# **Queries in MongoDB**

* **Select Operations in MongoDB**

|  |
| --- |
| Where clause in Mongodb  db.film.find({film\_id:1}).pretty() |
| Find films with id = 1 or 2  db.film.find({film\_id:{$in:[1,2]}}).pretty() |
| Specifiy AND condition  Select films with release\_year =2006 and rental\_duration = 3  db.film.find({  release\_year:2006 ,  rental\_duration : 3  }) |
| Count the number of films whose rental\_rate is more than $4  db.film.find({  rental\_rate: {$gt: 4}  }).count(); |
| Count the number of films whose rental\_rate is between $2 than $4  db.film.find({  rental\_rate: {$lt: 4 , $gt: 2}  }).count(); |
| Limit the results to the first 5 films  Count the number of films whose rental\_rate is between $2 than $4  db.film.find({  rental\_rate: {$lt: 4 , $gt: 2}  }).limit(5).pretty(); |
| Select Distinct operation  db.film.distinct(“rating”) |
| Projection Operation :  Select title , rating , replacement\_cost from film  db.film.find(  {replacement\_cost : {$lt: 20}} ,  { \_id: 0 , title:1 , rating: 1, replacement\_cost : 1}  ).pretty() |
| Sort the previous result by ascending order of replacement\_cost  db.film.find(  {replacement\_cost : {$lt: 20}} ,  { \_id: 0 , title:1 , rating: 1, replacement\_cost : 1}  ).sort(  {  replacement\_cost: 1  }  ).pretty() |
| Logical OR operation example :  db.film.find( {$or: [{rating:"G"},{rating:"PG"}]} ).count() |

* **Aggregate**

General Syntax

db.<collection>.aggregate( [ { <stage1> }, { <stage2> } ... ] )

|  |
| --- |
| Select count(\*) from film group by rating  db.film.aggregate([{$group : {\_id:"$rating",NumFilms:{$sum:1}}   }]) |
| Select min(replacement\_cost) from film group by rating  db.film.aggregate([{$group : {\_id:"$rating",MinReplacementCost:{$min: "$replacement\_cost"}}   }]) |
|  |

* **InClass Assignment**

|  |
| --- |
| **List the distinct rental durations across all documents in the collection**  db.film.distinct("rental\_duration") |
| **Sort the documents in the descending order of the rental duration**  db.film.find().sort({rental\_duration:-1}) |
| **List the average rental rate by the rental duration**  db.film.aggregate([{$group:{\_id:"$rental\_duration", RentalRate:{$avg:"$rental\_rate"}}}]) |
| **Select film\_id and title the films whose rental\_duration is >= 3 and <= 5 days**  db.film.find({ rental\_duration: {$lte:5, $gte:3} },{\_id:0, title:1} ).pretty(); |
| **List the count of the films whose rental\_duration is >= 3 and <= 5 days**  db.film.find({ rental\_duration: {$lte:5, $gte:3} }).count(); |