

Assignment: CRUD Operations

Date: 07/10/2020 Name: D.Saravanan

1. Create a collection, insert 6 documents, perform all the CRUD operations and display the output.

Insert

```
> show dbs
admin    0.000GB
config  0.000GB
local    0.000GB
module   0.000GB
> use assignment
switched to db assignment
> db
assignment
> show collections
> db.createCollection("Schedule", {capped: true, size: 250000, max: 6})
{ "ok" : 1 }
> show collections
Schedule
> db.Schedule.insert({Date: "10/07/2017", Time: "8:00 am - 12:00 pm", RoomNo: 106,
... Title: "Software Carpentry Scientific Python", Speaker: "Maxim Belkin", Affiliation: "Software Carpentry"})
WriteResult({ "nInserted" : 1 })
> db.Schedule.insertMany([
... {Date: "10/07/2017", Time: "8:00 am - 12:00 pm", RoomNo: 106,
... Title: "Modern Optimization Methods in Python", Speaker: "Michael McKerns", Affiliation: "UQ Foundation"},
... {Date: "10/07/2017", Time: "1:30 pm - 5:30 pm", RoomNo: 106,
... Title: "Automatic Code Generation with SymPy", Speaker: ["Jason Moore", "Aaron Meurer", "Bjorn Dahlgren", "Kenneth Lyons"]},
... Affiliation: ["PyPy and SymPy", "University of South Carolina", "KTH Royal Institute of Technology", "University of California Davis"]},
... {Date: "11/07/2017", Time: "8:00 am - 12:00 pm", RoomNo: 106,
... Title: "Computational Statistics", Speaker: "Allen Downey", Affiliation: "Olin College"},
... {Date: "11/07/2017", Time: "8:00 am - 12:00 pm", RoomNo: 101,
... Title: "Parallel Data Analysis in Python", Speaker: ["Matthew Rocklin", "Ben Zaitlen", "Aron Ahmadi"]},
... Affiliation: ["Continuum Analytics", "Continuum Analytics", "Capital One"]},
... {Date: "11/07/2017", Time: "1:30 pm - 5:30 pm", RoomNo: 106,
... Title: "Network Science and Statistics", Speaker: "Eric Ma", Affiliation: "MIT"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("5f7dec9423412aac4e904756"),
    ObjectId("5f7dec9423412aac4e904757"),
    ObjectId("5f7dec9423412aac4e904758"),
    ObjectId("5f7dec9423412aac4e904759"),
    ObjectId("5f7dec9423412aac4e90475a")
  ]
}
```

Query and Projection

```
> show collections
Schedule
> db.Schedule.find({RoomNo: 106})
{ "_id" : ObjectId("5f7dec9423412aac4e904756"), "Date" : "10/07/2017", "Time" : "8:00 am - 12:00 pm", "RoomNo" : 106, "Title" : "Modern Optimization Methods in Python", "Speaker" : "Michael McKerns", "Affiliation" : "UQ Foundation" }
{ "_id" : ObjectId("5f7dec9423412aac4e90475a"), "Date" : "11/07/2017", "Time" : "1:30 pm - 5:30 pm", "RoomNo" : 106, "Title" : "Network Science and Statistics", "Speaker" : "Eric Ma", "Affiliation" : "MIT" }
> db.Schedule.find({RoomNo: 106}).pretty()
{
  "_id" : ObjectId("5f7dec9423412aac4e904756"),
  "Date" : "10/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "RoomNo" : 106,
  "Title" : "Modern Optimization Methods in Python",
  "Speaker" : "Michael McKerns",
  "Affiliation" : "UQ Foundation"
}
{
  "_id" : ObjectId("5f7dec9423412aac4e90475a"),
  "Date" : "11/07/2017",
  "Time" : "1:30 pm - 5:30 pm",
  "RoomNo" : 106,
  "Title" : "Network Science and Statistics",
  "Speaker" : "Eric Ma",
  "Affiliation" : "MIT"
}
> db.Schedule.findOne({RoomNo: 106})
{
  "_id" : ObjectId("5f7dec9423412aac4e904756"),
  "Date" : "10/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "RoomNo" : 106,
  "Title" : "Modern Optimization Methods in Python",
  "Speaker" : "Michael McKerns",
  "Affiliation" : "UQ Foundation"
}
> db.Schedule.find({RoomNo: 101, RoomNo: 106}, {"_id": 0, "Date": 1, "Time": 1, "Title": 1, "Speaker": 1})
{ "Date" : "10/07/2017", "Time" : "8:00 am - 12:00 pm", "Title" : "Modern Optimization Methods in Python", "Speaker" : "Michael McKerns" }
{ "Date" : "11/07/2017", "Time" : "1:30 pm - 5:30 pm", "Title" : "Network Science and Statistics", "Speaker" : "Eric Ma" }
> db.Schedule.findOne({RoomNo: 106}, {"_id": 0, "Date": 1, "Time": 1, "Title": 1, "Speaker": 1})
{
  "Date" : "10/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "Title" : "Modern Optimization Methods in Python",
  "Speaker" : "Michael McKerns"
}
> db.Schedule.findOne({Speaker: "Kenneth Lyons"}, {"_id": 0, "RoomNo": 1, "Title": 1})
{ "RoomNo" : 103, "Title" : "Automatic Code Generation with SymPy" }
> db.Schedule.findOne({Title: "Computational Statistics"}, {"_id": 0, "Speaker": 1, "Affiliation": 1})
{ "Speaker" : "Allen Downey", "Affiliation" : "Olin College" }
>
```

Update

```
> db.Schedule.find({RoomNo: 103}).pretty()
{
  "_id" : ObjectId("5f7de4c823412aac4e904755"),
  "Date" : "10/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "RoomNo" : 103,
  "Title" : "Software Carpentry Scientific Python",
  "Speaker" : "Maxim Belkin",
  "Affiliation" : "Software Carpentry"
}

{
  "_id" : ObjectId("5f7de4c9423412aac4e904757"),
  "Date" : "10/07/2017",
  "Time" : "1:30 pm - 5:30 pm",
  "RoomNo" : 103,
  "Title" : "Automatic Code Generation with SymPy",
  "Speaker" : [
    "Jason Moore",
    "Aaron Meurer",
    "Bjorn Dahlgren",
    "Kenneth Lyons"
  ],
  "Affiliation" : [
    "PyPy and SymPy",
    "University of South Carolina",
    "KTH Royal Institute of Technology",
    "University of California Davis"
  ]
}
]
> db.Schedule.update({RoomNo: 103}, {$set: {RoomNo: 105}}, {multi: false})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Schedule.find({RoomNo: 105}).pretty()
{
  "_id" : ObjectId("5f7de4c823412aac4e904755"),
  "Date" : "10/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "RoomNo" : 105,
  "Title" : "Software Carpentry Scientific Python",
  "Speaker" : "Maxim Belkin",
  "Affiliation" : "Software Carpentry"
}

{
  "_id" : ObjectId("5f7de4c9423412aac4e904758"),
  "Date" : "11/07/2017",
  "Time" : "8:00 am - 12:00 pm",
  "RoomNo" : 105,
  "Title" : "Computational Statistics",
  "Speaker" : "Allen Downey",
  "Affiliation" : "Olin College"
}
]>
```

Delete

```
> db.Schedule.find({}, { _id: 0, Date: 1, RoomNo: 1, Title: 1, Speaker: 1 })
{ "Date" : "10/07/2017", "RoomNo" : 105, "Title" : "Software Carpentry Scientific Python", "Speaker" : "Maxim Belkin" }
{ "Date" : "10/07/2017", "RoomNo" : 106, "Title" : "Modern Optimization Methods in Python", "Speaker" : "Michael McKerns" }
{ "Date" : "10/07/2017", "RoomNo" : 103, "Title" : "Automatic Code Generation with SymPy", "Speaker" : [ "Jason Moore", "Aaron Meurer", "Bjorn Dahlgren", "Kenneth Lyons" ] }
{ "Date" : "11/07/2017", "RoomNo" : 105, "Title" : "Computational Statistics", "Speaker" : "Allen Downey" }
{ "Date" : "11/07/2017", "RoomNo" : 101, "Title" : "Parallel Data Analysis in Python", "Speaker" : [ "Matthew Rocklin", "Ben Zaitlen", "Aron Ahmadi" ] }
{ "Date" : "11/07/2017", "RoomNo" : 106, "Title" : "Network Science and Statistics", "Speaker" : "Eric Ma" }
> db.Schedule.remove({Speaker: "Eric Ma"})
WriteResult({
  "nRemoved" : 0,
  "writeError" : {
    "code" : 20,
    "errmsg" : "cannot remove from a capped collection: assignment.Schedule"
  }
})
> db.Schedule.remove({RoomNo: 105}, true)
WriteResult({
  "nRemoved" : 0,
  "writeError" : {
    "code" : 20,
    "errmsg" : "cannot remove from a capped collection: assignment.Schedule"
  }
})
> db.Schedule.isCapped()
true
> db.Schedule.remove({})
WriteResult({
  "nRemoved" : 0,
  "writeError" : {
    "code" : 20,
    "errmsg" : "cannot remove from a capped collection: assignment.Schedule"
  }
})
> db.Schedule.find({}, { _id: 0, Date: 1, RoomNo: 1, Title: 1, Speaker: 1 })
{ "Date" : "10/07/2017", "RoomNo" : 105, "Title" : "Software Carpentry Scientific Python", "Speaker" : "Maxim Belkin" }
{ "Date" : "10/07/2017", "RoomNo" : 106, "Title" : "Modern Optimization Methods in Python", "Speaker" : "Michael McKerns" }
{ "Date" : "10/07/2017", "RoomNo" : 103, "Title" : "Automatic Code Generation with SymPy", "Speaker" : [ "Jason Moore", "Aaron Meurer", "Bjorn Dahlgren", "Kenneth Lyons" ] }
{ "Date" : "11/07/2017", "RoomNo" : 105, "Title" : "Computational Statistics", "Speaker" : "Allen Downey" }
{ "Date" : "11/07/2017", "RoomNo" : 101, "Title" : "Parallel Data Analysis in Python", "Speaker" : [ "Matthew Rocklin", "Ben Zaitlen", "Aron Ahmadi" ] }
{ "Date" : "11/07/2017", "RoomNo" : 106, "Title" : "Network Science and Statistics", "Speaker" : "Eric Ma" }
>
```