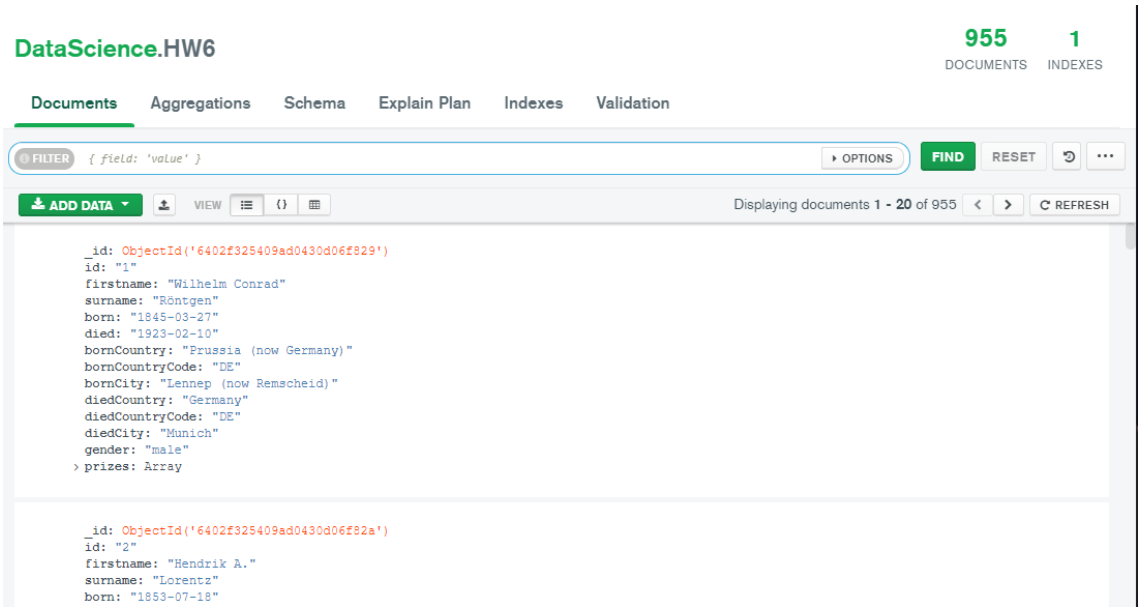
CS 457 - Homework Assignment 6: NoSQL

CS 457 Data Science
Habib University
Spring 2023

1. Create a Database and Load/Import the dataset into collection using dataset laureate2.json

Solution:
Loading DataSet into MongoDb



2. Query the collection and interpret the results, displaying:
   (a) the count of total number of records in the collection

Solution:

```
> db.HW6.count()
< 'DeprecationWarning: Collection.count() is deprecated. Use countDocuments or estimatedDocumentCount.'
< 955
Atlas atlas-jrf3pt-shard-0 [primary] DataScience>
```

   (b) the count of records for each diedCountryCode in descending order of count.

Solution:

```
db.HW6.aggregate([{$group: {_id: '$diedCountryCode', count: {$sum:1}}},
\\ {$sort:{count:-1}}])
```

```
> db.HW6.aggregate([{$group: {_id: '$diedCountryCode', count: {$sum:1}}},{$sort:{count:-1}}])
< { _id: null, count: 335 }
  { _id: 'US', count: 223 }
  { _id: 'GB', count: 83 }
  { _id: 'DE', count: 59 }
  { _id: 'FR', count: 51 }
  { _id: 'SE', count: 29 }
  { _id: 'CH', count: 27 }
  { _id: 'RU', count: 15 }
  { _id: 'IT', count: 14 }
  { _id: 'ES', count: 10 }
  { _id: 'NL', count: 10 }
  { _id: 'DK', count: 10 }
  { _id: 'JP', count: 8 }
  { _id: 'NO', count: 8 }
```

(c) the count of records for each prizes.category in descending order of count.

Solution:

```
> db.HW6.aggregate([{$unwind: '$prizes'},{$group: {_id: '$prizes.category', count:{$sum:1}}},{$sort:{count:-1}}])
< { _id: 'medicine', count: 222 }
  { _id: 'physics', count: 216 }
  { _id: 'chemistry', count: 186 }
  { _id: 'peace', count: 135 }
  { _id: 'literature', count: 117 }
  { _id: 'economics', count: 86 }
Atlas atlas-jrf3pt-shard-0 [primary] DataScience>
```

(d) the count of records for each gender, diedCountryCode, prize.category when prize.category is "physics". Order the output by diedCountryCode.

Solution:

```
    db.collection.aggregate([
  {
    $match: { "prizes.category": "physics" }
  },
  {
    $group: {
      _id: {
        gender: "$gender",
        diedCountryCode: "$diedCountryCode",
        prizeCategory: "$prizes.category"
      },
      count: { $sum: 1 }
    }
  },
  {
    $sort: {
      "_id.diedCountryCode": 1
    }
  }
])
```

```
{ _id: { gender: 'male', prizeCategory: [ 'physics' ] },
  count: 76 }
{ _id: { gender: 'female', prizeCategory: [ 'physics' ] },
  count: 2 }
{ _id:
   { gender: 'male',
     diedCountryCode: 'AT',
     prizeCategory: [ 'physics' ] },
  count: 1 }
{ _id:
   { gender: 'male',
     diedCountryCode: 'CA',
     prizeCategory: [ 'physics' ] },
  count: 2 }
```

Another Sample Output

```
{ _id:
   { gender: 'male',
     diedCountryCode: 'DE',
     prizeCategory: [ 'physics' ] },
  count: 15 }
{ _id:
   { gender: 'male',
     diedCountryCode: 'DK',
     prizeCategory: [ 'physics' ] },
  count: 2 }
{ _id:
   { gender: 'female',
     diedCountryCode: 'FR',
     prizeCategory: [ 'physics', 'chemistry' ] },
  count: 1 }
{ _id:
   { gender: 'male',
     diedCountryCode: 'FR',
     prizeCategory: [ 'physics' ] },
  count: 8 }
```

(e) Come up with your own query to show any interesting insight. Use atleast two fields for match and two fields for group

Solution:

// For this Query, I checked the number of records or users who are born in Netherland and died in Netherland. Thsi was I can know those who died in their born country.

```
    db.HW6.aggregate([
  {
    $match: { "diedCountryCode":"NL" , "bornCountryCode":"NL"}
  },
  {
    $group: {
      _id: {
        gender: "$gender",

        prizeCategory: "$prizes.category"
      },
      count: { $sum: 1 }
    }
  },
  {
    $sort: {
      "_id.diedCountryCode": 1
    }
  }
])
```

```
{ _id: { gender: 'male', prizeCategory: [ 'physics' ] },
  count: 6 }
{ _id: { gender: 'male', prizeCategory: [ 'medicine' ] },
  count: 1 }
{ _id: { gender: 'male', prizeCategory: [ 'peace' ] },
  count: 1 }
{ _id: { gender: 'male', prizeCategory: [ 'economics' ] },
  count: 1 }
Atlas atlas-jrf3pt-shard-0 [primary] DataScience>
```