

DAW2 - M2. Bases de dades

UF4. MongoDB

Activitat consultes 5a

Utilitzarem la col·lecció zips: <http://media.mongodb.org/zips.json>

Exercicis

1. Obtén la població total de cada ciutat, ordena per nom de ciutat de forma descendent.

```
> db.zips.aggregate([ { $group: { _id: "$city", pop: { $sum: "$pop" } } }, { $sort: { _id: 1 } } ] )
{ "_id" : "AARON", "pop" : 270 }
{ "_id" : "AARONSBURG", "pop" : 100 }
{ "_id" : "ABAC", "pop" : 27906 }
{ "_id" : "ABBEVILLE", "pop" : 23400 }
{ "_id" : "ABBOT VILLAGE", "pop" : 1193 }
{ "_id" : "ABBOTSFORD", "pop" : 2480 }
{ "_id" : "ABBOTT", "pop" : 23.080000000000002 }
{ "_id" : "ABBOTT PARK", "pop" : 26542 }
{ "_id" : "ABBOTTSTOWN", "pop" : 1777 }
{ "_id" : "ABBYVILLE", "pop" : 267 }
{ "_id" : "ABELL", "pop" : 601 }
{ "_id" : "ABERDEEN", "pop" : 96073 }
{ "_id" : "ABERDEEN PROVING", "pop" : 5294 }
{ "_id" : "ABERNATHY", "pop" : 139.28 }
{ "_id" : "ABIE", "pop" : 282 }
```

2. El mateix per aquelles ciutats amb més de 1000000 habitants.

```
> db.zips.aggregate([ { $group: { _id: "$city", pop: { $sum: "$pop" } } }, { $match: { pop: { $gt: 1000000 } } }, { $sort: { _id: 1 } } ] )
{ "_id" : "BRONX", "pop" : 1209548 }
{ "_id" : "BROOKLYN", "pop" : 2341387 }
{ "_id" : "CHICAGO", "pop" : 2452177 }
{ "_id" : "LOS ANGELES", "pop" : 2102295 }
{ "_id" : "NEW YORK", "pop" : 1476790 }
{ "_id" : "PHILADELPHIA", "pop" : 1639862 }
{ "_id" : "SAN DIEGO", "pop" : 1049498.72 }
```

3. Obtén la població total de cada estat. Ordena per estat.

```
> db.zips.aggregate([ { $group: { _id: "$state", pop: { $sum: "$pop" } } }, { $sort: { _id: 1 } } ] )
{ "_id" : "AK", "pop" : 544698 }
{ "_id" : "AL", "pop" : 4040587 }
{ "_id" : "AR", "pop" : 2350725 }
{ "_id" : "AZ", "pop" : 3665228 }
{ "_id" : "CA", "pop" : 29754890 }
{ "_id" : "CO", "pop" : 3293755 }
{ "_id" : "CT", "pop" : 3287116 }
{ "_id" : "DC", "pop" : 606900 }
{ "_id" : "DE", "pop" : 666168 }
{ "_id" : "FL", "pop" : 12686644 }
{ "_id" : "GA", "pop" : 6478216 }
{ "_id" : "HI", "pop" : 1108229 }
```

4. Obtén la població total de cada estat. Mostra només aquells estats amb una població superior a 10 millions. Ordena per població.

```
> db.zips.aggregate([ { $group: { _id: "$state", pop: { $sum: "$pop" } } }, { $match: { pop: { $gt: 10000000 } } }, { $sort: { pop: 1 } } ] )
{ "_id" : "OH", "pop" : 10846517 }
{ "_id" : "IL", "pop" : 11427576 }
{ "_id" : "PA", "pop" : 11881643 }
{ "_id" : "FL", "pop" : 12686644 }
{ "_id" : "NY", "pop" : 17990402 }
{ "_id" : "CA", "pop" : 29754890 }
>
```

5. Mostra quantes ciutats hi ha en cada estat. Ordena per nom d'estat.

```
> db.zips.aggregate([ { $group: { _id: "$state", cities: { $sum: 1 } } }, { $sort: { state: 1 } } ] )
{ "_id" : "WA", "cities" : 484 }
{ "_id" : "OR", "cities" : 384 }
{ "_id" : "ME", "cities" : 410 }
{ "_id" : "OH", "cities" : 1007 }
{ "_id" : "AR", "cities" : 578 }
{ "_id" : "KY", "cities" : 809 }
{ "_id" : "MS", "cities" : 363 }
{ "_id" : "IA", "cities" : 922 }
{ "_id" : "AL", "cities" : 567 }
{ "_id" : "PA", "cities" : 1458 }
{ "_id" : "NV", "cities" : 104 }
{ "_id" : "GA", "cities" : 635 }
{ "_id" : "IL", "cities" : 1237 }
{ "_id" : "SC", "cities" : 350 }
{ "_id" : "NE", "cities" : 574 }
{ "_id" : "FL", "cities" : 804 }
{ "_id" : "WY", "cities" : 156 }
>
```

6. Obtén la ciutat amb més població en total.

```
> db.zips.aggregate([ { $group: { _id: "$city", pop: { $max: "$pop" } } }, { $sort: { pop: -1 } }, { $limit: 1 } ] )
{ "_id" : "CHICAGO", "pop" : 112047 }
>
```

7. Obtén les 10 ciutats amb menys població en total, que sigui diferent de zero.

```
> db.zips.aggregate([ { $group: { _id: "$city", pop: { $max: "$pop" } } }, { $match: { pop: { $gt: 0 } } }, { $sort: { pop: 1 } }, { $limit: 10 } ] )
{ "_id" : "PORT O CONNOR", "pop" : 0.2 }
{ "_id" : "MULDOON", "pop" : 0.32 }
{ "_id" : "TARPLEY", "pop" : 0.8 }
{ "_id" : "ART", "pop" : 0.96 }
{ "_id" : "JEMEZ SPRINGS", "pop" : 1 }
{ "_id" : "CROOKED CREEK", "pop" : 1 }
{ "_id" : "SUMMER LAKE", "pop" : 1 }
{ "_id" : "KENNEDY SPACE CE", "pop" : 1 }
{ "_id" : "NEW HYDE PARK", "pop" : 1 }
{ "_id" : "WRIGHTSBORO", "pop" : 1.04 }
>
```

Activitat consultes 5b

Utilitzarem la col·lecció students. Importa-la amb:

```
>mongoimport /db:test /collection:students /type:csv /headerline  
/file:students.csv
```

```
$ mongoimport -d test -c students --type csv --file students.csv --headerline
```

Exercicis

8. Mostra el nombre d'estudiants del sexe masculí agrupats per any de naixement.

```
> db.students.aggregate({ $match: { gender: "H" } }, { $group: { _id: "$birth_year", students: { $sum: 1 } } })  
{ "_id" : 1971, "students" : 79 }  
{ "_id" : 1991, "students" : 83 }  
{ "_id" : 1961, "students" : 89 }  
{ "_id" : 1974, "students" : 76 }  
{ "_id" : 1965, "students" : 93 }  
{ "_id" : 1977, "students" : 79 }  
{ "_id" : 1970, "students" : 85 }  
{ "_id" : 1967, "students" : 101 }  
{ "_id" : 1990, "students" : 88 }  
{ "_id" : 1985, "students" : 87 }  
{ "_id" : 1968, "students" : 97 }  
{ "_id" : 1976, "students" : 84 }  
{ "_id" : 1960, "students" : 106 }  
{ "_id" : 1982, "students" : 87 }  
{ "_id" : 1986, "students" : 109 }  
{ "_id" : 1992, "students" : 87 }
```

9. Mostra el nombre d'estudiants agrupats per any de naixement.

```
> db.students.aggregate({ $group: { _id: "$birth_year", students: { $sum: 1 } } })  
{ "_id" : 1971, "students" : 87 }  
{ "_id" : 1961, "students" : 97 }  
{ "_id" : 1977, "students" : 87 }  
{ "_id" : 1988, "students" : 87 }  
{ "_id" : 1963, "students" : 104 }  
{ "_id" : 1991, "students" : 92 }  
{ "_id" : 1981, "students" : 86 }  
{ "_id" : 1985, "students" : 96 }  
{ "_id" : 1968, "students" : 108 }  
{ "_id" : 1976, "students" : 92 }  
{ "_id" : 1960, "students" : 122 }  
{ "_id" : 1990, "students" : 98 }  
{ "_id" : 1967, "students" : 106 }
```

10. Obtenir l'alumne de sexe masculí més jove.

```
> db.students.find({ gender: "H" }).sort({ birth_year: -1 }).limit(1).pretty()
{
  "_id" : ObjectId("5eb9b2f178ddfa341a6bdcd6"),
  "firstname" : "Jaume",
  "lastname1" : "Volkoff",
  "lastname2" : "Adell",
  "dni" : "08885964Y",
  "gender" : "H",
  "email" : "kellybomaholly@hotmail.com",
  "phone" : "629.219.229",
  "phone_aux" : "",
  "birth_year" : 1993
}
```

11. Quants noms de pila hi ha repetits? Mostra els diferents noms, i el nombre de vegades que es repeteixen.

```
> db.students.aggregate( { $group: { _id: "$firstname", times_repeated: { $sum: 1 } } }, { $sort: { "times_repeated": -1 } } )
{ "_id" : "Jordi", "times_repeated" : 143 }
{ "_id" : "David", "times_repeated" : 118 }
{ "_id" : "Daniel", "times_repeated" : 78 }
{ "_id" : "Javier", "times_repeated" : 75 }
{ "_id" : "Carlos", "times_repeated" : 67 }
{ "_id" : "Xavier", "times_repeated" : 65 }
{ "_id" : "Albert", "times_repeated" : 55 }
{ "_id" : "Sergio", "times_repeated" : 49 }
{ "_id" : "Oscar", "times_repeated" : 48 }
{ "_id" : "Alejandro", "times_repeated" : 45 }
{ "_id" : "Antonio", "times_repeated" : 45 }
{ "_id" : "Marc", "times_repeated" : 43 }
{ "_id" : "Joan", "times_repeated" : 37 }
{ "_id" : "Jorge", "times_repeated" : 35 }
{ "_id" : "Alberto", "times_repeated" : 35 }
{ "_id" : "Josep", "times_repeated" : 33 }
{ "_id" : "Ivan", "times_repeated" : 29 }
{ "_id" : "Raul", "times_repeated" : 28 }
{ "_id" : "Oriol", "times_repeated" : 28 }
{ "_id" : "Manuel", "times_repeated" : 27 }
Type "it" for more
>
```

12. Mostra la mitjana d'edat per sexe.

```
> db.students.aggregate({ $group: { _id: "$gender", avg_age: { $avg: { $subtract: [ { "$year": new Date() }, "$birth_year" ] } } } })
{ "_id" : "H", "avg_age" : 43.766493955094994 }
{ "_id" : "M", "avg_age" : 43.82471264367816 }
```