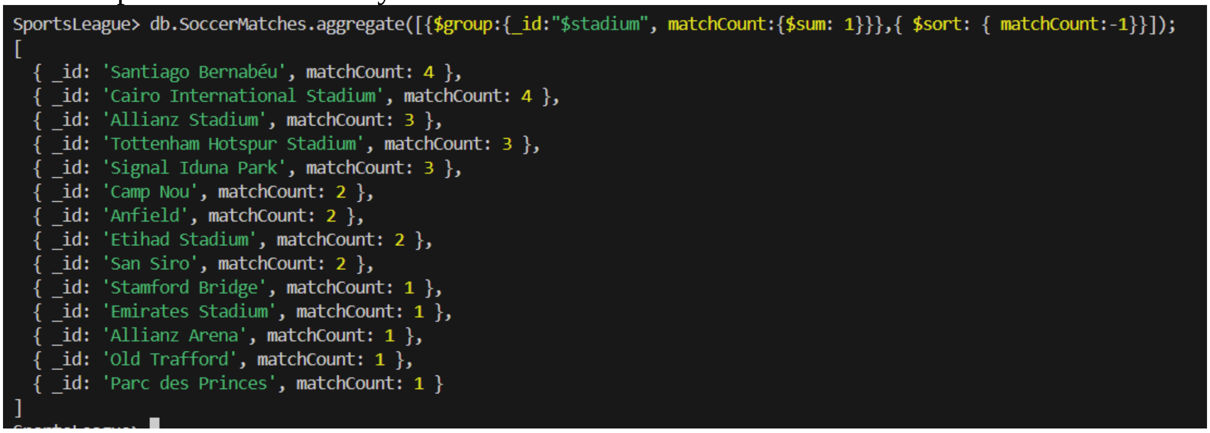
```
SportsLeague> db.BasketBallPlayers.find({$or:[{position:"Guard"},{ points: { $gt: 10000 }}]},{ name: 1, position: 1, points: 1, _id:0});
[
  { name: 'LeBron James', position: 'Forward', points: 25000 },
  { name: 'Anthony Davis', position: 'Center', points: 18000 },
  { name: "D'Angelo Russell", position: 'Guard', points: 10000 },
  { name: 'Austin Reaves', position: 'Guard', points: 6000 },
  { name: 'Stephen Curry', position: 'Guard', points: 32000 },
  { name: 'Klay Thompson', position: 'Guard', points: 18000 },
  { name: 'Andrew Wiggins', position: 'Forward', points: 12000 },
  { name: 'Kevin Durant', position: 'Forward', points: 27000 },
  { name: 'Kyrie Irving', position: 'Guard', points: 22000 },
  { name: 'Ben Simmons', position: 'Guard', points: 8000 },
  { name: 'Jayson Tatum', position: 'Forward', points: 24000 },
  { name: 'Jaylen Brown', position: 'Forward', points: 16000 },
  { name: 'Marcus Smart', position: 'Guard', points: 8000 },
  { name: 'Al Horford', position: 'Center', points: 14000 },
  { name: 'Derrick White', position: 'Guard', points: 7000 },
  { name: 'Jimmy Butler', position: 'Forward', points: 21000 },
  { name: 'Bam Adebayo', position: 'Center', points: 13000 },
  { name: 'Tyler Herro', position: 'Guard', points: 8000 },
  { name: 'Kyle Lowry', position: 'Guard', points: 16000 },
  { name: 'Joel Embiid', position: 'Center', points: 28000 }
]
```

9. Group football players by their team and calculate the total goals scored by each team:



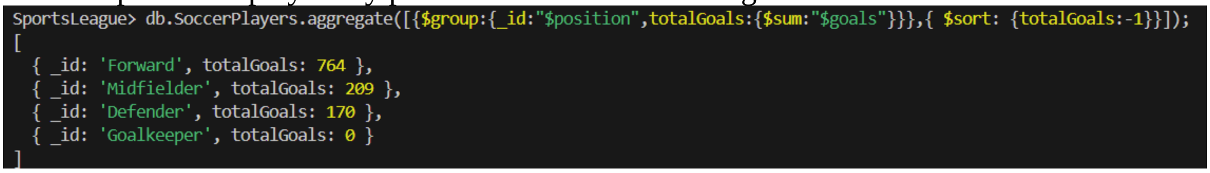
```
SportsLeague> db.SoccerPlayers.aggregate([{$group: {_id:"$team",total_goals: {$sum:"$goals"}}}])
[
  { _id: 'Manchester City', total_goals: 72 },
  { _id: 'FC Barcelona', total_goals: 77 },
  { _id: 'Liverpool', total_goals: 74 },
  { _id: 'Al Zamalek', total_goals: 41 },
  { _id: 'Eintracht Frankfurt', total_goals: 57 },
  { _id: 'Napoli', total_goals: 54 },
  { _id: 'Juventus', total_goals: 62 },
  { _id: 'Bayern Munich', total_goals: 74 },
  { _id: 'Real Madrid', total_goals: 56 },
  { _id: 'AC Milan', total_goals: 38 },
  { _id: 'Chelsea', total_goals: 53 },
  { _id: 'Manchester United', total_goals: 49 },
  { _id: 'PSG', total_goals: 92 },
  { _id: 'Arsenal', total_goals: 56 },
  { _id: 'Borussia Dortmund', total_goals: 57 },
  { _id: 'Inter Milan', total_goals: 59 },
  { _id: 'Tottenham Hotspur', total_goals: 56 },
  { _id: 'Al Ahly', total_goals: 38 },
  { _id: 'AS Roma', total_goals: 41 },
  { _id: 'Atlético Madrid', total_goals: 37 }
]
SportsLeague>
```

10. Group football matches by stadium and count matches:



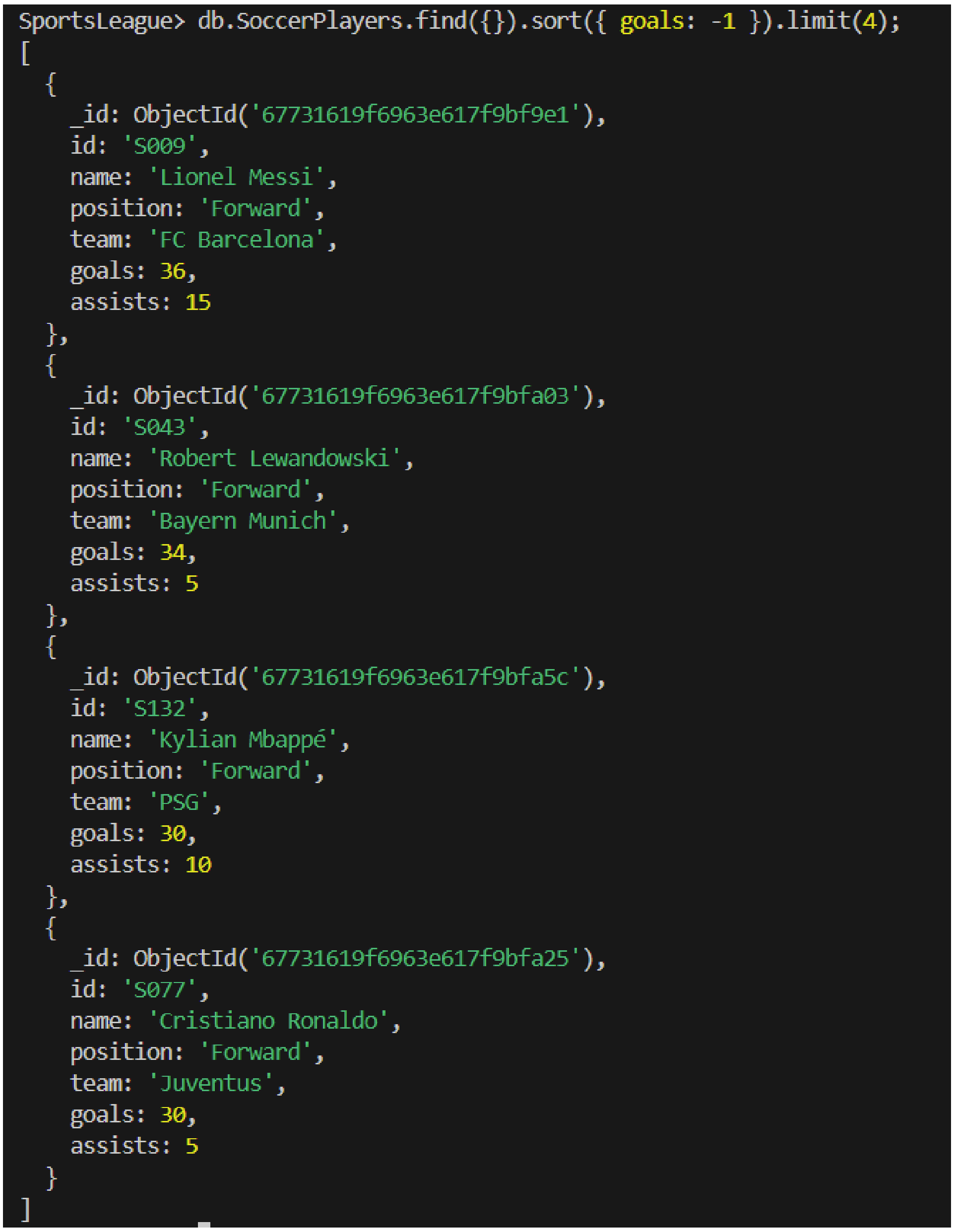
```
SportsLeague> db.SoccerMatches.aggregate([{$group:{_id:"$stadium", matchCount:{$sum: 1}}},{ $sort: { matchCount:-1}}]);
[
  { _id: 'Santiago Bernabéu', matchCount: 4 },
  { _id: 'Cairo International Stadium', matchCount: 4 },
  { _id: 'Allianz Stadium', matchCount: 3 },
  { _id: 'Tottenham Hotspur Stadium', matchCount: 3 },
  { _id: 'Signal Iduna Park', matchCount: 3 },
  { _id: 'Camp Nou', matchCount: 2 },
  { _id: 'Anfield', matchCount: 2 },
  { _id: 'Etihad Stadium', matchCount: 2 },
  { _id: 'San Siro', matchCount: 2 },
  { _id: 'Stamford Bridge', matchCount: 1 },
  { _id: 'Emirates Stadium', matchCount: 1 },
  { _id: 'Allianz Arena', matchCount: 1 },
  { _id: 'Old Trafford', matchCount: 1 },
  { _id: 'Parc des Princes', matchCount: 1 }
]
SportsLeague>
```

11. Group football players by position and calculate total goals for each:



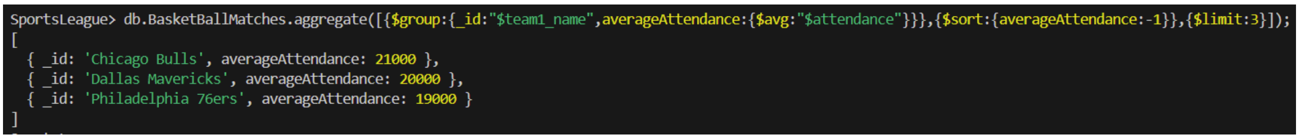
```
SportsLeague> db.SoccerPlayers.aggregate([{$group:{_id:"$position",totalGoals:{$sum:"$goals"}}},{ $sort: {totalGoals:-1}}]);
[
  { _id: 'Forward', totalGoals: 764 },
  { _id: 'Midfielder', totalGoals: 209 },
  { _id: 'Defender', totalGoals: 170 },
  { _id: 'Goalkeeper', totalGoals: 0 }
]
```

12. Find the top 4 football players by goals:



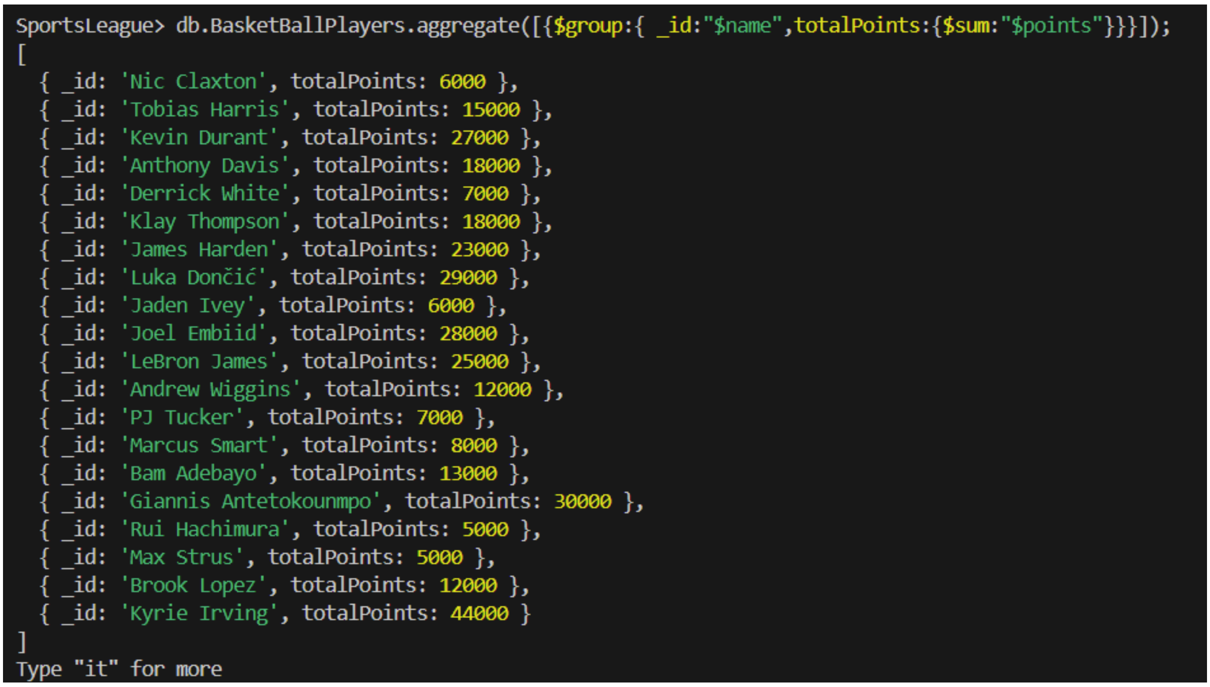
```
SportsLeague> db.SoccerPlayers.find({}).sort({ goals: -1 }).limit(4);
[
  {
    _id: ObjectId('67731619f6963e617f9bf9e1'),
    id: 'S009',
    name: 'Lionel Messi',
    position: 'Forward',
    team: 'FC Barcelona',
    goals: 36,
    assists: 15
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa03'),
    id: 'S043',
    name: 'Robert Lewandowski',
    position: 'Forward',
    team: 'Bayern Munich',
    goals: 34,
    assists: 5
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa5c'),
    id: 'S132',
    name: 'Kylian Mbappé',
    position: 'Forward',
    team: 'PSG',
    goals: 30,
    assists: 10
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa25'),
    id: 'S077',
    name: 'Cristiano Ronaldo',
    position: 'Forward',
    team: 'Juventus',
    goals: 30,
    assists: 5
  }
]
```

13. List the top 3 basketball teams by average attendance in their matches:



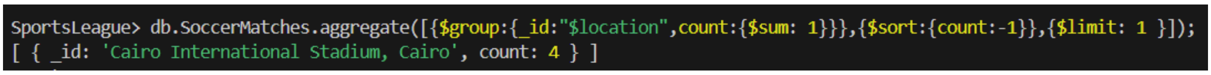
```
SportsLeague> db.BasketBallMatches.aggregate([{$group:{_id:"$team1_name",averageAttendance:{$avg:"$attendance"}}},{$sort:{averageAttendance:-1}},{$limit:3}]);
[
  { _id: 'Chicago Bulls', averageAttendance: 21000 },
  { _id: 'Dallas Mavericks', averageAttendance: 20000 },
  { _id: 'Philadelphia 76ers', averageAttendance: 19000 }
]
```

14. Calculate total points scored by each basketball player:



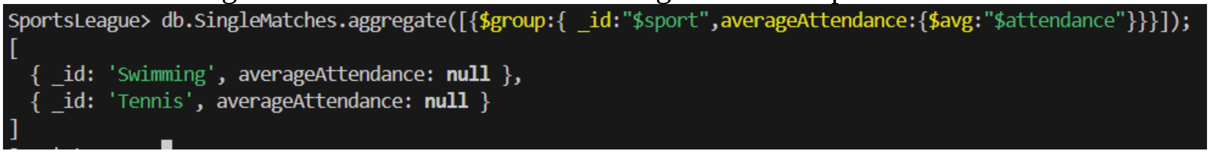
```
SportsLeague> db.BasketBallPlayers.aggregate([{$group:{ _id:"$name",totalPoints:{$sum:"$points"}}}]);
[
  { _id: 'Nic Claxton', totalPoints: 6000 },
  { _id: 'Tobias Harris', totalPoints: 15000 },
  { _id: 'Kevin Durant', totalPoints: 27000 },
  { _id: 'Anthony Davis', totalPoints: 18000 },
  { _id: 'Derrick White', totalPoints: 7000 },
  { _id: 'Klay Thompson', totalPoints: 18000 },
  { _id: 'James Harden', totalPoints: 23000 },
  { _id: 'Luka Dončić', totalPoints: 29000 },
  { _id: 'Jaden Ivey', totalPoints: 6000 },
  { _id: 'Joel Embiid', totalPoints: 28000 },
  { _id: 'LeBron James', totalPoints: 25000 },
  { _id: 'Andrew Wiggins', totalPoints: 12000 },
  { _id: 'PJ Tucker', totalPoints: 7000 },
  { _id: 'Marcus Smart', totalPoints: 8000 },
  { _id: 'Bam Adebayo', totalPoints: 13000 },
  { _id: 'Giannis Antetokounmpo', totalPoints: 30000 },
  { _id: 'Rui Hachimura', totalPoints: 5000 },
  { _id: 'Max Strus', totalPoints: 5000 },
  { _id: 'Brook Lopez', totalPoints: 12000 },
  { _id: 'Kyrie Irving', totalPoints: 44000 }
]
Type "it" for more
```

15. Find the most frequent match location in football:



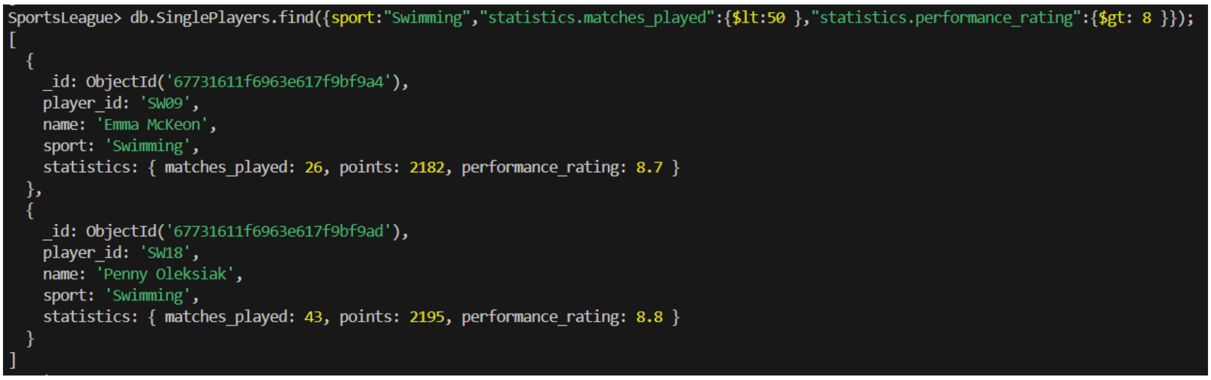
```
SportsLeague> db.SoccerMatches.aggregate([{$group:{_id:"$location",count:{$sum: 1}}},{$sort:{count:-1}},{$limit: 1 }]);
[ { _id: 'Cairo International Stadium, Cairo', count: 4 } ]
```

16. Get the average match attendance for swimming and tennis sports:



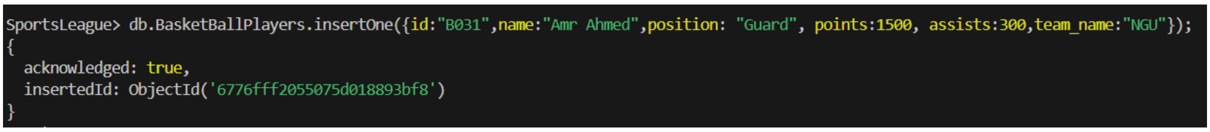
```
SportsLeague> db.SingleMatches.aggregate([{$group:{ _id:"$sport",averageAttendance:{$avg:"$attendance"}}}]);
[
  { _id: 'Swimming', averageAttendance: null },
  { _id: 'Tennis', averageAttendance: null }
]
```

17. Find swimming players who participated in fewer than 50 matches but have a high-performance rating above 8:



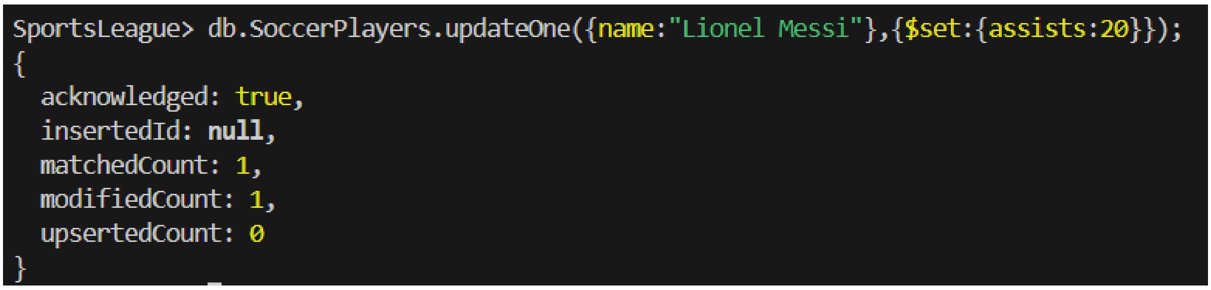
```
SportsLeague> db.SinglePlayers.find({sport:"Swimming","statistics.matches_played":{$lt:50 },"statistics.performance_rating":{$gt: 8 }});
[
  {
    _id: ObjectId('67731611f6963e617f9bf9a4'),
    player_id: 'SW09',
    name: 'Emma McKeon',
    sport: 'Swimming',
    statistics: { matches_played: 26, points: 2182, performance_rating: 8.7 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9ad'),
    player_id: 'SW18',
    name: 'Penny Oleksiak',
    sport: 'Swimming',
    statistics: { matches_played: 43, points: 2195, performance_rating: 8.8 }
  }
]
```

18. Insert a new basketball player into the BasketBallPlayers collection:



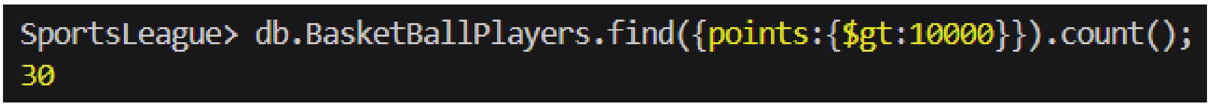
```
SportsLeague> db.BasketBallPlayers.insertOne({id:"B031",name:"Amr Ahmed",position: "Guard", points:1500, assists:300,team_name:"NGU"});
{
  acknowledged: true,
  insertedId: ObjectId('6776fff2055075d018893bf8')
}
```

19. Update the assists of "Lionel Messi" in the SoccerPlayers collection:



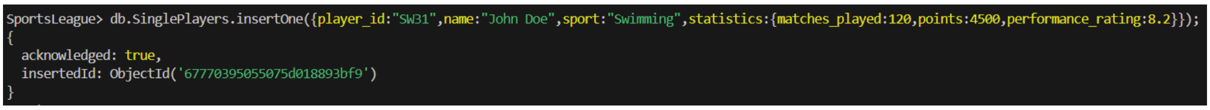
```
SportsLeague> db.SoccerPlayers.updateOne({name:"Lionel Messi"},{$set:{assists:20}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

20. Count the total number of basketball players who have scored more than 10,000 points:



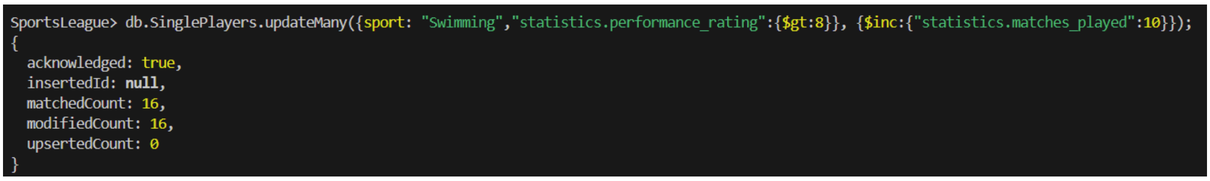
```
SportsLeague> db.BasketBallPlayers.find({points:{$gt:10000}}).count();
30
```

21. Insert a new swimmer:



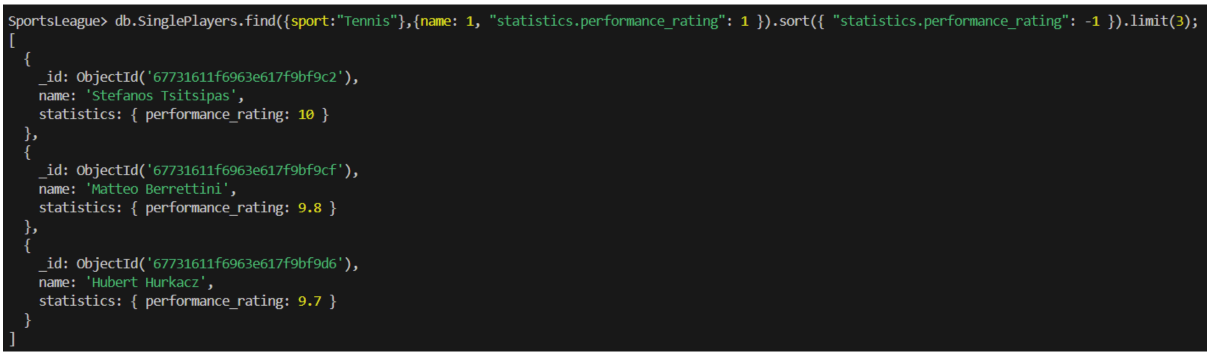
```
SportsLeague> db.SinglePlayers.insertOne({player_id:"SW31",name:"John Doe",sport:"Swimming",statistics:{matches_played:120,points:4500,performance_rating:8.2}});
{
  acknowledged: true,
  insertedId: ObjectId('67770395055075d018893bf9')
}
```

22. Increase the matches_played by 10 for all swimming players with a performance rating greater than 8:



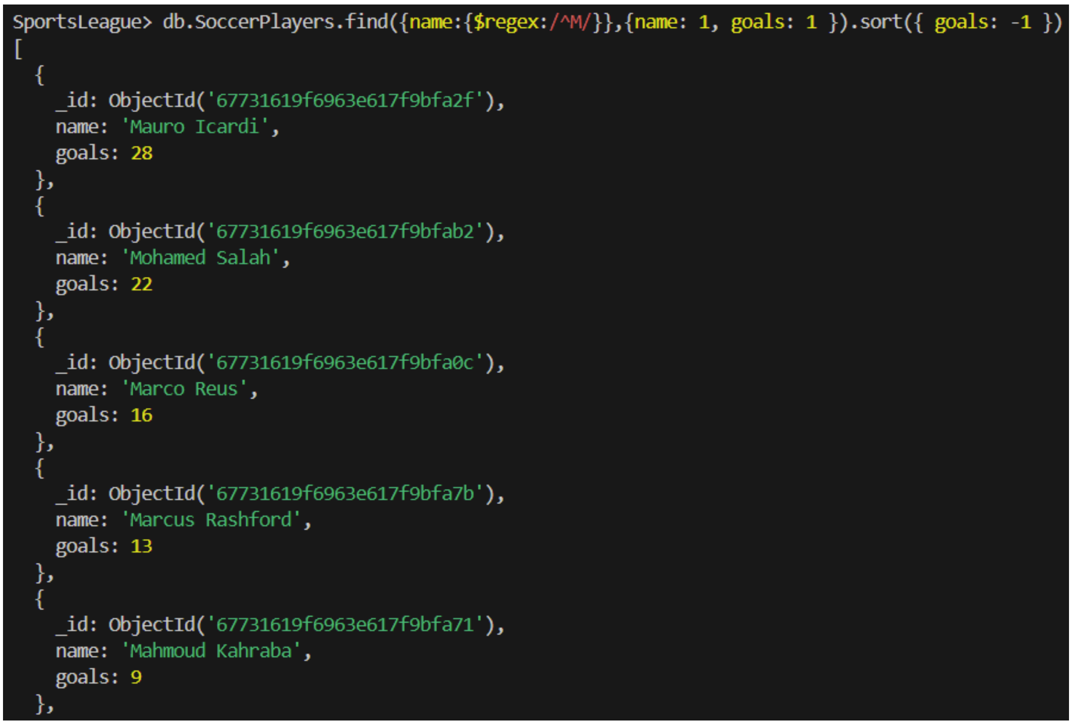
```
SportsLeague> db.SinglePlayers.updateMany({sport: "Swimming","statistics.performance_rating":{$gt:8}}, {$inc:{"statistics.matches_played":10}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 16,
  modifiedCount: 16,
  upsertedCount: 0
}
```

23. Find the top 3 tennis players with the highest performance rating:



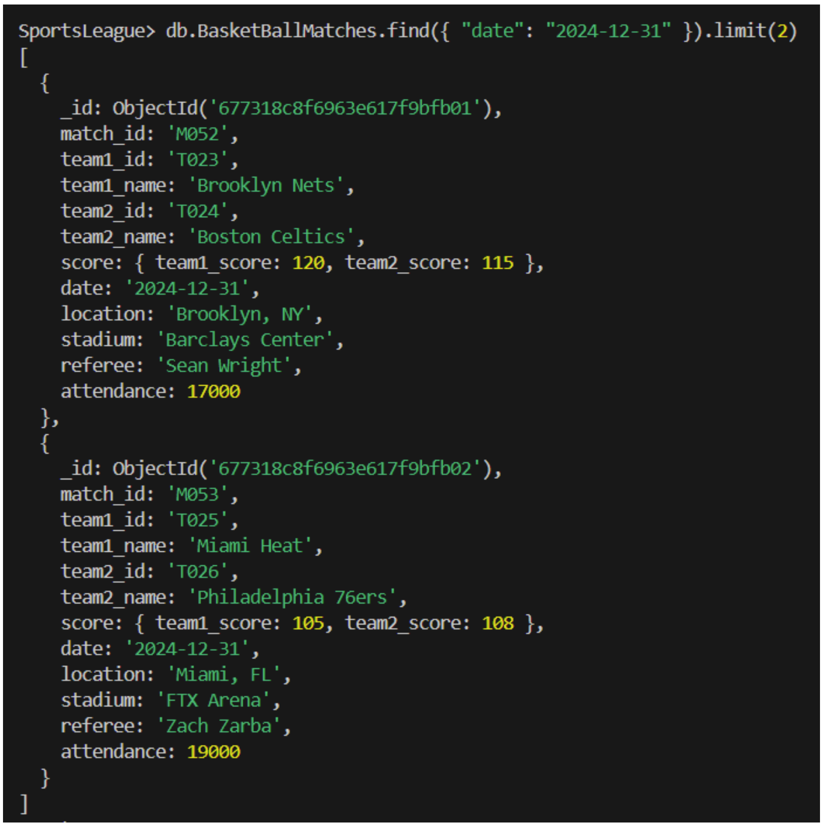
```
SportsLeague> db.SinglePlayers.find({sport:"Tennis"},{name: 1, "statistics.performance_rating": 1 }).sort({ "statistics.performance_rating": -1 }).limit(3);
[
  {
    _id: ObjectId('67731611f6963e617f9bf9c2'),
    name: 'Stefanos Tsitsipas',
    statistics: { performance_rating: 10 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9cf'),
    name: 'Matteo Berrettini',
    statistics: { performance_rating: 9.8 }
  },
  {
    _id: ObjectId('67731611f6963e617f9bf9d6'),
    name: 'Hubert Hurkacz',
    statistics: { performance_rating: 9.7 }
  }
]
```

24. Find players with names starting with "M" and display their name and goals, sorted by goals in descending order:



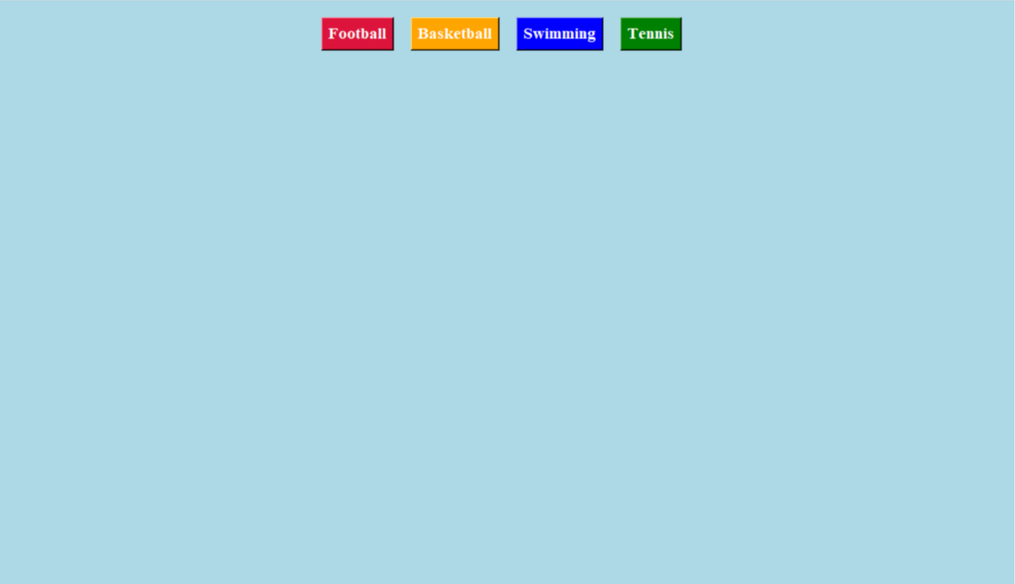
```
SportsLeague> db.SoccerPlayers.find({name:{$regex:/^M/}},{name: 1, goals: 1 }).sort({ goals: -1 })
[
  {
    _id: ObjectId('67731619f6963e617f9bfa2f'),
    name: 'Mauro Icardi',
    goals: 28
  },
  {
    _id: ObjectId('67731619f6963e617f9bfab2'),
    name: 'Mohamed Salah',
    goals: 22
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa0c'),
    name: 'Marco Reus',
    goals: 16
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa7b'),
    name: 'Marcus Rashford',
    goals: 13
  },
  {
    _id: ObjectId('67731619f6963e617f9bfa71'),
    name: 'Mahmoud Kahraba',
    goals: 9
  },
```

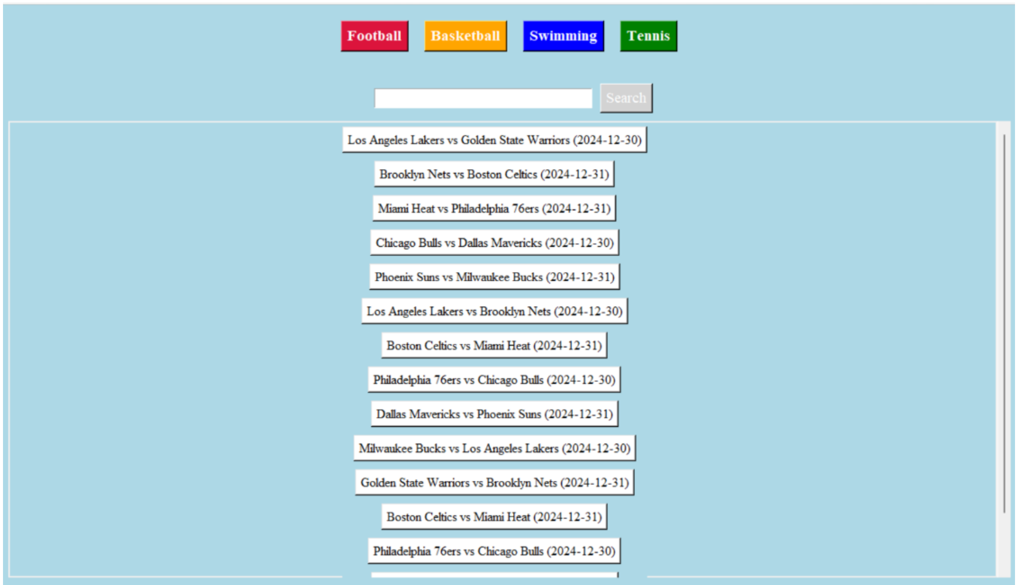
25. Find 2 matches held on "2024-12-31":



```
SportsLeague> db.BasketBallMatches.find({ "date": "2024-12-31" }).limit(2)
[
  {
    _id: ObjectId('677318c8f6963e617f9bfb01'),
    match_id: 'M052',
    team1_id: 'T023',
    team1_name: 'Brooklyn Nets',
    team2_id: 'T024',
    team2_name: 'Boston Celtics',
    score: { team1_score: 120, team2_score: 115 },
    date: '2024-12-31',
    location: 'Brooklyn, NY',
    stadium: 'Barclays Center',
    referee: 'Sean Wright',
    attendance: 17000
  },
  {
    _id: ObjectId('677318c8f6963e617f9bfb02'),
    match_id: 'M053',
    team1_id: 'T025',
    team1_name: 'Miami Heat',
    team2_id: 'T026',
    team2_name: 'Philadelphia 76ers',
    score: { team1_score: 105, team2_score: 108 },
    date: '2024-12-31',
    location: 'Miami, FL',
    stadium: 'FTX Arena',
    referee: 'Zach Zarba',
    attendance: 19000
  }
]
```

2.3 Graphical User Interface





Los Angeles Lakers vs Golden State Warriors (2024-12-30)

Brooklyn Nets vs Boston Celtics (2024-12-31)

Miami Heat vs Philadelphia 76ers (2024-12-31)

Chicago Bulls vs Dallas Mavericks (2024-12-30)

Phoenix Suns vs Milwaukee Bucks (2024-12-31)

Los Angeles Lakers vs Brooklyn Nets (2024-12-30)

Boston Celtics vs Miami Heat (2024-12-31)

Philadelphia 76ers vs Chicago Bulls (2024-12-30)

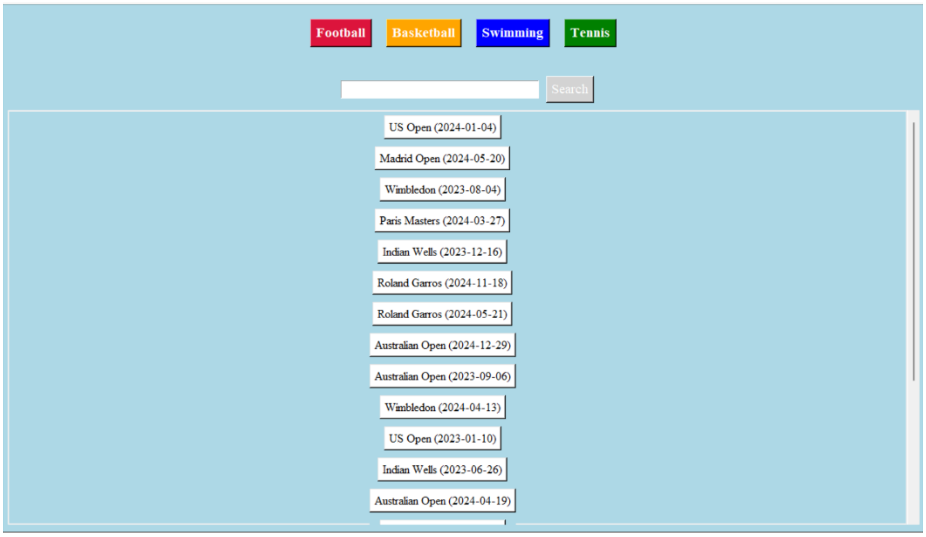
Dallas Mavericks vs Phoenix Suns (2024-12-31)

Milwaukee Bucks vs Los Angeles Lakers (2024-12-30)

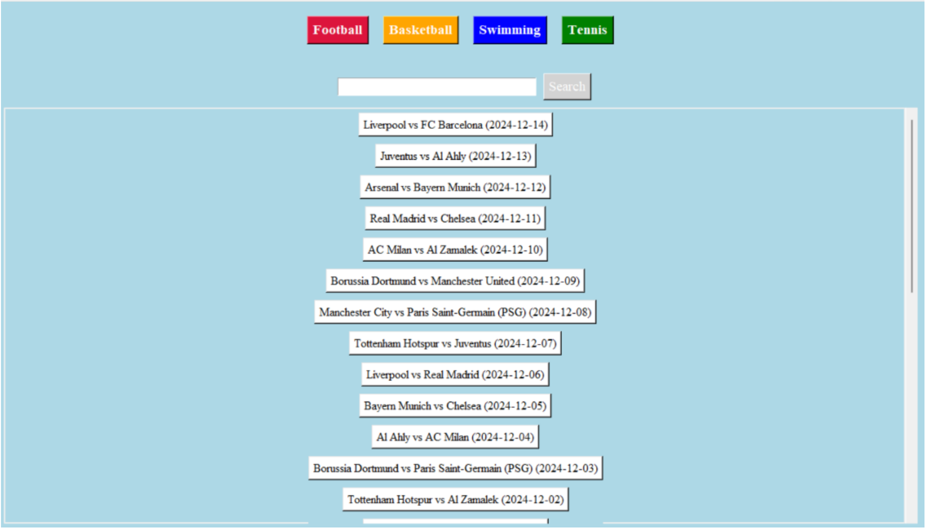
Golden State Warriors vs Brooklyn Nets (2024-12-31)

Boston Celtics vs Miami Heat (2024-12-31)

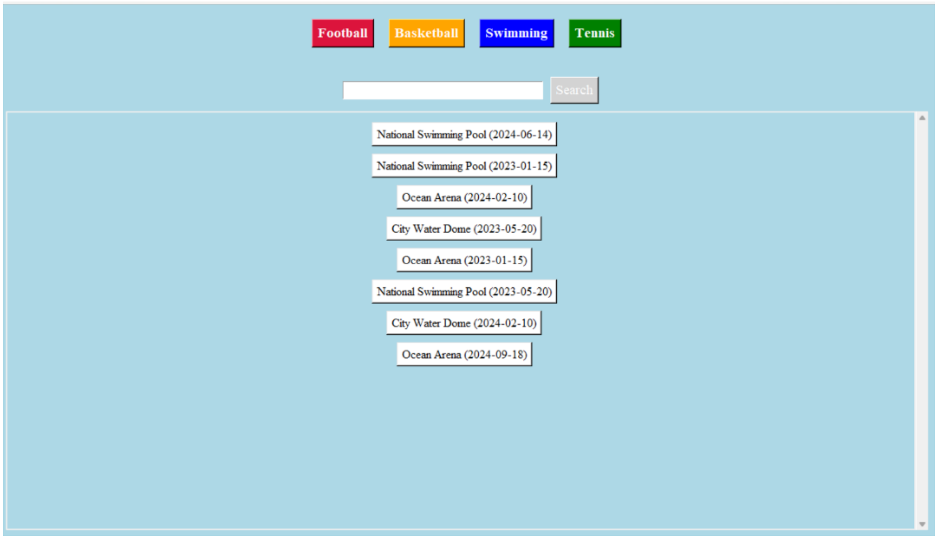
Philadelphia 76ers vs Chicago Bulls (2024-12-30)



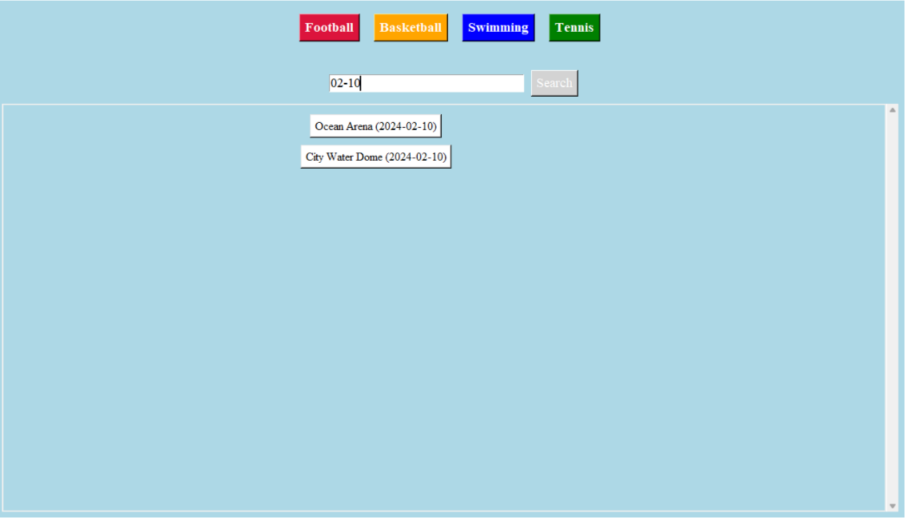
US Open (2024-01-04)
Madrid Open (2024-05-20)
Wimbledon (2023-08-04)
Paris Masters (2024-03-27)
Indian Wells (2023-12-16)
Roland Garros (2024-11-18)
Roland Garros (2024-05-21)
Australian Open (2024-12-29)
Australian Open (2023-09-06)
Wimbledon (2024-04-13)
US Open (2023-01-10)
Indian Wells (2023-06-26)
Australian Open (2024-04-19)



Liverpool vs FC Barcelona (2024-12-14)
Juventus vs Al Ahly (2024-12-13)
Arsenal vs Bayern Munich (2024-12-12)
Real Madrid vs Chelsea (2024-12-11)
AC Milan vs Al Zamalek (2024-12-10)
Borussia Dortmund vs Manchester United (2024-12-09)
Manchester City vs Paris Saint-Germain (PSG) (2024-12-08)
Tottenham Hotspur vs Juventus (2024-12-07)
Liverpool vs Real Madrid (2024-12-06)
Bayern Munich vs Chelsea (2024-12-05)
Al Ahly vs AC Milan (2024-12-04)
Borussia Dortmund vs Paris Saint-Germain (PSG) (2024-12-03)
Tottenham Hotspur vs Al Zamalek (2024-12-02)



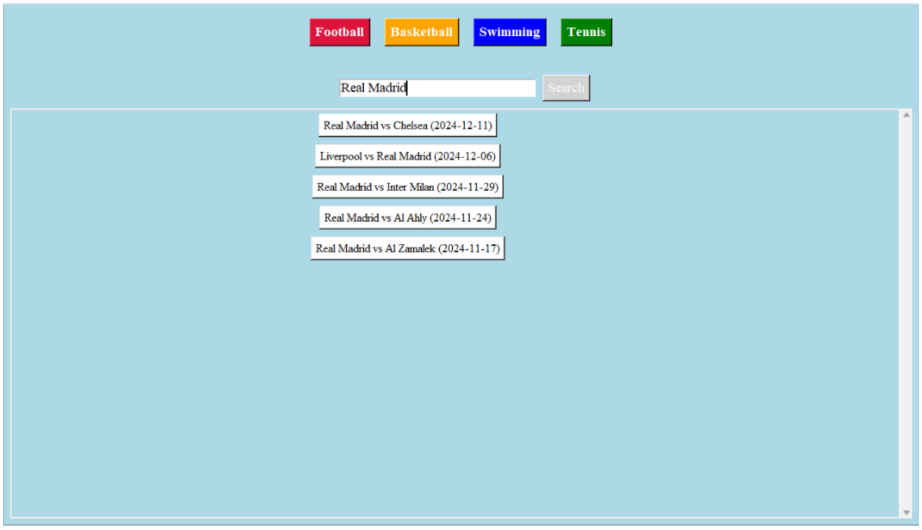
National Swimming Pool (2024-06-14)
National Swimming Pool (2023-01-15)
Ocean Arena (2024-02-10)
City Water Dome (2023-05-20)
Ocean Arena (2023-01-15)
National Swimming Pool (2023-05-20)
City Water Dome (2024-02-10)
Ocean Arena (2024-09-18)



02-10

Ocean Arena (2024-02-10)

City Water Dome (2024-02-10)



Real Madrid

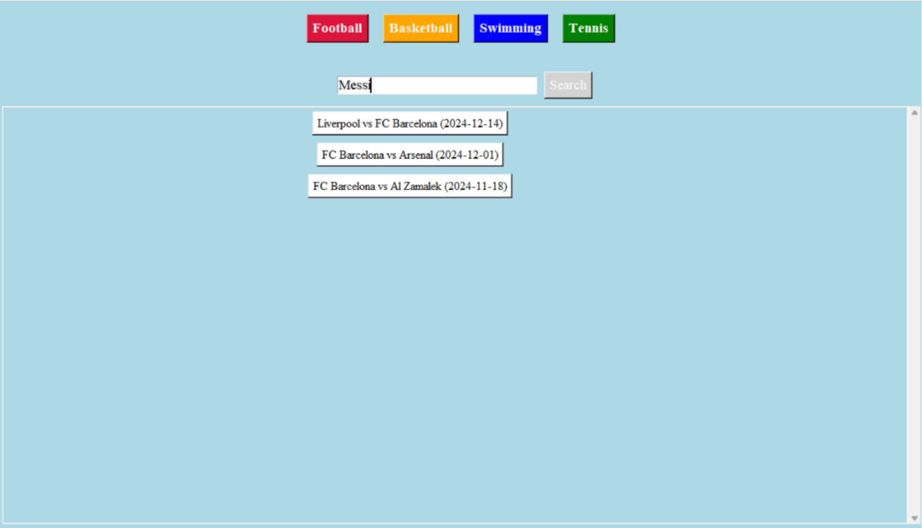
Real Madrid vs Chelsea (2024-12-11)

Liverpool vs Real Madrid (2024-12-06)

Real Madrid vs Inter Milan (2024-11-29)

Real Madrid vs Al Ahly (2024-11-24)

Real Madrid vs Al Zamalek (2024-11-17)

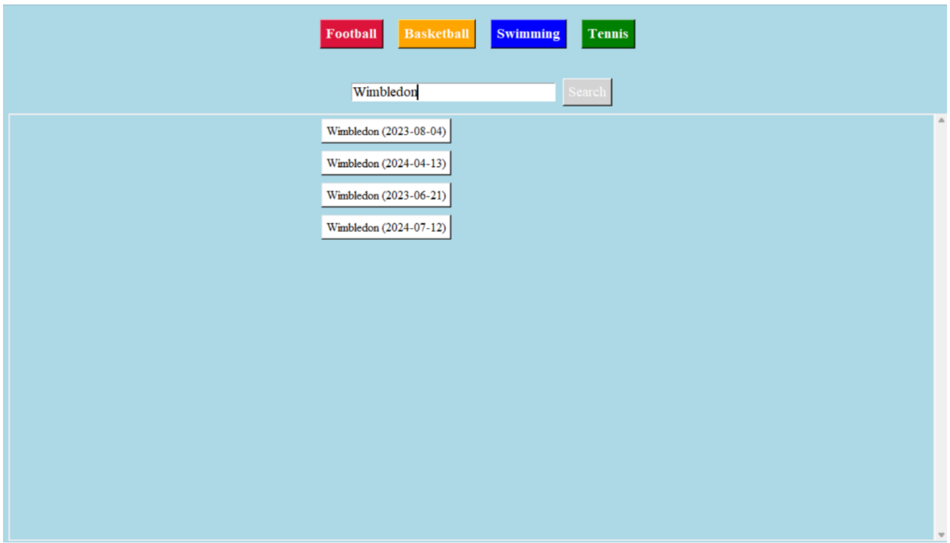


Messi

Liverpool vs FC Barcelona (2024-12-14)

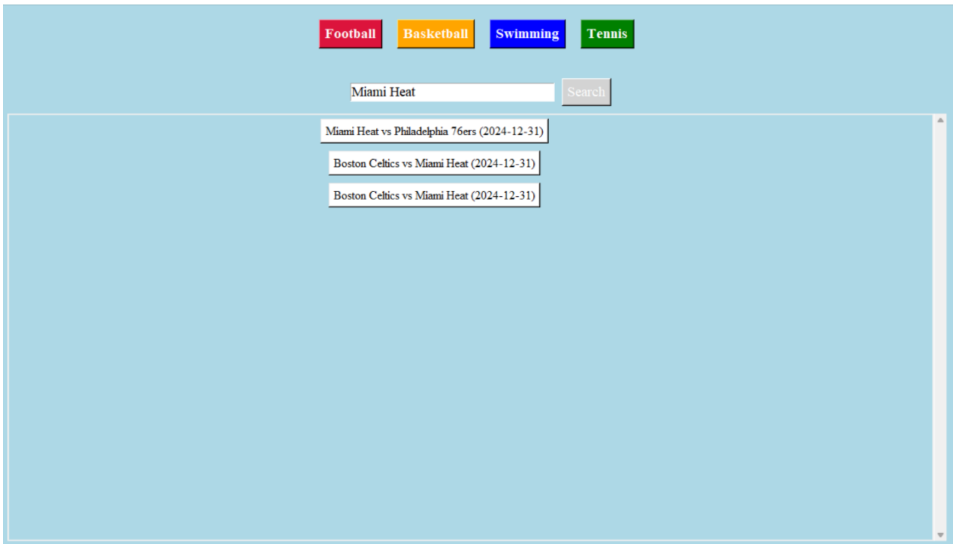
FC Barcelona vs Arsenal (2024-12-01)

FC Barcelona vs Al Zamalek (2024-11-18)



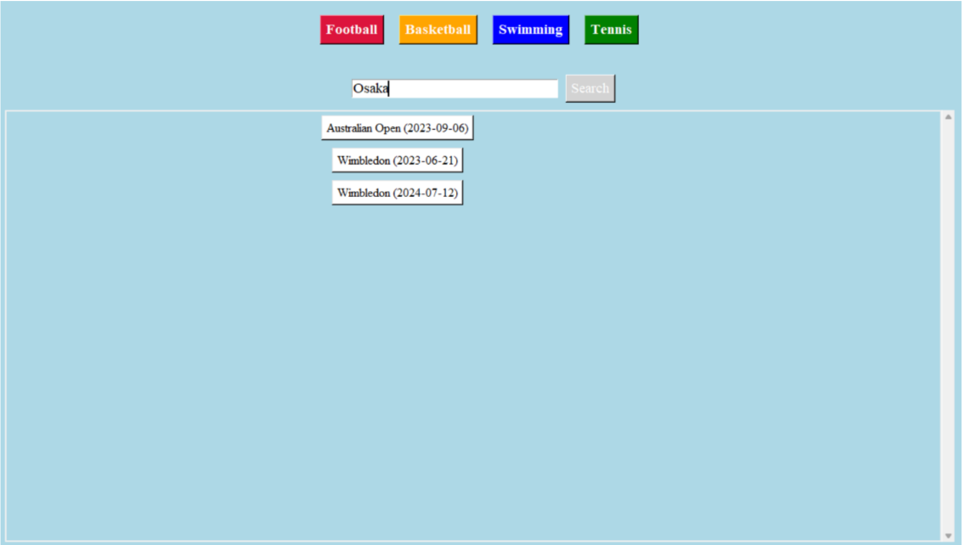
Wimbledon

Wimbledon (2023-08-04)
Wimbledon (2024-04-13)
Wimbledon (2023-06-21)
Wimbledon (2024-07-12)



Miami Heat

Miami Heat vs Philadelphia 76ers (2024-12-31)
Boston Celtics vs Miami Heat (2024-12-31)
Boston Celtics vs Miami Heat (2024-12-31)



Osaka

Australian Open (2023-09-06)
Wimbledon (2023-06-21)
Wimbledon (2024-07-12)