1. Display the number of movies a director has made. Display only the name of the Director and the count of his/her movies.

**db.movie.aggregate([{$group:{\_id:{"director":"$director"},count:{$sum:1}}}])**

****

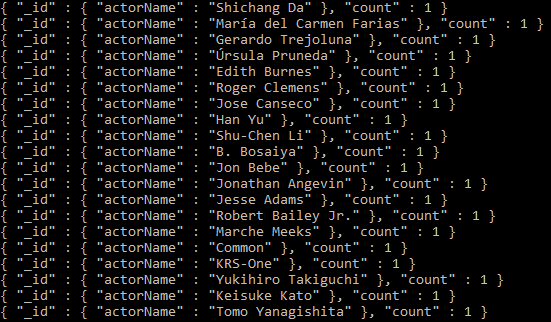
1. Number of Movies with a runtime greater than 120mins

**db.movie.aggregate([{$match: {"runtime":{$gt: 120}}},{$group:{\_id:"Total",count:{$sum:1}}}])**

****

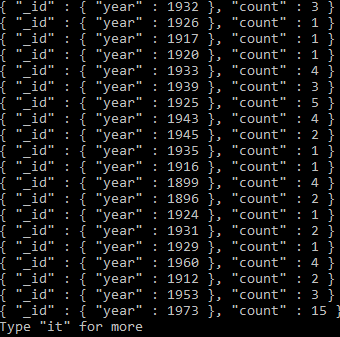
1. Display the number of movies an actor has starred in. Display only the name of the Actor/Actress and the count of his/her movies.

**db.movie.aggregate([{$unwind:"$actors"},{$group:{\_id:{"actorName":"$actors"},count:{$sum:1}}}])**



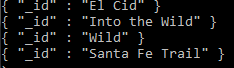
1. Display the number of Movies shown in a specific year. Display only the year and the count of movies.

**db.movie.aggregate([{$group:{\_id:{"year":"$year"},count:{$sum:1}}}])**

****

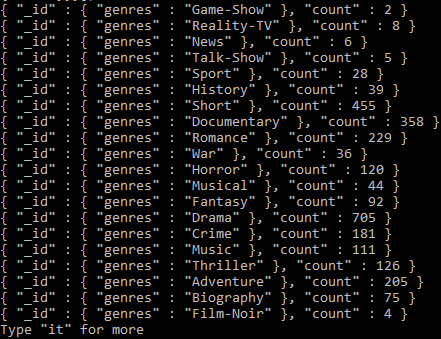
1. Display the Movie/Movies that meets the genres Adventure, Biography, Drama

**db.movie.aggregate([{$match:{$and:[{genres: "Adventure"},{genres: "Biography"},{genres: "Drama"}]}},{$group:{\_id:"$title"}}]) Display the number of movies per specific Genre.**

****

1. Display the number of movies per specific Genre.

**db.movie.aggregate([{$unwind:"$genres"}, {$group:{\_id:{genres:"$genres"}, count:{$sum:1}}}])**

****

1. Updated all the movies with a rated “R” rating to rated “R-18”

**db.movie.updateMany({"rated":"R"},{$set:{"rated":"R-18"}})**



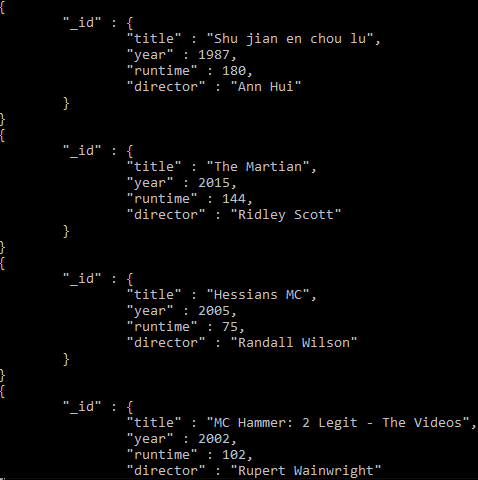
1. Movies who starred actor “Hayden Christensen” and directed by “George Lucas” display only Movie Name, Year, and Runtime.

**db.movie.aggregate([{$unwind: "$actors"},{$match:{$and:[{actors: "Hayden Christensen"},{director: "George Lucas"}]}},{$group:{\_id:{"title":"$title","year":"$year","runtime":"$runtime"}}}])**

****

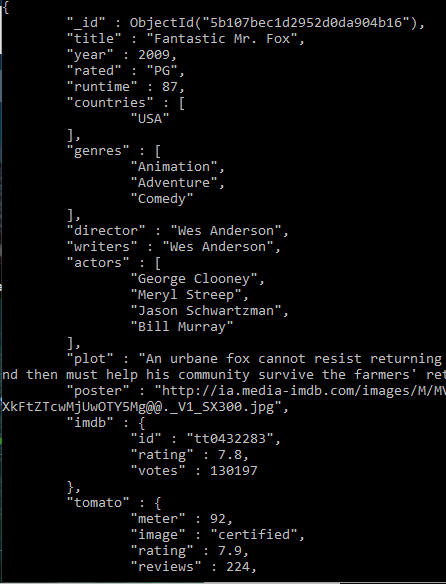
1. Display movie name, year, runtime, and director. Order the result by movie name, year in ascending order.

**db.movie.aggregate([{$group:{\_id:{"title":"$title","year":"$year","runtime":"$runtime","director":"$director"}}},{$sort:({"year":1,"title":1})}]).pretty()**

****

1. Display movies that includes “Wes Anderson” as a writer with a metacritic score greater than 80

**db.movie.aggregate([{$unwind:"$writers"},{$match:{writers:"Wes Anderson",metacritic:{$gt:80}}}]).pretty()**

****

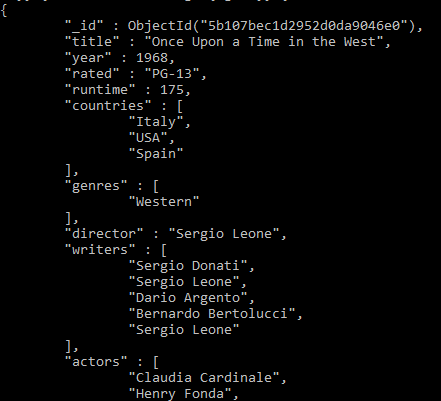
1. Remove all movies without an imdb rating or votes or whose imdb ratings or votes values are null

**db.movie.deleteMany({$or:[{imdb:{$exists: false}},{"imdb.votes":{$exists: false}},{imdb:null},{"imdb.votes": null}]})**

****

1. Display only the movies that has all of the following:
   1. Imdb rating greater than 6
   2. Imdb votes greater than 5,000
   3. a tomato meter greater than 70
   4. a tomato rating greater than 5
   5. a tomato userRating greater than 3
   6. a tomato userReviews greater than 1,000
   7. a metacritic rating greater than 40

**db.movie.find({$and:[{"imdb.rating":{$gt:6}},{"imdb.votes":{$gt:5000}},{"tomato.meter":{$gt:70}},{"tomato.rating":{$gt:5}},{"tomato.userRating":{$gt:3}},{"tomato.userReviews":{$gt:1000}},{"metacritic":{$gt:40}}]}).pretty()**

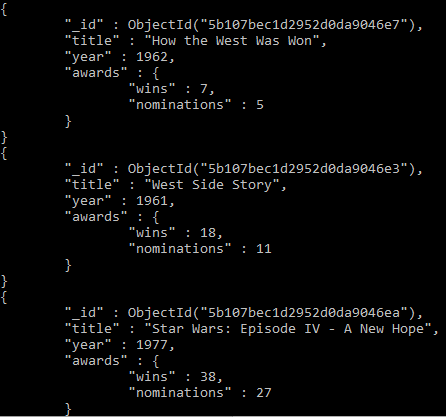
****

1. Display the number of movies found on #12  
   **db.movie.find({$and:[{"imdb.rating":{$gt:6}},{"imdb.votes":{$gt:5000}},{"tomato.meter":{$gt:70}},{"tomato.rating":{$gt:5}},{"tomato.userRating":{$gt:3}},{"tomato.userReviews":{$gt:1000}},{"metacritic":{$gt:40}}]}).count()**



1. Display all the movies that were either nominated for or won an Oscar. Display the movie name ,year, and number of award wins and nominations.

**db.movie.find({"awards.text" : {$regex : ".\*Oscar.\*"}},{"title":1,"year":1,"awards.wins":1,"awards.nominations":1}).pretty()**

****

1. Search for information on the 2019 movie Parasite, and add it as a document to this collection.

**Required:**

* Title: Parasite
* Year: 2019
* Rated: PG
* Runtime: 132
* Countries: South Korea
* Genres: Thriller/Comedy
* Director: Bong Joon-Ho
* Writers: Bong Joon-Ho, Han Jin-won
* Actors: [[Song Kang-ho](https://en.wikipedia.org/wiki/Song_Kang-ho), [Lee Sun-kyun](https://en.wikipedia.org/wiki/Lee_Sun-kyun), [Cho Yeo-jeong](https://en.wikipedia.org/wiki/Cho_Yeo-jeong), [Choi Woo-shik](https://en.wikipedia.org/wiki/Choi_Woo-shik), [Park So-dam](https://en.wikipedia.org/wiki/Park_So-dam), [Lee Jung-eun](https://en.wikipedia.org/wiki/Lee_Jung-eun_(actress)), [Jang Hye-jin](https://en.wikipedia.org/wiki/Jang_Hye-jin_(actress)), [Park Myung-hoon](https://en.wikipedia.org/wiki/Park_Myung-hoon), [Jung Ji-so](https://en.wikipedia.org/wiki/Jung_Ji-so), [Jung Hyeon-jun](https://en.wikipedia.org/wiki/Jung_Hyeon-jun), Park Keun-rok, [Park Seo-joon](https://en.wikipedia.org/wiki/Park_Seo-joon)]
* Plot: The Kim family—father Ki-taek, mother Chung-sook, daughter Ki-jung and son Ki-woo—live in a small semi-[basement apartment](https://en.wikipedia.org/wiki/Basement_apartment) (banjiha), have low-paying temporary jobs as pizza box folders, and struggle to make ends meet. University student Min-hyuk, a friend of Ki-woo's, gives the family a [scholar's rock](https://en.wikipedia.org/wiki/Suseok) meant to promise wealth. Leaving to study abroad, he suggests that Ki-woo take over his job as an English tutor for the daughter of the wealthy Park family, Da-hye. Ki-woo poses as a [Yonsei University](https://en.wikipedia.org/wiki/Yonsei_University" \o "Yonsei University) student and is hired by the Parks.
* imdb{id: tt6751668 , rating:8.6, votes:491247}

If it is not required you can give it a null value.

The document added should follow the exact same format as other documents in the collection.

**db.movie.insert({title : "Parasite", year : 2019, rated : "PG", runtime : 132, countries : ["South Korea"], genres : [ "Thriller", "Comedy" ], director : "Bong Joon-Ho", writers : [ "Bong Joon-Ho", "Han Jin-won"], actors : ["Song Kang-ho", "Lee Sun-kyun", "Cho Yeo-jeong", "Choi Woo-shik", "Park So-dam", "Lee Jung-eun", "Jang Hye-jin", "Park Myung-hoon", "Jung Ji-so", "Jung Hyeon-jun", "Park Keun-rok", "Park Seo-joon"], plot : "The Kim family—father Ki-taek, mother Chung-sook, daughter Ki-jung and son Ki-woo—live in a small semi-basement apartment (banjiha), have low-paying temporary jobs as pizza box folders, and struggle to make ends meet. room.", poster : null, imdb : {"id" : "tt6751668", "rating" : 8.6, "votes" : 491247}, tomato : { meter : null, image : null, rating : null, reviews : null, fresh : null, consensus : null, userMeter : null,userRating : null, userReviews : null}, metacritic : null,awards : {wins : null, nominations : null, text : null}, type : null})**

****

1. Display documents in a readable manner that were released after 1995 and whose name starts with the word “Star”. Sort by imdb ratings from descending order.

**db.movie.aggregate({$match:{title : {$regex : /^Star/}}},{$sort:({"imdb.rating":-1})}).pretty()**

****

1. Display the Movie/Movies that contain the name “Anakin Skywalker” in the plot

**db.movie.find({"plot" : {$regex : ".\*Anakin Skywalker.\*"}}).pretty()**

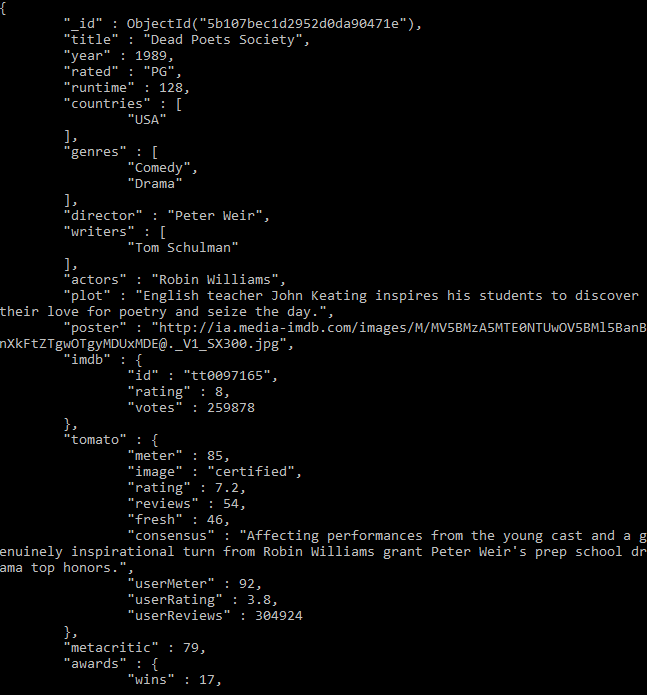


Bonus:

Guess/Find one of my favorite movies through the use of mongoDB queries!

It is a **comedy, drama** starring **Robin Williams** that has won one **Oscar** award

**db.movie.aggregate([{$unwind:"$actors"},{$match:{$and:[{"awards.text":{$regex:".\*Oscar.\*"}},{genres:"Drama"},{genres:"Comedy"},{actors:"Robin Williams"}]}}]).pretty()**

****