Chelsea Aguilar
CECS 323
11/30/22

CECS 323: Project 3 The Mongo Mash

Due Date: December 10, 2022

Deliverables

You must deliver to me, via Dropbox, a **single PDF** containing the following documents:

- Title page (name of course, title of assignment, team member names, due date)
- Schemas for adding actors to films. Include **all** schemas that you created or modified to implement this.
- A **written justification** for the decisions you made in implementing the actors (and related) schemas. This **1-2 paragraph response, in full English sentences**, should communicate the depth and breadth of decisions you made in designing this part of the schema. You should, at a minimum, answer the questions posed to you in the "A New Collection" section, in the context of your final answer.
- JSON documents for the theaters, films, showings, and actors that you were required to create.
- Mongosh commands that you wrote to solve the 3 queries. **I do not need the *output* of those queries, just the statements themselves**.

Schemas:

- Schemas for adding actors to films. Include **all** schemas that you created or modified to implement this.

1. **schema_actors.json**

```
{
    "$jsonSchema": {
        "title": "actors",
```

Chelsea Aguilar
CECS 323
11/30/22

```
        "description": "a list of actors for a film",
        "bsonType": "object",
        "required" : ["_id", "firstname", "lastname", "birthday",
    "imdbURL"],
        "properties": {
            "_id": {
                "bsonType": "objectId"
            },
            "firstname": {
                "bsonType": "string"
            },
            "lastname": {
                "bsonType": "string"
            },
            "birthday": {
                "bsonType": "date"
            },
            "imdbURL": {
                "bsonType": "string"
            }
        }}


}
```

Chelsea Aguilar
CECS 323
11/30/22

2. **schema_films.json**

```
{
    "$jsonSchema": {
        "title": "film",
        "description": "A film shown at theaters.",
        "bsonType": "object",
        "required": [
            "title",
            "duration",
            "releaseDate",
            "rating",
            "genres",
            "actors"
        ],
        "properties": {
            "_id": {
                "bsonType": "objectId"
            },
            "title": {
                "bsonType": "string"
            },
            "duration": {
                "bsonType": "int",
                "minimum": 1
            },
            "releaseDate": {
                "bsonType": "date"
            },
            "rating": {
                "bsonType": "string",
                "enum": [
                    "NR",
```

Chelsea Aguilar
CECS 323
11/30/22

```
                "G",
                "PG",
                "PG-13",
                "R",
                "NC-17"
            ]
        },
        "genres": {
            "bsonType": "array",
            "minItems": 1,
            "items": {
                "bsonType": "string"
            }
        },
        "actors" : {
            "bsonType": "array",
            "items" :{
                "title": "actors",
                "description": "a list of actors for a film",
                "bsonType": "object",
                "required" : ["_id", "firstname", "lastname", "birthday",
            "imdbURL"],
                "properties": {
                    "_id": {
                        "bsonType": "objectId"
                    },
                    "firstname": {
                        "bsonType": "string"
                    },
                    "lastname": {
                        "bsonType": "string"
                    },
                    "birthday": {
                        "bsonType": "date"
                    },
                    "imdbURL": {
                        "bsonType": "string"
```

Chelsea Aguilar
CECS 323
11/30/22

```
                    }
                }}

            }
        }
    }
}

                    "_id": {
                        "bsonType": "objectId"
                    },
                    "firstname": {
                        "bsonType": "string"
                    },
                    "lastname": {
                        "bsonType": "string"
                    },
                    "birthday": {
                        "bsonType": "date"
                    },
                    "imdbURL": {
                        "bsonType": "string"
                    }
                }}

            }
        }
    }
}
```

Schema Justifications:

- A **written justification** for the decisions you made in implementing the actors (and related) schemas. This **1-2 paragraph response, in full English sentences**, should communicate the depth and breadth of decisions you made in designing this part of the schema. You should, at a minimum, answer the questions posed to you in the "A New Collection" section, in the context of your final answer.

Chelsea Aguilar
CECS 323
11/30/22

I made a new actors schema instead of embedding into schema_films. A film has many actors and an actor stars in many films. An actor is able to be in many films and a film can have many actors such as the film Black Panther Wakanda Forever which had 130 actors in its cast. A good portion of the actors are extras but there will be people who want information on the stars of the film. When the Parent embeds the child there is also a concern that we may hit a max size on a document which can occur in a large film production with many extras. Avengers Infinity War also had a cast of 144 so embedding actors into films would bring up the document size further proving another reason to create a new actors schema and actors collection. If we had embedded the actor into films,the additional actor attributes would also take up more space and be unnecessary.

In this theater enterprise, it is unlikely that someone will want to buy tickets to showings based on which actors are in the films. When we load up a film document to display a webpage with a film's details, we would like to know the cast of the film by listing an actor's first and last name.We wouldn't want to expand on that by listing the actor's birthday on the web page so having a few details on the actor would suffice but not all of them. Based on the theater enterprise, we will be more interested in the film details overall. There is an opportunity for denormalization and redundancy because there are repeated attributes of Angela Bassett appearing in multiple films. I don't think it is necessary to quickly find all films starring a particular actor since people who are on a theater website searching for showings of films, are going to be more interested in the film details itself rather than any actors.

JSON documents:

- JSON documents for the theaters, films, showings, and **actors** that you were required to create.

Chelsea Aguilar
CECS 323
11/30/22

---

1. **films.json**


```
[{
  "_id": {
    "$oid": "63954b6f44484c9b37b864cd"
  },
  "title": "Akeelah and the Bee",
  "duration": 112,
  "releaseDate": {
    "$date": {
      "$numberLong": "1142467200000"
    }
  },
  "rating": "PG",
  "genres": [
    "Family",
    "Drama"
  ],
  "actorsid": [
    {
      "$oid": "6395483544484c9b37b864c8"
    }
  ]
},{
  "_id": {
    "$oid": "63954b7a44484c9b37b864cf"
  },
  "title": "Black Panther: Wakanda Forever",
  "duration": 161,
  "releaseDate": {
    "$date": {
      "$numberLong": "1668038400000"
    }
  },
  "rating": "PG-13",
  "genres": [
    "Action",
```

Chelsea Aguilar
CECS 323
11/30/22


```
    "Adventure",
    "Superhero"
  ],
  "actorsid": [
    {
      "$oid": "6395481644484c9b37b864c2"
    },
    {
      "$oid": "6395482144484c9b37b864c4"
    },
    {
      "$oid": "6395482b44484c9b37b864c6"
    },
    {
      "$oid": "6395483544484c9b37b864c8"
    }
  ]
},{
  "_id": {
    "$oid": "63954b8444484c9b37b864d1"
  },
  "title": "No Time to Die",
  "duration": 163,
  "releaseDate": {
    "$date": {
      "$numberLong": "1633651200000"
    }
  },
  "rating": "PG-13",
  "genres": [
    "Action",
    "Adventure",
    "Thriller"
  ],
  "actorsid": []
}]
```

Chelsea Aguilar
CECS 323
11/30/22

---

2. **theaters.json**


```
[{
 "_id": {
   "$oid": "638b0c1d343fd06a93529efb"
 },
 "name": "Regal Edwards Long Beach",
 "address": "7501 E Carson St",
 "city": "Long Beach",
 "state": "CA",
 "zipcode": "90808",
 "rooms": [
   {
     "title": "Screen 1",
     "capacity": 5,
     "formats": [
       "Standard",
       "IMAX"
     ],
     "seats": [
       {
         "number": "1",
         "Labels": [
           "reclining"
         ]
       },
       {
         "number": "2",
         "Labels": [
           "reclining"
         ]
       },
       {
         "number": "3",
         "Labels": [
           "reclining"
```

Chelsea Aguilar
CECS 323
11/30/22


```
          ]
        },
        {
          "number": "4",
          "Labels": [
            "reclining"
          ]
        },
        {
          "number": "5",
          "Labels": [
            "accessible seating",
            "non-reclining"
          ]
        }
      ]
    },
    {
      "title": "Screen 2",
      "capacity": 1,
      "formats": [
        "ScreenX",
        "IMAX"
      ],
      "seats": [
        {
          "number": "1"
        }
      ]
    }
  ]
}]
```

Chelsea Aguilar
CECS 323
11/30/22

---

3. **showings.json**


```
[{
  "_id": {
    "$oid": "638bdcbf3ba95162eba34255"
  },
  "theaterId": {
    "$oid": "638b0c1d343fd06a93529efb"
  },
  "filmId": {
    "$oid": "63954b7a44484c9b37b864cf"
  },
  "showTime": {
    "$date": {
      "$numberLong": "1668095100000"
    }
  },
  "format": "Standard",
  "tickets": [
    {
      "_id": {
        "$oid": "3c4f626a656374496428293e"
      },
      "price": 18,
      "seatNumber": "1"
    }
  ]
},{
  "_id": {
    "$oid": "638bdeeb3ba95162eba34261"
  },
  "theaterId": {
    "$oid": "638b0c1d343fd06a93529efb"
  },
  "filmId": {
    "$oid": "63954b7a44484c9b37b864cf"
```

Chelsea Aguilar
CECS 323
11/30/22


```
  },
  "showTime": {
    "$date": {
      "$numberLong": "1668258000000"
    }
  },
  "format": "Standard",
  "tickets": []
},{
  "_id": {
    "$oid": "638bdfad3ba95162eba34263"
  },
  "theaterId": {
    "$oid": "638b0c1d343fd06a93529efb"
  },
  "filmId": {
    "$oid": "63954b7a44484c9b37b864cf"
  },
  "showTime": {
    "$date": {
      "$numberLong": "1668106800000"
    }
  },
  "format": "IMAX",
  "tickets": [
    {
      "_id": {
        "$oid": "3c4f626a656374496428293e"
      },
      "price": 22,
      "seatNumber": "1"
    },
    {
      "_id": {
        "$oid": "3c4f626a656374496428293e"
      },
      "price": 15,
```

Chelsea Aguilar
CECS 323
11/30/22


```
      "seatNumber": "5"
    }
  ]
}]
```


---

4. **actors.json**


```
[{
  "_id": {
    "$oid": "6395481644484c9b37b864c2"
  },
  "firstname": "Letitia",
  "lastname": "Wright",
  "birthday": {
    "$date": {
      "$numberLong": "752025600000"
    }
  },
  "imdbURL": "https://www.imdb.com/name/nm4004793/?ref_=tt_cl_t_7"
},{
  "_id": {
    "$oid": "6395482144484c9b37b864c4"
  },
  "firstname": "Lupita",
```

Chelsea Aguilar
CECS 323
11/30/22


```
  "lastname": "Nyong'o",
  "birthday": {
    "$date": {
      "$numberLong": "415324800000"
    }
  },
  "imdbURL": "https://www.imdb.com/name/nm2143282/?ref_=tt_cl_t_3"
},{
  "_id": {
    "$oid": "6395482b44484c9b37b864c6"
  },
  "firstname": "Danai",
  "lastname": "Gurira",
  "birthday": {
    "$date": {
      "$numberLong": "256262400000"
    }
  },
  "imdbURL": "https://www.imdb.com/name/nm1775091/?ref_=tt_cl_t_4"
},{
  "_id": {
    "$oid": "6395483544484c9b37b864c8"
  },
  "firstname": "Angela",
```

Chelsea Aguilar
CECS 323
11/30/22


```
  "lastname": "Bassett",
  "birthday": {
    "$date": {
      "$numberLong": "-359078400000"
    }
  },
  "imdbURL": "https://www.imdb.com/name/nm0000291/"
}]
```

MONGO Queries:

1. **Select all films that have "Action" as one of their genres, sorted by film title.**

```
[{
      $match: {
       genres: "Action",
    },
  },
  {
    $sort: {
      title: 1,
    },
  },
```

Chelsea Aguilar
CECS 323
11/30/22

]

2. **Select the showtime of all Black Panther: Wakanda Forever showings that are on November 11. You don't need to use joins; you can hard-code the filmId of this film. Hint: you can use a filter with both $lt and $gt comparisons. A date is on November 11 2022 if it is greater than midnight of November 11 and less than ___.... To put a date into a query, you must write `ISODate("...")`, where the parameter is the date in YYYY-MM-DD format.**

```
[
 {
  /**
   * query: The query in MQL.
   */
  $match: {
   filmId: ObjectID("63954b7a44484c9b37b864cf"),
   showTime: {
    $gt: ISODate("2022-10-11"),
    $lt: ISODate("2022-12-12"),
   },
  },
 },
]
```

Chelsea Aguilar
CECS 323
11/30/22

3. **For each theater, select the name of the theater and also a list of all the film titles and showtimes of all showings scheduled in the theater. This requires a join from theater to showing.**

```
[
 {
  /**
   * specifications: The fields to
   *   include or exclude.
   */
  $lookup: {
   from: "showings",
   localField: "theatersID",
   foreignField: "showingsid",
   as: "allShowings",
  },
 },
 {
  /**
   * from: The target collection.
   * localField: The local join field.
   * foreignField: The target join field.
   * as: The name for the results.
   * pipeline: Optional pipeline to run on the foreign collection.
   * let: Optional variables to use in the pipeline field stages.
   */
  $lookup: {
   from: "films",
   localField: "showingsID",
   foreignField: "filmsid",
   as: "allFilms",
  },
 },
 {
  /**
   * from: The target collection.
   * localField: The local join field.
   * foreignField: The target join field.
```

Chelsea Aguilar
CECS 323
11/30/22

```
    * as: The name for the results.
    * pipeline: Optional pipeline to run on the foreign collection.
    * let: Optional variables to use in the pipeline field stages.
    */
   $lookup: {
    from: "theaters",
    localField: "theaterID",
    foreignField: "theaterid",
    as: "theatername",
   },
  },
  {
   /**
    * specifications: The fields to
    *   include or exclude.
    */
   $project: {
    allFilms: { title: 1 },
    allShowings: { showTime: 1 },
    theatername: { name: 1 },
   },
  },
]
```