1. Create a database called 'my\_first\_db'. = **Use** my\_first\_db
2. Create students collection. = **db.createCollection**("mycollection")
3. Each document you insert into this collection should have the following format: ({name: STRING, home\_state: STRING, lucky\_number: NUMBER, birthday: {month: NUMBER, day: NUMBER, year: NUMBER}})
4. Create 5 students with the appropriate info. **= db.<collection>.insert({key:value, key:value}) =** = **Db.mycollection.insert({name: Glen, home\_state: NY, lucky\_number: 7, birthday: {month: 9, day: 19, year: 1978}})**

> db.mycollection.insert({name: 'Arielle', home\_state: 'WA', lucky\_number: 11, b

irthday: {month: 7, day: 11, year: 1979}})

WriteResult({ "nInserted" : 1 })

> db.mycollection.insert({name: 'Lucy', home\_state: 'WA', lucky\_number: 1, birth

day: {month: 1, day: 1, year: 1111}})

WriteResult({ "nInserted" : 1 })

> db.mycollection.insert({name: 'Linus', home\_state: 'WA', lucky\_number: 2, birt

hday: {month: 2, day: 2, year: 1112}})

WriteResult({ "nInserted" : 1 })

> db.mycollection.insert({name: 'Max', home\_state: 'WA', lucky\_number: 3, birthd

ay: {month: 3, day: 3, year: 1113}})

WriteResult({ "nInserted" : 1 })

1. Get all students. Db.collectionname.find() = **db.mycollection.find()**
2. Retrieve all students who are from California (San Jose Dojo) **or** Washington (Seattle Dojo).

db.mycolection.find ({**$or:[**{"key":"value"},{"key": "value"}**]**}).pretty()

db.mycollection.find({$or:[{"home\_state":"WA"},{ "home\_state":"NY"}]}).pretty(

db.mycollection.find({$or:[{"home\_state": "WA"},{"home\_state":"NY"}]}).pretty()

1. Get all students whose lucky number is:

**greater** than 3

db.mycollection.find({"<key>":{$<condition>:<value>}}).pretty()

db.mycollection.find({"lucky\_number":{$gt:3}}).pretty()

**less than or equal** to 10

db.mycollection.find({"lucky\_number":{$lte:10}}).pretty()

**between** 1 and 9 (inclusive)

db.mycollection.find({$and:[{"lucky\_number": {$gte:1}},{"lucky\_number": {$lte:9}}]}).pretty()

1. Add a field to each student collection called 'interests' that is an ARRAY. It should contain the following entries: 'coding', 'brunch', 'MongoDB'. Do this in ONE operation.

**db.mycollection.updateMany({},{$set: {"interests": [ "coding", "brunch", "MongoDB" ]}})**

1. Add some unique interests for each particular student into each of their interest arrays.

db. **mycollection**.update(

{ "name": "Linus" },

{ $addToSet: { interests: "boxes" } }

)

db. **mycollection**.update(

{ "name": "Lucy" },

{ $addToSet: { interests: "cocktails" } }

)

db. **mycollection**.update(

{ "name": "Glen" },

{ $addToSet: { interests: "rock and roll" } }

)

db. **mycollection**.update(

{ "name": "Arielle" },

{ $addToSet: { interests: "shiney things" } }

)

1. Add the interest 'taxes' into someone's interest array.

db.mycollection.update(

{"name":"Max"},

{$addToSet: {interests: "taxes"}}

)

1. Remove the 'taxes' interest you just added.

db.mycollection.update(

{"name":"Max"},

{ $pull: {interests: "taxes"}}

)

1. Remove all students who are from California (or Washington).

db.mycollection.deleteMany({home\_state: "WA" })

1. Remove a student by name.

db. mycollection.deleteOne( { name: "Glen" })

1. Remove a student whose lucky number is greater than 5 (JUST ONE)

Db.mycollection.deleteOne({"lucky\_number"})

db.mycollection. deleteOne ({"lucky\_number":{$gte:5}})

1. Add a field to each student collection called 'number\_of\_belts' and set it to 0.

db.mycollection.updateMany({},{$set: {"number\_of\_belts": "0"}})

1. Increment this field by 1 for all students in Washington (Seattle Dojo).

db.mycollection.insert({name: "Glen", home\_state: "NY", lucky\_number: 11, birthday: {month: 9, day: 19, year: 1978}})

**db.mycollection.updateMany(**

**{**home\_state:"NY"**},{$inc: {lucky\_number:+1}}**

**)**

1. Rename the 'number\_of\_belts' field to 'belts\_earned'

db.**mycollection**.updateMany( {}, { $rename: { "number\_of\_belts": " belts\_earned " } } )

1. Remove the 'lucky\_number' field.

db.**mycollection**.update({},{$unset: { lucky\_number:1}}, {multi: true})

1. Add a 'updated\_on' field, and set the value as the current date.

db.mycollection.updateMany({},{$set: {updated\_on: new Date()}})