Redovisa svar, querys samt länka till Mongos officiella dokumentation där operatorer eller funktioner som används i aktuell query presenteras.

**Databas: DB2-1   
Aktuell samling: citys**

1. Vilken stad har id 01026 ?

Query: db.citys.find({\_id: "01026"}) för att visa den staden som id 01026 befinner sig

Svar: city: 'CUMMINGTON'

<https://docs.mongodb.com/manual/reference/method/db.collection.find/#mongodb-method-db.collection.find>

1. Vilken stad befinner sig på lat/long: -96.734103, 33.449513 ?

Query: db.citys.find({loc: [-96.734103, 33.449513 ]}) för att visa staden som är med location : -96.734103, 33.449513

Svar: city: 'GUNTER'

<https://docs.mongodb.com/manual/tutorial/query-arrays/>

1. Hur många städer börjar på bokstaven P ?

Query: db.citys.find({city:{$regex:/^P/i}}).count() för att räkna ut städerna börjar på bokstaven “P”

Svar:1681

<https://docs.mongodb.com/manual/reference/operator/query/regex/>

<https://docs.mongodb.com/manual/reference/method/db.collection.count/#mongodb-method-db.collection.count>

1. Hur många städer har en befolkningsmängd under 50.000 ?

Query: db.citys.find({pop: {$lt: 50000}}).count() för att räkna ut städerna som har en befolkningsmängd under 50000

Svar: 28904

<https://docs.mongodb.com/manual/reference/operator/query/lt/#mongodb-query-op.-lt>

1. Lista endast 5 stycken städer med befolkningsmängd över 50.000 i fallande ordning?

Query: db.citys.find({pop: {$gt: 50000}}).limit(5).sort({pop:-1}) för att hämta ut 5 stycken städer med befolkningsmängd över 50000 i fallande ordning

Svar: city: 'CHICAGO', city: 'BROOKLYN’, city: 'NEW YORK', city: 'NEW YORK', city: 'BELL GARDENS'}

<https://docs.mongodb.com/manual/reference/operator/query/gt/>

<https://docs.mongodb.com/manual/reference/operator/aggregation/limit/>

<https://docs.mongodb.com/manual/reference/operator/update/sort/>

1. Vilken population (pop) har städerna med namnet SHEFFIELD, observera att svaret på din query skall endast innehålla just attributet pop *(Projektion används)*.

Query: db.citys.find({city: "SHEFFIELD"},{\_id:0,pop:1}) för att visa populationer har staden med namnet ”SHEFFIELD”

Svar: pop: 1839, pop: 600, pop: 2382, pop: 10685, pop: 1988, pop: 1964, pop: 428

<https://docs.mongodb.com/manual/reference/method/db.collection.find/#std-label-method-find-projection>

**Databas: DB2-2  
Aktuella samlingar: ”Samtliga”**

1. Visa order med OrderID 10267.*(Aggregation skall användas).*

Query: db.orders.aggregate({$match:{OrderID:10267}}) för att visa informationer av OrderID 10267

Svar: { \_id: ObjectId("5c62cb2c4f1ca7422ce2a96d"),

OrderID: 10267,

CustomerID: 'FRANK',

EmployeeID: 4,

OrderDate: '1996-07-29 00:00:00.000',

RequiredDate: '1996-08-26 00:00:00.000',

ShippedDate: '1996-08-06 00:00:00.000',

ShipVia: 1,

Freight: 208.58,

ShipName: 'Frankenversand',

ShipAddress: 'Berliner Platz 43',

ShipCity: 'München',

ShipRegion: 'NULL',

ShipPostalCode: 80805,

ShipCountry: 'Germany' }

<https://docs.mongodb.com/manual/reference/operator/aggregation/match/#mongodb-pipeline-pipe.-match>

1. Hämta order med OrderID 10267 samt inkludera (”join”) dokumentet som beskriver kunden (CustomerID ). *(Aggregation skall användas).*

Query: db.orders.aggregate({$match:{OrderID: 10267}},{$lookup: {from: "customers",localField: "CustomerID",foreignField: "OrderID",as: "order10267\_customer"}},{$project:{\_id:0,CustomerID:1}}) för att hämta Customer ID och att gömma id samtidigt

Svar: CustomerID: 'FRANK'

[*https://docs.mongodb.com/manual/reference/operator/aggregation/lookup/*](https://docs.mongodb.com/manual/reference/operator/aggregation/lookup/)

[*https://docs.mongodb.com/manual/reference/operator/aggregation/project/#mongodb-pipeline-pipe.-project*](https://docs.mongodb.com/manual/reference/operator/aggregation/project/#mongodb-pipeline-pipe.-project)

1. Hämta order med OrderID 10267 samt 10259 och inkludera (”join”) dokumentet som beskriver kunden i båda. *(Aggregation skall användas)*

Query: db.orders.aggregate({$match:{$or:[{OrderID:{$eq:10267}},{OrderID:{$eq:10259}}]}},{$lookup:{from:"customers",localField:"CustomerID",foreignField:"OrderID", as:"orders\_customer"}},{$project:{\_id:1,CustomerID:1}}) för att hämta både id och CustomerID

Svar: { \_id: ObjectId("5c62cb2c4f1ca7422ce2a964"),

CustomerID: 'CENTC' }

{ \_id: ObjectId("5c62cb2c4f1ca7422ce2a96d"),

CustomerID: 'FRANK' }

<https://docs.mongodb.com/manual/reference/operator/query/or/>

<https://docs.mongodb.com/manual/reference/operator/aggregation/eq/>

1. Hur många kunder kommer antingen ifrån Sverige eller Mexico?

Query: db.customers.find({$or:[{Country:{$eq: "Sweden"}},{Country:{$eq: "Mexico"}}]}).count()

Svar: 7

**Databas: DB2-3  
Aktuell samling: ”Du har en tilldelad samling i databasen”**

1. Uppdatera valfritt attribut i dokumentet som finns i samlingen.

Query: db.s16.updateOne({\_id: ObjectId("61f7a2ffbdd65f37689e00c7")},{$set:{happy:false}}) för att ändra true mot false på attribut ”happy” i \_id:ObjectId(("61f7a2ffbdd65f37689e00c7")

Svar: { acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0 }

<https://docs.mongodb.com/manual/reference/operator/update/positional/#update-embedded-documents-using-multiple-field-matches>

1. Lägg till ett valfritt attribut med ett värde i dokumentet som finns i samlingen.

Query: db.s16.insertMany([{today:true},{feeling:"good"}]) för att lägga till flera värde i samlingen

Svar: { \_id: ObjectId("61f7a2ffbdd65f37689e00c7"),

happy: false,

weather: 'sunny' }

{ \_id: ObjectId("621b539ef4783e4cb255f57e"), today: true }

{ \_id: ObjectId("621b539ef4783e4cb255f57f"), feeling: 'good' }

<https://docs.mongodb.com/manual/reference/method/db.collection.insertMany/#examples>

1. Lägg till ett nytt dokument i samlingen. Dokumentet skall beskriva din omgivning och innehålla minst en av varje av följande datatyper: String, Number, Boolean, Array samt minst ett nästlat dokument (embedded).

Query: db.s16.insertMany([{\_id: ObjectId("61f7a2ffbdd65f37689e00c8"),instock:[{name:"Emily", age:25, study:true}]},{\_id:ObjectId("21f7a2ffbdd65f37689e00c8"),instock:[{name:"Hark", age:21, study:false}]}]) för att lägga till ett nytt dokument som innebär att en Array sparar data typer string i name, number i age and boolean i study

Svar: { acknowledged: true,

insertedIds:

{ '0': ObjectId("61f7a2ffbdd65f37689e00c8"),

'1': ObjectId("21f7a2ffbdd65f37689e00c8") } }

db.s16.find({})

{ \_id: ObjectId("61f7a2ffbdd65f37689e00c7"),

happy: false,

weather: 'sunny' }

{ \_id: ObjectId("621b539ef4783e4cb255f57e"), today: true }

{ \_id: ObjectId("621b539ef4783e4cb255f57f"), feeling: 'good' }

{ \_id: ObjectId("61f7a2ffbdd65f37689e00c8"),

instock: [ { name: 'Emily', age: 25, study: true } ] }

{ \_id: ObjectId("21f7a2ffbdd65f37689e00c8"),

instock: [ { name: 'Hark', age: 21, study: false } ] }

<https://docs.mongodb.com/manual/tutorial/query-array-of-documents/>