# Hands on Mongodb

Source: <https://www.youtube.com/watch?v=ynPAkZyH3R8>

## Basic Commands

* To view all the existing databases

show databases

* Switching to db hockey, if don't exist create mongo creates it, once data is added after switching to not-existing database.

use hockey

* To show name of current working database

db

* Deleting current database

db.dropDatabase()

* Inserting single record

db.players.insert(

{

"position":"Right Wing",

"id":8465166,

"weight":200,

"height":"6' 0\"",

"imageUrl":"<http://1.cdn.nhle.com/photos/mugs/8465166.jpg>",

"birthplace":"Seria, BRN",

"age":37,

"name":"Craig Adams",

"birthdate":"April 26, 1977",

"number":27

}

)

* Inserting multiple records. You need to provide data in array, represented with jsondatas in between [ ] brackets.

db.players.insert([

{

"position":"Right Wing",

"id":8475761,

"weight":195,

"height":"6' 2\"",

"imageUrl":"<http://1.cdn.nhle.com/photos/mugs/8475761.jpg>",

"birthplace":"Gardena, CA, USA",

"age":23,

"name":"Beau Bennett",

"birthdate":"November 27, 1991",

"number":19

},

{

"position":"Left Wing",

"id":8471260,

"weight":202,

"height":"6' 1\"",

"imageUrl":"<http://3.cdn.nhle.com/photos/mugs/8471260.jpg>",

"birthplace":"Meadow Lake, SK, CAN",

"age":29,

"name":"BlakeComeau",

"birthdate":"February 18, 1986",

"number":17

}

])

* Listing collections in current database

show collections

* To retrieve all data in players database (equivalent to select \* from players in RDBMS)

db.payers.find()

* For formatted output, use pretty() method

db.players.find().pretty()

* finds the first record in players collection

db.players.findOne()

* Remove record from players collection where \_id is ObjectId("55a476532e6d8e4b3102fc84")

db.players.remove(

{"\_id" :ObjectId("55a476532e6d8e4b3102fc84")}

)

* Update record for the \_id :ObjectId("55a496291ec1e1108434aecf")

db.players.update(

{"\_id" :ObjectId("55a496291ec1e1108434aecf")},

{

"position":"Goalie",

"id":8471300,

"weight":220,

"height":"6' 1\"",

"imageUrl":"<http://1.cdn.nhle.com/photos/mugs/8471306.jpg>",

"birthplace":"Fussen, DEU",

"age":29,

"name":"ThomasGreiss",

"birthdate":"January 29, 1986",

"number":1

}

)

* To delete players collection

db.players.drop()

## Working with Find Command

* Find record in players collection where position is Goalie

db.players.find(

{"position":"Goalie"}

)

* Find first record in players collection where position is Goalie

db.players.findOne(

{"position":"Goalie"}

)

* Find Document in Collection where position is “Goalie” and age is 21 years.

db.players.find(

{"position":"Goalie","age":21},

)

* Or operation: Find all Documents or records in Collection where position is “Left Wing” or “Right Wing”.

db.players.find(

{

$or:[

{"position":"Left Wing"},

{"position":"Right Wing"}

]

}

* Find all players where age is greater than 30

db.players.find(

{"age":{$gt:30}}

)

* Find all players where age is greater or equals to 30

db.players.find(

{"age":{$gte:30}}

)

* Display selected fields, 1 for true and 0 for false

db.players.find(

{"position":"Center"},

{"name":1,\_id:0}

)

* Only display 3 documents (records)

db.players.find(

{"position":"Center"},

{"name":1,\_id:0}

).limit(3)

* Skips first 2 Documents or records

db.players.find(

{"position":"Center"},

{"name":1,\_id:0}

).skip(2)

* To display second record only

db.players.find(

{"position":"Center"},

{"name":1,\_id:0}

).skip(1).limit(1)

## Aggregation function

* To find sum of all records grouped by eyeColors field

db.users.aggregate({

$group : {

\_id: "$eyeColor",

total : {$sum: 1}

}

})

* To find average of all records grouped by eyeColors field

db.users.aggregate({

$group : {

\_id: "$gender",

avg : {$avg: "$age"}

}

})

* To find record with highest balance grouped by gender

db.users.aggregate({

$group : {

\_id: "$gender",

richest : {$max: "$balance"}

}

})