REPORT ON BIG DATA MANAGEMENT – MONGO DB

ASSIGNMENT 2

PAVAN BABU BHEESETTI

1002207463

Pxb7463@mavs.uta.edu

1. Introduction

This project explores the use of MongoDB, a document-oriented NoSQL database, for storing and querying complex objects. The objective is to migrate data from flat relational files to MongoDB's document-based format, reflecting the structure and relationships relevant to the Soccer World Cup. The project emphasizes the differences between relational and document-oriented approaches and demonstrates query operations specific to NoSQL systems.

2. Environment Setup

Database System: MongoDB

Programming Language: Python (used for data extraction, transformation, and loading)

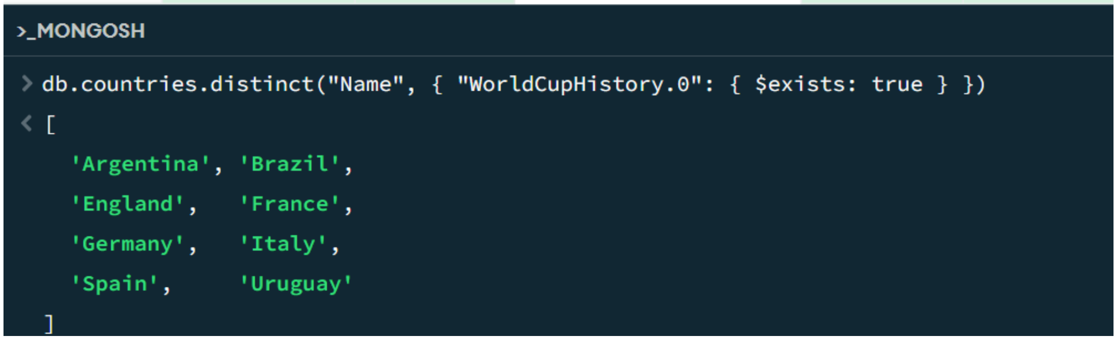
Tools Used: MongoDB Compass (for database visualization), pymongo library (for MongoDB integration with Python)

Data Input: Flat relational files containing information about countries, players, stadiums, and matches.

The Python Files are uploaded in the Zip file as countries and Stadiums

3. MongoDB Queries and Results:

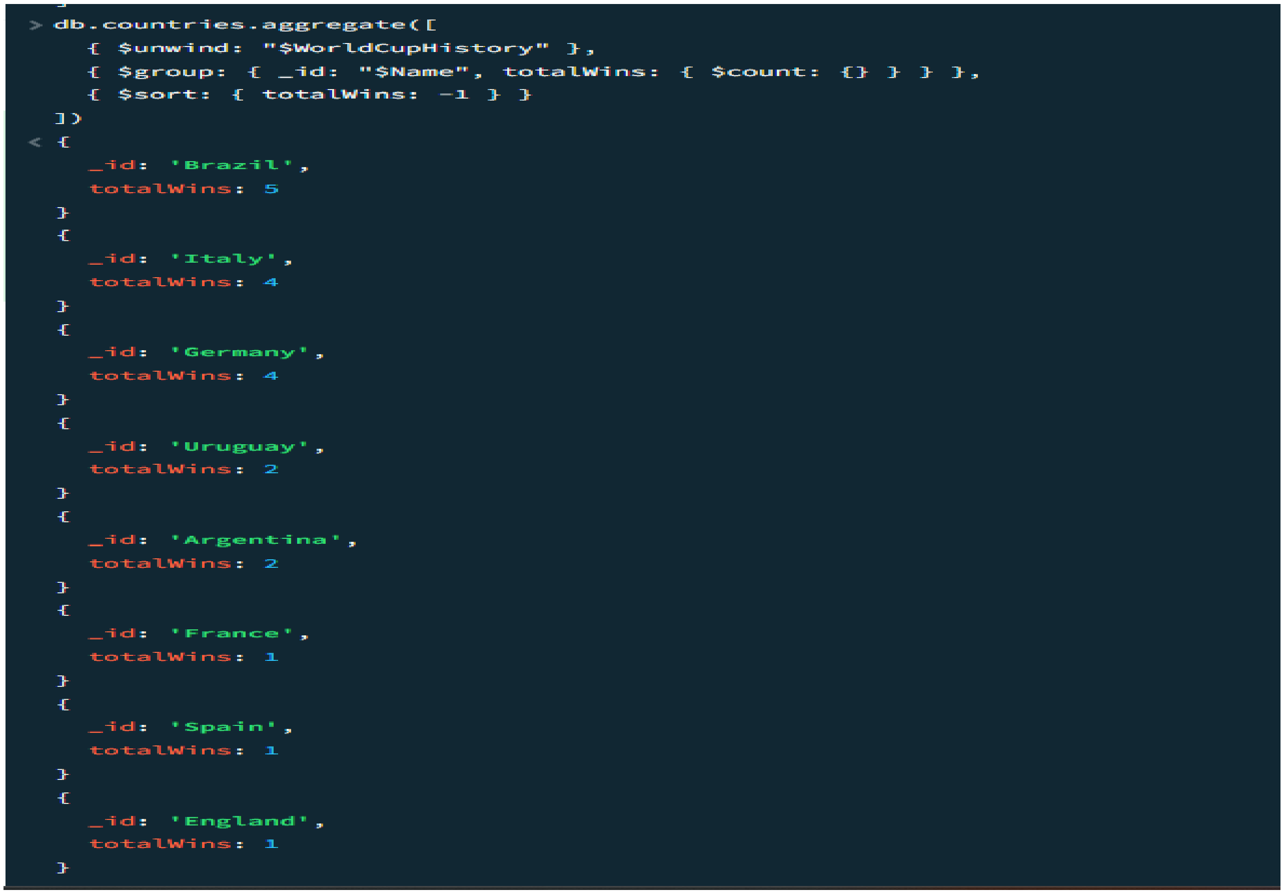
Retrieve the list of country names that have won a world cup:



>_MONGOSH

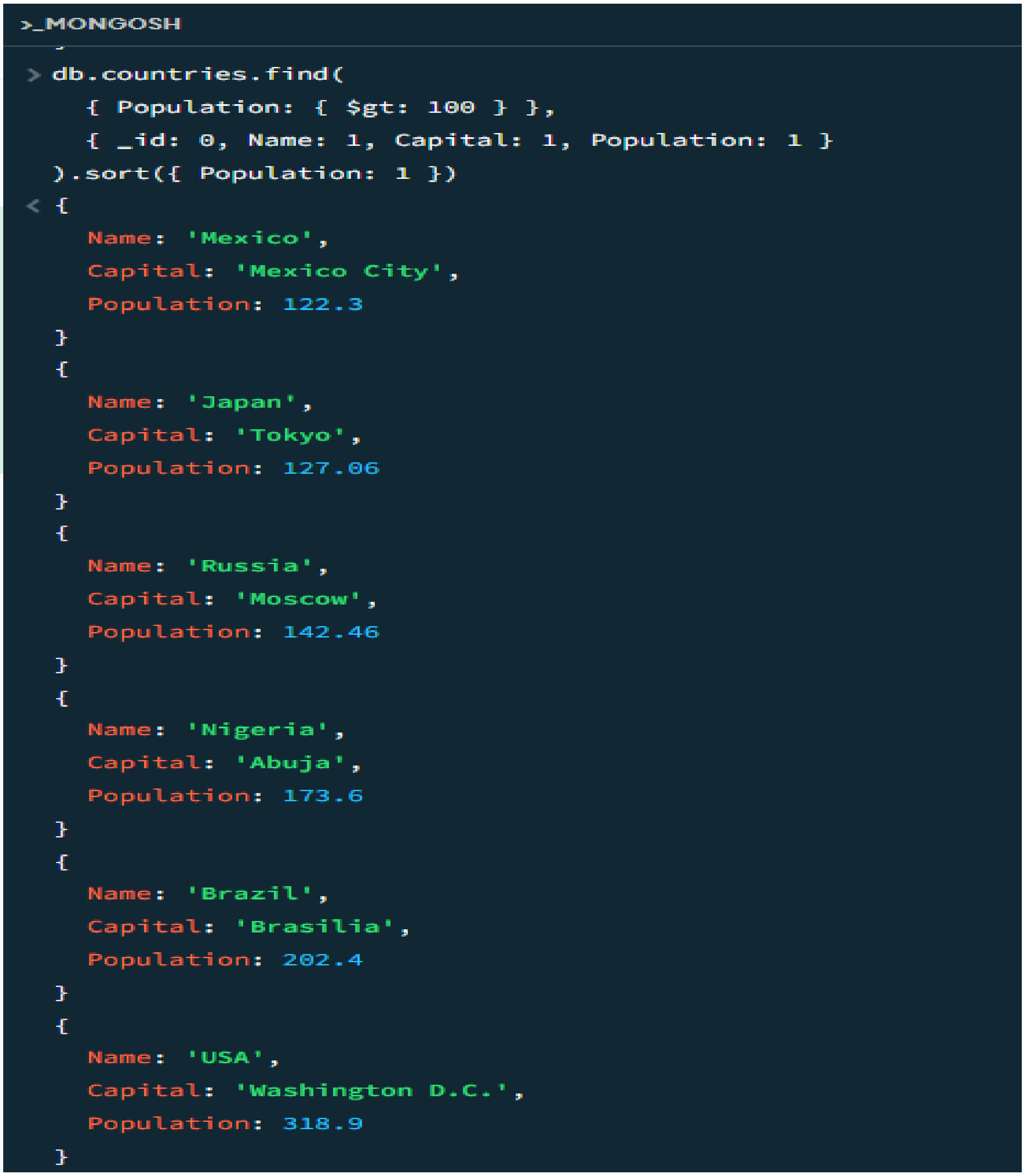
```
> db.countries.distinct("Name", { "WorldCupHistory.0": { $exists: true } })
< [
    'Argentina', 'Brazil',
    'England',   'France',
    'Germany',   'Italy',
    'Spain',     'Uruguay'
  ]
```

Retrieve the list of country names that have won a world cup and the number of wins in descending order:



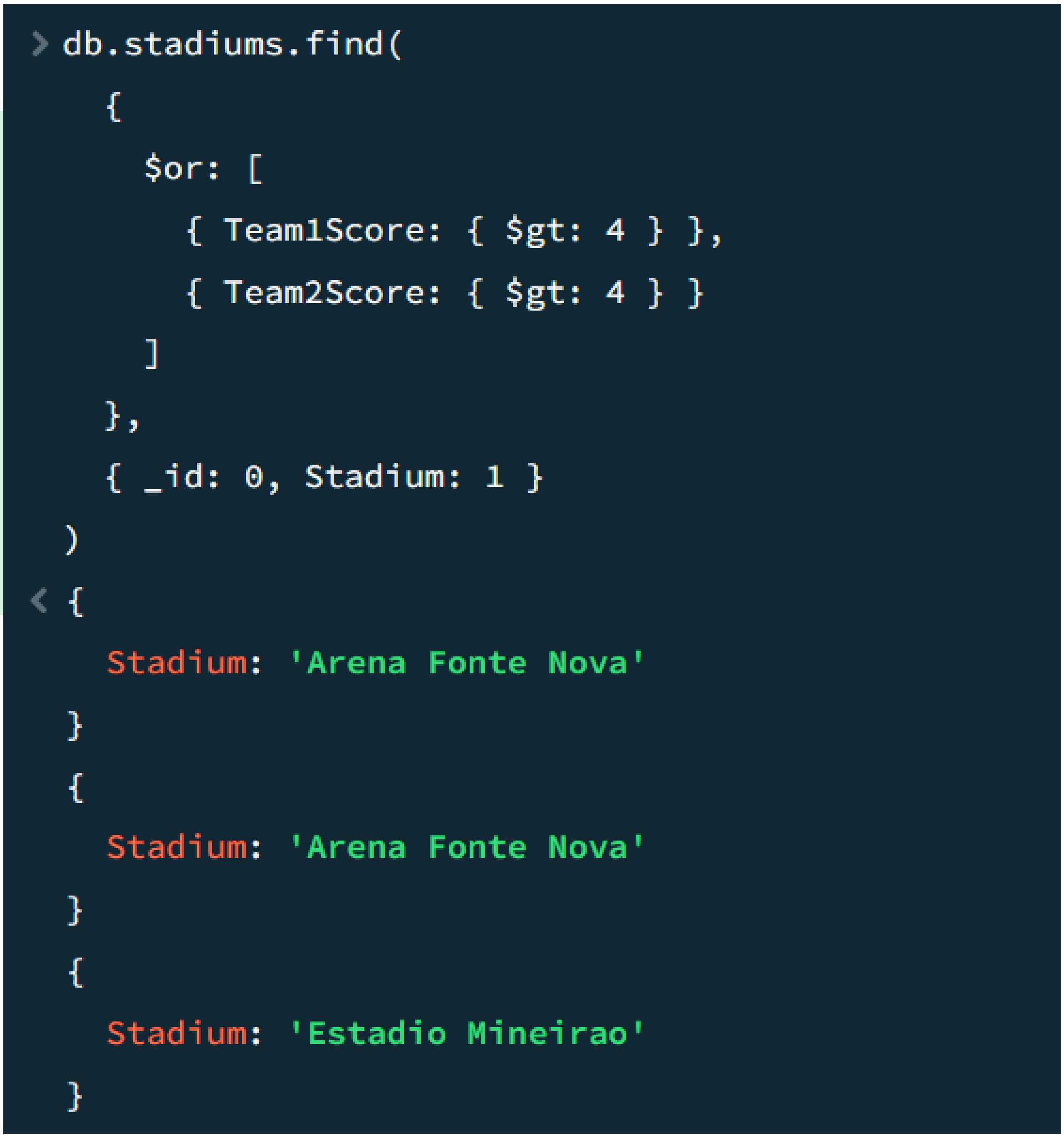
```
> db.countries.aggregate([
    { $unwind: "$WorldCupHistory" },
    { $group: { _id: "$Name", totalWins: { $count: {} } } },
    { $sort: { totalWins: -1 } }
  ])
< {
    _id: 'Brazil',
    totalWins: 5
  }
  {
    _id: 'Italy',
    totalWins: 4
  }
  {
    _id: 'Germany',
    totalWins: 4
  }
  {
    _id: 'Uruguay',
    totalWins: 2
  }
  {
    _id: 'Argentina',
    totalWins: 2
  }
  {
    _id: 'France',
    totalWins: 1
  }
  {
    _id: 'Spain',
    totalWins: 1
  }
  {
    _id: 'England',
    totalWins: 1
  }
```

List the capital cities in increasing order of population for countries with more than 100 million population:



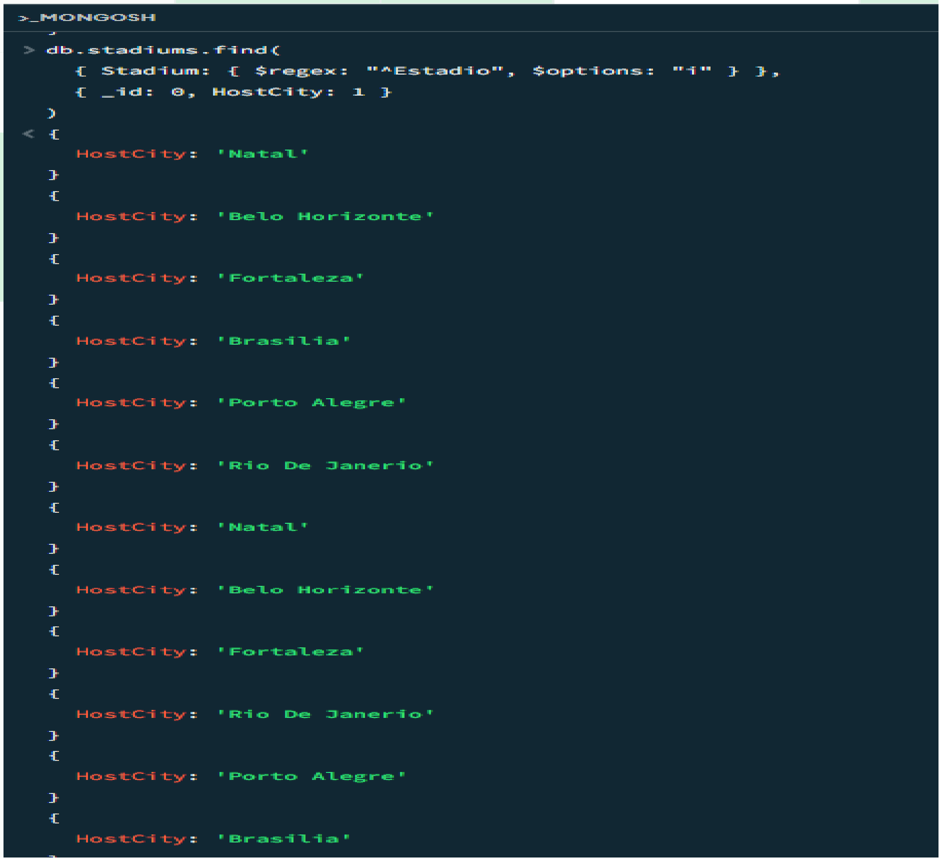
```
> db.countries.find(
    { Population: { $gt: 100 } },
    { _id: 0, Name: 1, Capital: 1, Population: 1 }
  ).sort({ Population: 1 })
< {
    Name: 'Mexico',
    Capital: 'Mexico City',
    Population: 122.3
  }
  {
    Name: 'Japan',
    Capital: 'Tokyo',
    Population: 127.06
  }
  {
    Name: 'Russia',
    Capital: 'Moscow',
    Population: 142.46
  }
  {
    Name: 'Nigeria',
    Capital: 'Abuja',
    Population: 173.6
  }
  {
    Name: 'Brazil',
    Capital: 'Brasilia',
    Population: 202.4
  }
  {
    Name: 'USA',
    Capital: 'Washington D.C.',
    Population: 318.9
  }
```

List the stadiums hosting matches where a team scored more than 4 goals:

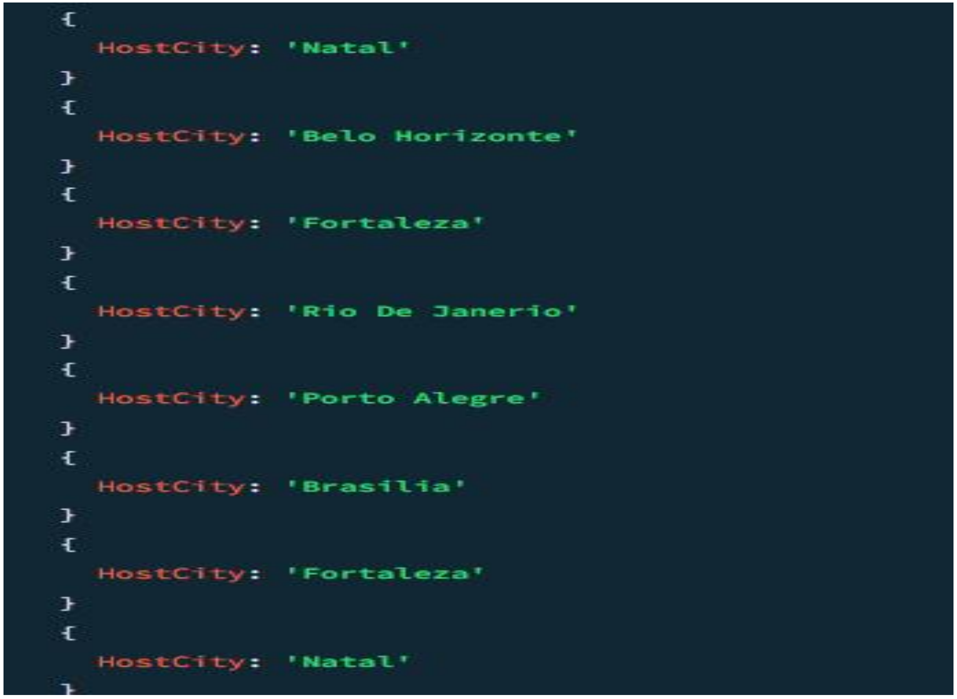


```
> db.stadiums.find(
    {
      $or: [
        { Team1Score: { $gt: 4 } },
        { Team2Score: { $gt: 4 } }
      ]
    },
    { _id: 0, Stadium: 1 }
  )
< {
    Stadium: 'Arena Fonte Nova'
  }
  {
    Stadium: 'Arena Fonte Nova'
  }
  {
    Stadium: 'Estadio Mineirao'
  }
```

List cities with stadiums starting with "Estadio":

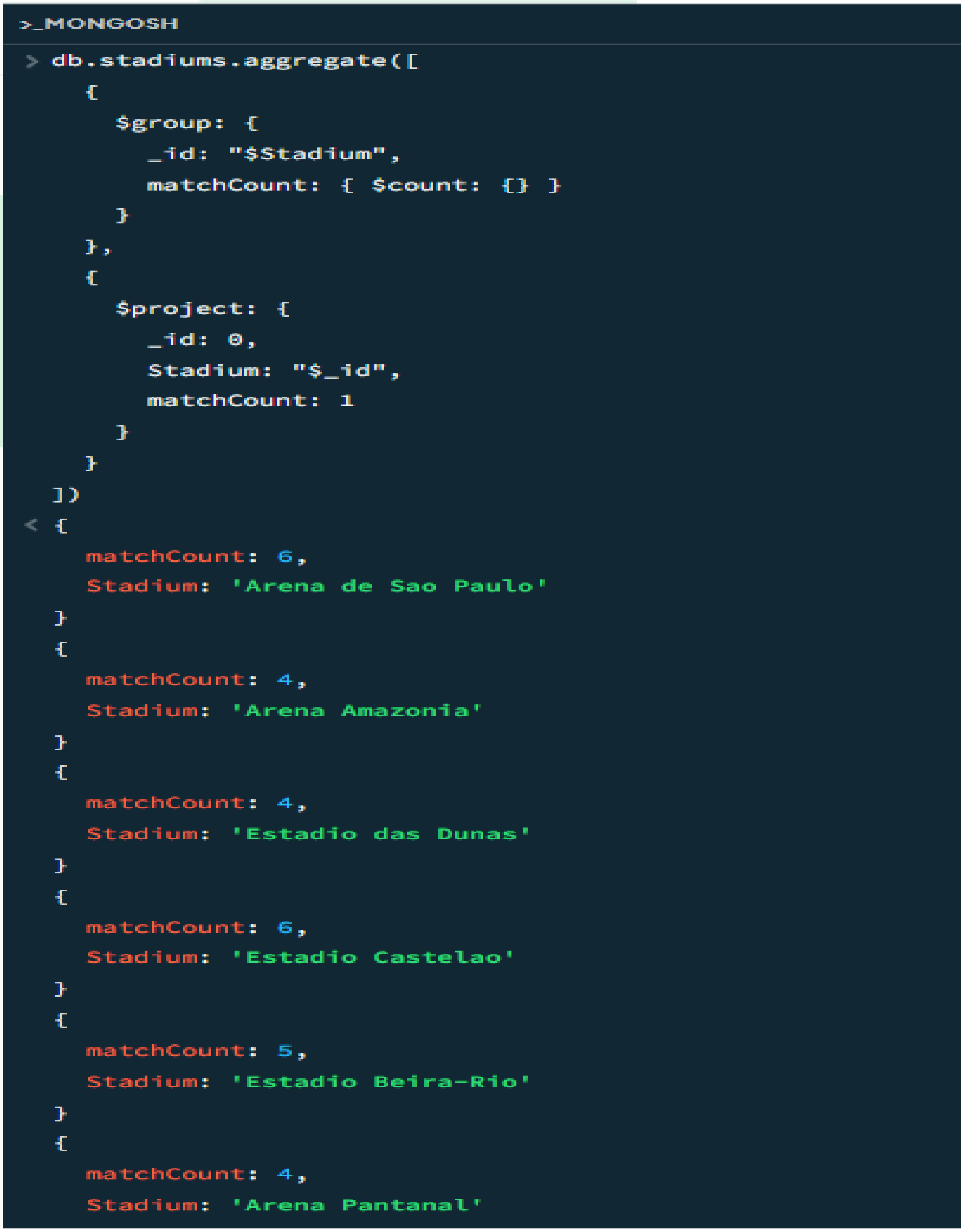


```
> db.stadiums.find(
    { Stadium: { $regex: "^Estadio", $options: "i" } },
    { _id: 0, HostCity: 1 }
  )
< {
    HostCity: 'Natal'
  }
  {
    HostCity: 'Belo Horizonte'
  }
  {
    HostCity: 'Fortaleza'
  }
  {
    HostCity: 'Brasilia'
  }
  {
    HostCity: 'Porto Alegre'
  }
  {
    HostCity: 'Rio De Janerio'
  }
  {
    HostCity: 'Natal'
  }
  {
    HostCity: 'Belo Horizonte'
  }
  {
    HostCity: 'Fortaleza'
  }
  {
    HostCity: 'Rio De Janerio'
  }
  {
    HostCity: 'Porto Alegre'
  }
  {
    HostCity: 'Brasilia'
```

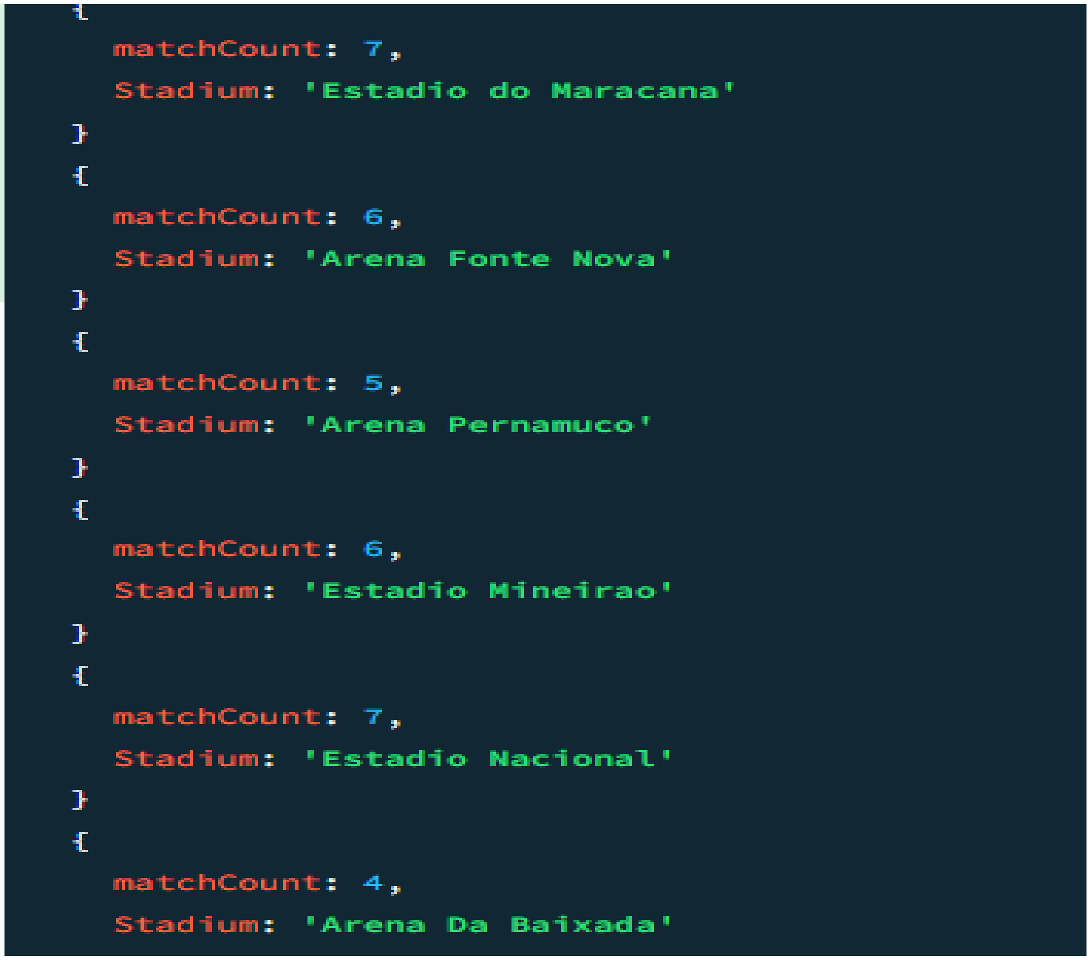


```
  {
    HostCity: 'Natal'
  }
  {
    HostCity: 'Belo Horizonte'
  }
  {
    HostCity: 'Fortaleza'
  }
  {
    HostCity: 'Rio De Janerio'
  }
  {
    HostCity: 'Porto Alegre'
  }
  {
    HostCity: 'Brasilia'
  }
  {
    HostCity: 'Fortaleza'
  }
  {
    HostCity: 'Natal'
  }
```

List all stadiums and the number of matches hosted by each:

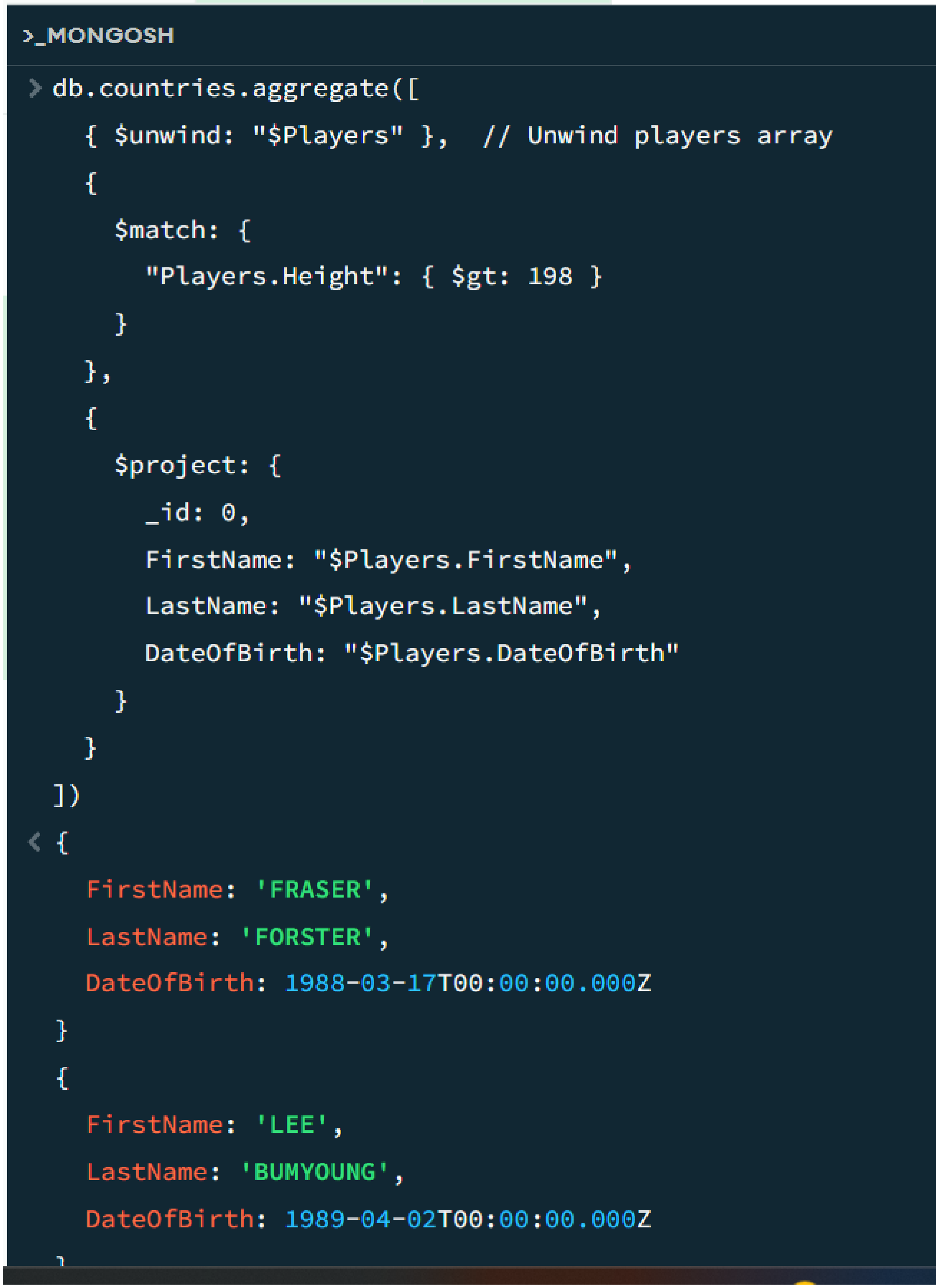


```
> db.stadiums.aggregate([
    {
      $group: {
        _id: "$Stadium",
        matchCount: { $count: {} }
      }
    },
    {
      $project: {
        _id: 0,
        Stadium: "$_id",
        matchCount: 1
      }
    }
  ])
< {
    matchCount: 6,
    Stadium: 'Arena de Sao Paulo'
  }
  {
    matchCount: 4,
    Stadium: 'Arena Amazonia'
  }
  {
    matchCount: 4,
    Stadium: 'Estadio das Dunas'
  }
  {
    matchCount: 6,
    Stadium: 'Estadio Castelao'
  }
  {
    matchCount: 5,
    Stadium: 'Estadio Beira-Rio'
  }
  {
    matchCount: 4,
    Stadium: 'Arena Pantanal'
```



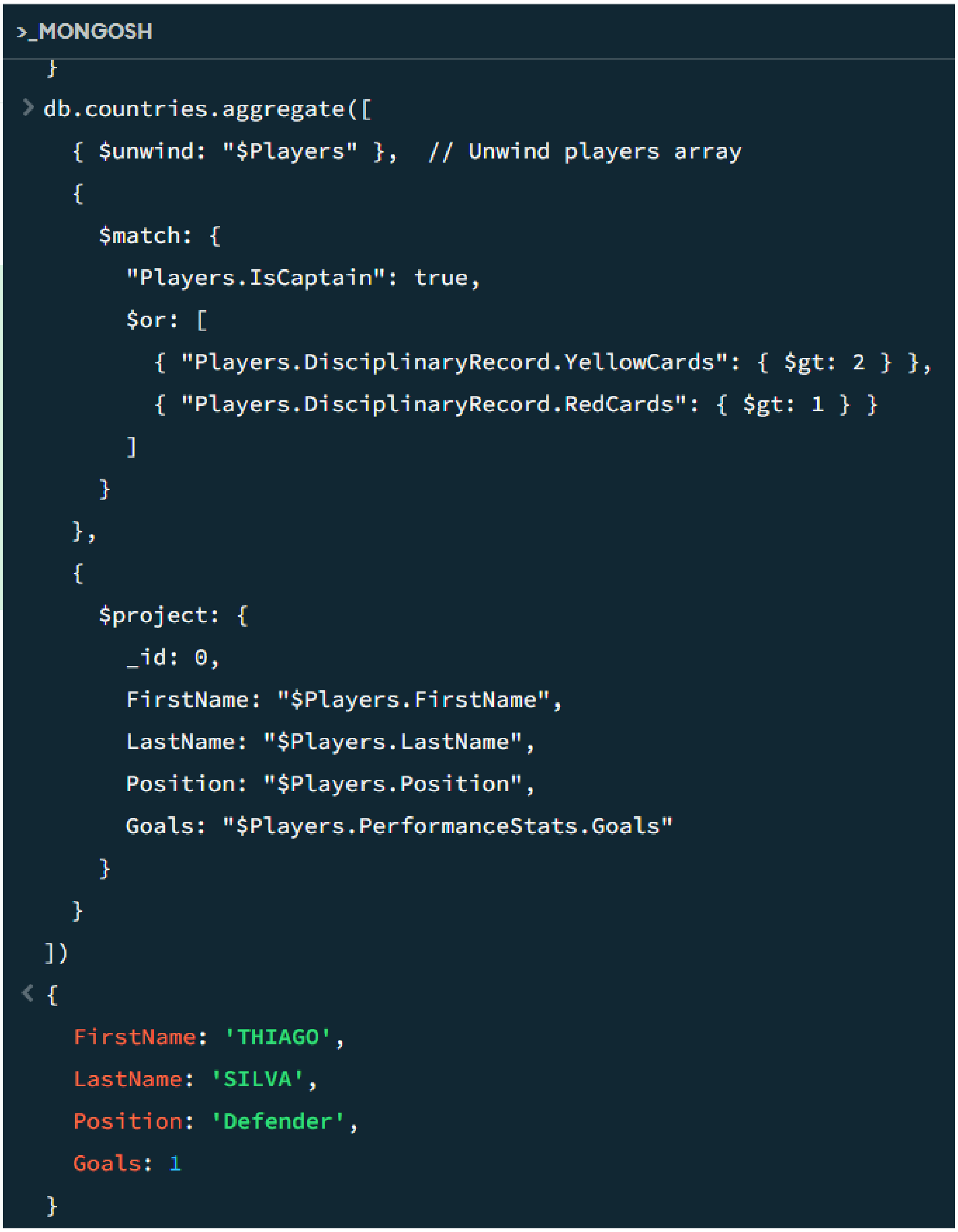
```
  {
    matchCount: 7,
    Stadium: 'Estadio do Maracana'
  }
  {
    matchCount: 6,
    Stadium: 'Arena Fonte Nova'
  }
  {
    matchCount: 5,
    Stadium: 'Arena Pernamuco'
  }
  {
    matchCount: 6,
    Stadium: 'Estadio Mineirao'
  }
  {
    matchCount: 7,
    Stadium: 'Estadio Nacional'
  }
  {
    matchCount: 4,
    Stadium: 'Arena Da Baixada'
```

List players taller than 198 cm:



```
> db.countries.aggregate([
    { $unwind: "$Players" },  // Unwind players array
    {
      $match: {
        "Players.Height": { $gt: 198 }
      }
    },
    {
      $project: {
        _id: 0,
        FirstName: "$Players.FirstName",
        LastName: "$Players.LastName",
        DateOfBirth: "$Players.DateOfBirth"
      }
    }
  ])
< {
    FirstName: 'FRASER',
    LastName: 'FORSTER',
    DateOfBirth: 1988-03-17T00:00:00.000Z
  }
  {
    FirstName: 'LEE',
    LastName: 'BUMYOUNG',
    DateOfBirth: 1989-04-02T00:00:00.000Z
```

List captains with more than 2 yellow cards or 1 red card:



```
  }
> db.countries.aggregate([
    { $unwind: "$Players" },  // Unwind players array
    {
      $match: {
        "Players.IsCaptain": true,
        $or: [
          { "Players.DisciplinaryRecord.YellowCards": { $gt: 2 } },
          { "Players.DisciplinaryRecord.RedCards": { $gt: 1 } }
        ]
      }
    },
    {
      $project: {
        _id: 0,
        FirstName: "$Players.FirstName",
        LastName: "$Players.LastName",
        Position: "$Players.Position",
        Goals: "$Players.PerformanceStats.Goals"
      }
    }
  ])
< {
    FirstName: 'THIAGO',
    LastName: 'SILVA',
    Position: 'Defender',
    Goals: 1
  }
```

Conclusion

This project demonstrated how to migrate relational data into a document-oriented NoSQL system like MongoDB. It provided insight into the advantages of nested data structures and complex queries in NoSQL databases. The queries successfully retrieved meaningful information, highlighting MongoDB's flexibility.