

MongoDB - Esercitazione (Lookup)

Lookup

Games & NBA2016

1. Unire la collection games con la collection nba2016players

```
db.games.aggregate([  
  $unwind: "$box"  
},{  
  $unwind: "$box.players"  
},{  
  $group: {  
    "_id": "$box.players.player"  
  }  
},{  
  $lookup: {  
    from: 'nba2016players',  
    localField: '_id',  
    foreignField: 'name',  
    as: 'stat'  
  }  
}])
```

2. Dalla query precedente, escludere i giocatori per cui non sono state trovate statistiche
 - a. Attenzione: se non vengono trovate statistiche, il nuovo campo sarà un array vuoto
3. Estendere la query precedente per visualizzare il nome del giocatore, il totale di punti fatti ed il valore suo contratto (contract.amount)
 - a. Attenzione: questo richiede di “srotolare” l’array
4. Estendere la query precedente per calcolare quanto “costa” un punto segnato da ciascun giocatore; ordinare i risultati sulla base di questo risultato
 - a. Attenzione: dividere il totale di punti fatti per il valore del contratto, moltiplicato per 1000
5. Unire la collection games con la collection nba2016teams
 - a. Attenzione: in games, i nomi delle squadre concatenano un’indicazione geografica (“Los Angeles”) ed il nome vero e proprio (“Lakers”), che in nba2016teams sono separati nei campi “region” e “name”. Per fare il join, bisogna:
 - i. Splittare il nome in games utilizzando *\$split: [nomeCampo, separatore]*
 - ii. Considerare l’ultimo valore nell’array risultante usando *\$slice: [array, valore]*
 - iii. Usare l’operatore \$unwind per trasformare l’array risultante in un oggetto
6. Estendere la query precedente per visualizzare il nome della squadra, il totale di punti fatti ed il costo medio di un biglietto (budget.ticketPrice)

YelpBusiness & Cities

7. Unire la collection yelpBusiness con la collection cities
8. Calcolare, per ogni categoria, la popolazione media e ordinare in base a quest’ultima
9. Calcolare, per ogni categoria, il rank medio della propria città e ordinare in base a quest’ultimo