1. Demonstrate the usage of $match

> db.movies.aggregate({$match: {Genre :"Horror"}})

Demonstrate the usage of $group

> db.movies.aggregate({$group: { \_id:"$Genre",sum\_of\_feedback:{$sum:"$No\_of\_feedback"},sum\_of\_types:{$sum :1}}})

Demonstrate the usage of $aggregate

db.movies.aggregate([{$match : {Referable:"True"}},{$group:{\_id:"$Genre",total:{$sum:1}}}])

1. Demonstrate the map-reduce aggregate function on the dataset

> var mapperfunctions = function(){emit(this.Genre,this.No\_of\_feedback)};

> var reducefunctions = function(Genre,No\_of\_feedback){ return Array.sum(No\_of\_feedback);};

> db.movies.mapReduce(mapperfunctions,reducefunctions,{out:"Sumfeedbacks"});

{ "result" : "Sumfeedbacks", "ok" : 1 }

> db.Sumfeedbacks.find().pretty()

1. Count the number of movies which belongs to thriller category and find out the total number of positive reviews in that category

>db.movies.aggregate({$match:{Genre:"Thriller"}},{$group:{\_id:"Genre",Number\_of\_movies:{$sum:1},total\_positive\_feedback:{$sum:"$No\_of\_feedback"}}});

1. Group all the records by genre and find out the total number of positive feedbacks by genre

> db.movies.aggregate({$group: { \_id:"$Genre",sum\_of\_feedback:{$sum:"$No\_of\_feedback"}}})