

## NoSQL Implementation:

Two tables(Nba Team, Players) have been created in the MongoDB playground. The following MongoDB queries were done:

### 1. Data insertion

Table 1: Nba Team

```
db.nba.drop();
db.nba.insert({"_id":"BNY","Player_ID":53,"Doctor_ID":41,"Physio_ID":9,"Manager_ID":37,"Coach_ID":63,"Team_Mascot":"Eagles", "Team_Name":"Brooklyn Nets","City":"Brooklyn"});
db.nba.insert({"_id":"BYN","Player_ID":30,"Doctor_ID":79,"Physio_ID":15,"Manager_ID":74,"Coach_ID":38,"Team_Mascot":"Bulldogs", "Team_Name":"Brooklyn Nets","City":"Brooklyn"});
db.nba.insert({"_id":"GSW","Player_ID":2,"Doctor_ID":80,"Physio_ID":77,"Manager_ID":57,"Coach_ID":3,"Team_Mascot":"Wolves", "Team_Name":"Golden StateWarriors", "City":"Golden"});
db.nba.insert({"_id":"LAL","Player_ID":2,"Doctor_ID":8,"Physio_ID":2,"Manager_ID":63,"Coach_ID":20,"Team_Mascot":"Dragons", "Team_Name":"Los Angeles Lakers", "City":"Los Angeles"});
db.nba.insert({"_id":"MIA","Player_ID":99,"Doctor_ID":53,"Physio_ID":94,"Manager_ID":21,"Coach_ID":43,"Team_Mascot":"Dragons", "Team_Name":"Miami Heat", "City":"Miami"});
db.nba.insert({"_id":"MIL","Player_ID":71,"Doctor_ID":73,"Physio_ID":65,"Manager_ID":72,"Coach_ID":45,"Team_Mascot":"Bears", "Team_Name":"Milwaukee BucksNets", "City":"Milwaukee"});
```

Table 2: Player

```
db.Player.drop();
db.Player.insert({"_id":53,"Goals_Scored":21,"Player_Name":"LonneyWeyland","Height":5,"3PTS":1,"2PTS":98, "Weight":87, "Position":"Center","Minutes_Played":"01:50:00","Diet_Plan":"Vegan", "Workout_Plan":"StrengthTraining","Games_Played":204, "DOB":"2001-08-18"});
db.Player.insert({"_id":30,"Goals_Scored":72,"Player_Name":"LayneKetting","Height":5,"3PTS":18,"2PTS":30, "Weight":76, "Position":"Center","Minutes_Played":"10:17:00","Diet_Plan":"Low Carb", "Workout_Plan":"Yoga","Games_Played":55, "DOB":"2005-12-13"});
db.Player.insert({"_id":2,"Goals_Scored":3,"Player_Name":"TerriCarnett","Height":5,"3PTS":67,"2PTS":59, "Weight":87, "Position":"Center","Minutes_Played":"10:35:00","Diet_Plan":"Low Carb", "Workout_Plan":"Pirates","Games_Played":377, "DOB":"1990-10-20"});
db.Player.insert({"_id":99,"Goals_Scored":89,"Player_Name":"VirgilPriest","Height":5,"3PTS":79,"2PTS":32, "Weight":78, "Position":"Center","Minutes_Played":"21:53:00","Diet_Plan":"High Protein", "Workout_Plan":"Cardio","Games_Played":198, "DOB":"1992-09-18"});
db.Player.insert({"_id":71,"Goals_Scored":41,"Player_Name":"LarryBagge","Height":5,"3PTS":69,"2PTS":81, "Weight":70, "Position":"Center","Minutes_Played":"00:25:00","Diet_Plan":"High Protein", "Workout_Plan":"Yoga","Games_Played":133, "DOB":"1988-07-24"});
db.Player.find();
```

Table 3: Manager

```
db.Manager.drop();
db.Manager.insert({"_id":7,"Name":"Kana","Email":"kdnms1c@desdev.org"});
db.Manager.insert({"_id":10,"Name":"Kana","Email":"xironen1a@cio.us"});
db.Manager.insert({"_id":14,"Name":"Zola","Email":"zbrileman@dyndns.org"});
db.Manager.insert({"_id":18,"Name":"Myca","Email":"mknowlsey@is.gd"});
db.Manager.insert({"_id":19,"Name":"Car","Email":"cghest10@ca.gov"});
db.Manager.find();
```

Table 4 : Coach

```
db.Coach.drop();
db.Coach.insert({"_id":1,"Name":"Beckia","Email":"bhpigssnnc@sourceforge.net"});
db.Coach.insert({"_id":3,"Name":"Saw","Email":"sneigho4@nyu.edu"});
db.Coach.insert({"_id":5,"Name":"Randene","Email":"rathary18@reliant.com"});
db.Coach.insert({"_id":7,"Name":"Zaccaria","Email":"zafalfredd@com.com"});
db.Coach.insert({"_id":8,"Name":"Yokre","Email":"yepppard@cisco.com"});
db.Coach.find();
```

### Queries :

#### 1.Query to Retrieve Players Born After 2000

```
db.Player.find({"DOB": {"$gte": "2000-01-01"}});
```

#### 2. Find the sum of the goals scored by each player

```
db.Players.aggregate([
{ $group: { _id: "$Player_Name", total: { $sum: "$Goals_Scored" } } },
{ $sort: { total: 1 } }
]);
```

#### 3. Map reduce: segregating goals scored by different diet plans.

```
// Map function
var mapFunction = function() {
emit(this.Diet_Plan, { totalGoals: this.Goals_Scored, count: 1 });
```

```
};  
// Reduce function  
var reduceFunction = function(key, values) {  
  var result = { totalGoals: 0, count: 0 };  
  values.forEach(function(value) {  
    result.totalGoals += value.totalGoals;  
    result.count += value.count;  
  });  
  return result;  
};  
// Finalize function (calculate average)  
var finalizeFunction = function(key, reducedValue) {  
  reducedValue.averageGoals = reducedValue.totalGoals / reducedValue.count;  
  return reducedValue;  
};  
// Execute MapReduce  
var result = db.player.mapReduce(  
  mapFunction,  
  reduceFunction,  
  { out: { inline: 1 }, finalize: finalizeFunction }  
);  
// Display the result  
printjson(result);
```