

Name:- Raviteja Kommalapati

SJSUID:- 012526358

Lab 4 Solution

- EC2 Instance Dashboard showing the running Mongo Instances with their Public IP

Running Instances

The screenshot displays the AWS EC2 Management Console. On the left, the navigation menu includes sections like EC2 Dashboard, INSTANCES, ELASTIC BLOCK STORE, and NETWORK & SECURITY. The main area shows a table of instances:

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)
mongo-secondary1	i-03155abd50ad09df	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-34-218-175-7
mongo-secondary2	i-0403453a7095aab...	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-161-232-7
mongo-primary	i-06e4ccd6e596552cf	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-35-161-242-1
Lab4	i-0302f599e2a3fa532	t2.micro	us-west-2a	running	2/2 checks ...	None	ec2-52-41-27-239
	i-012637266dc01a2ca	t2.micro	us-west-2a	stopped		None	

The 'mongo-primary' instance is selected. The details pane on the right shows:

- Instance: i-06e4ccd6e596552cf (mongo-primary) Elastic IP: 35.161.242.101
- Tab: Description
- Instance ID: i-06e4ccd6e596552cf
- Instance state: running
- Instance type: t2.micro
- Elastic IPs: 35.161.242.101*
- Availability zone: us-west-2a
- Security groups: mongodb-cluster, view inbound rules, view outbound rules
- Scheduled events: No scheduled events
- Public DNS (IPv4): ec2-35-161-242-101.us-west-2.compute.amazonaws.com
- IPv4 Public IP: 35.161.242.101
- IPv6 IPs: -
- Private DNS: ip-10-0-0-86.us-west-2.compute.internal
- Private IPs: 10.0.0.86
- Secondary private IPs: -
- VPC ID: vpc-093abca85ffc477fd

```
...ntu@ip-10-0-0-86: ~ — ssh -i cmpe281uswest2.pem ubuntu@35.161.242.101
GNU nano 2.8.3
127.0.0.1 localhost
35.161.242.101 primary
34.218.175.76 secondary1
35.161.232.76 secondary2

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe80::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
File: /etc/hosts
```

rs.status() log output

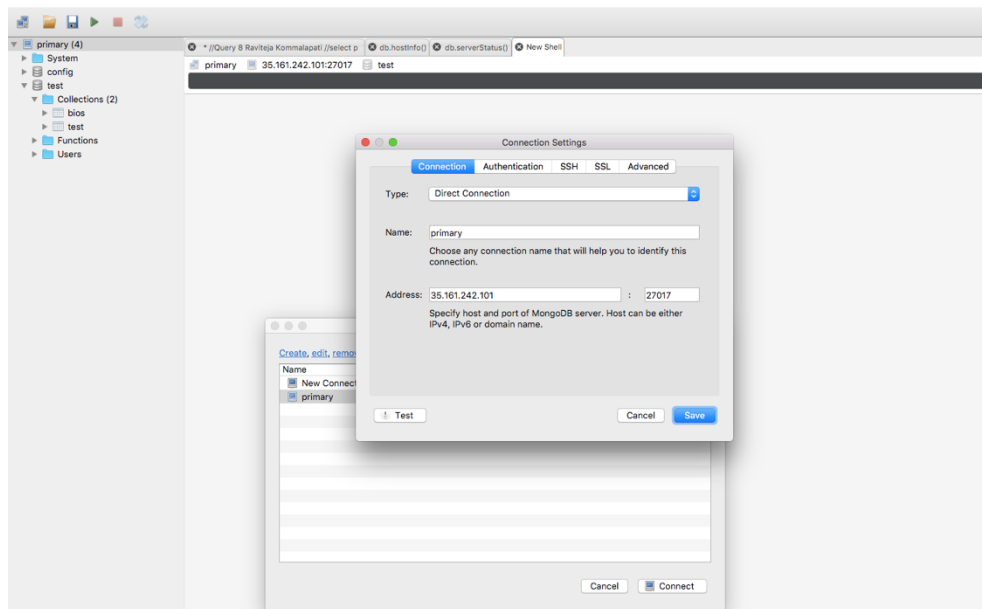
```
...ntu@ip-10-0-0-86: ~ — ssh -i cmpe281uswest2.pem ubuntu@35.161.242.101
{
  "_id" : 0,
  "name" : "primary:27017",
  "health" : 1,
  "state" : 1,
  "stateStr" : "PRIMARY",
  "uptime" : 16588,
  "optime" : {
    "ts" : Timestamp(1551000085, 1),
    "t" : NumberLong(1)
  },
  "optimeDate" : ISODate("2019-02-24T11:34:45Z"),
  "syncingTo" : "",
  "syncSourceHost" : "",
  "syncSourceId" : -1,
  "infoMessage" : "",
  "electionTime" : Timestamp(1551001144, 1),
  "electionDate" : ISODate("2019-02-24T09:39:04Z"),
  "configVersion" : 1,
  "self" : true,
  "lastHeartbeatMessage" : ""
},
{
  "_id" : 1,
  "name" : "secondary1:27017",
  "health" : 1,
  "state" : 2,
  "stateStr" : "SECONDARY",
  "uptime" : 8612,
  "optime" : {
    "ts" : Timestamp(1551000085, 1),
    "t" : NumberLong(1)
  },
  "optimeDurable" : {
    "ts" : Timestamp(1551000085, 1),
    "t" : NumberLong(1)
  },
  "optimeDate" : ISODate("2019-02-24T11:34:45Z"),
  "optimeDurableDate" : ISODate("2019-02-24T11:34:45Z"),
  "lastHeartbeat" : ISODate("2019-02-24T11:34:48.928Z"),
  "lastHeartbeatRecv" : ISODate("2019-02-24T11:34:48.927Z"),
  "pingMs" : NumberLong(0),
  "lastHeartbeatMessage" : "",
  "syncingTo" : "secondary2:27017",
  "syncSourceHost" : "secondary2:27017",
  "syncSourceId" : 2,
  "infoMessage" : "",
  "configVersion" : 1
},
{
  "_id" : 2,
  "name" : "secondary2:27017",
  "health" : 1,
  "state" : 2,
  "stateStr" : "SECONDARY",
  "uptime" : 8612,
  "optime" : {
    "ts" : Timestamp(1551000085, 1),
    "t" : NumberLong(1)
  },
  "optimeDurable" : {
    "ts" : Timestamp(1551000085, 1),
    "t" : NumberLong(1)
  }
}
```

MongoDB Shell Connections to your Mongo Cluster along with each of the 10 Mongo Queries with Results

Connecting mongodb in instance

```
...ntu@ip-10-0-0-86: ~ -- ssh -i cmpe281uswest2.pem ubuntu@35.161.242.101 ...tu@ip-10-0-0-32: ~ -- ssh -i cmpe281uswest2.pem ubuntu@34.218.175.76 ...@ip-10-0-0-25: ~ --
ubuntu@ip-10-0-0-86:~$ mongo
MongoDB shell version v4.0.6
connecting to: mongodb://127.0.0.1:27017/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("090117a4-4344-4c6d-8b84-143c8c16ec33") }
MongoDB server version: 4.0.6
cmpe281:PRIMARY> exit
bye
ubuntu@ip-10-0-0-86:~$ mongo -u admin -p admin --authenticationDatabase admin;
MongoDB shell version v4.0.6
connecting to: mongodb://127.0.0.1:27017/?authSource=admin&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("0c7f1ebc-b3ef-4012-b86a-dea46997a459") }
MongoDB server version: 4.0.6
Server has startup warnings:
2019-02-24T06:58:21.969+0000 I STORAGE [initandlisten]
2019-02-24T06:58:21.969+0000 I STORAGE [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2019-02-24T06:58:21.969+0000 I STORAGE [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten]
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/enabled is 'always'.
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten] ** We suggest setting it to 'never'
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten]
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugepage/defrag is 'always'.
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten] ** We suggest setting it to 'never'
2019-02-24T06:58:24.222+0000 I CONTROL [initandlisten]
cmpe281:PRIMARY>
```

Connecting from Robo3T



Query1 Result

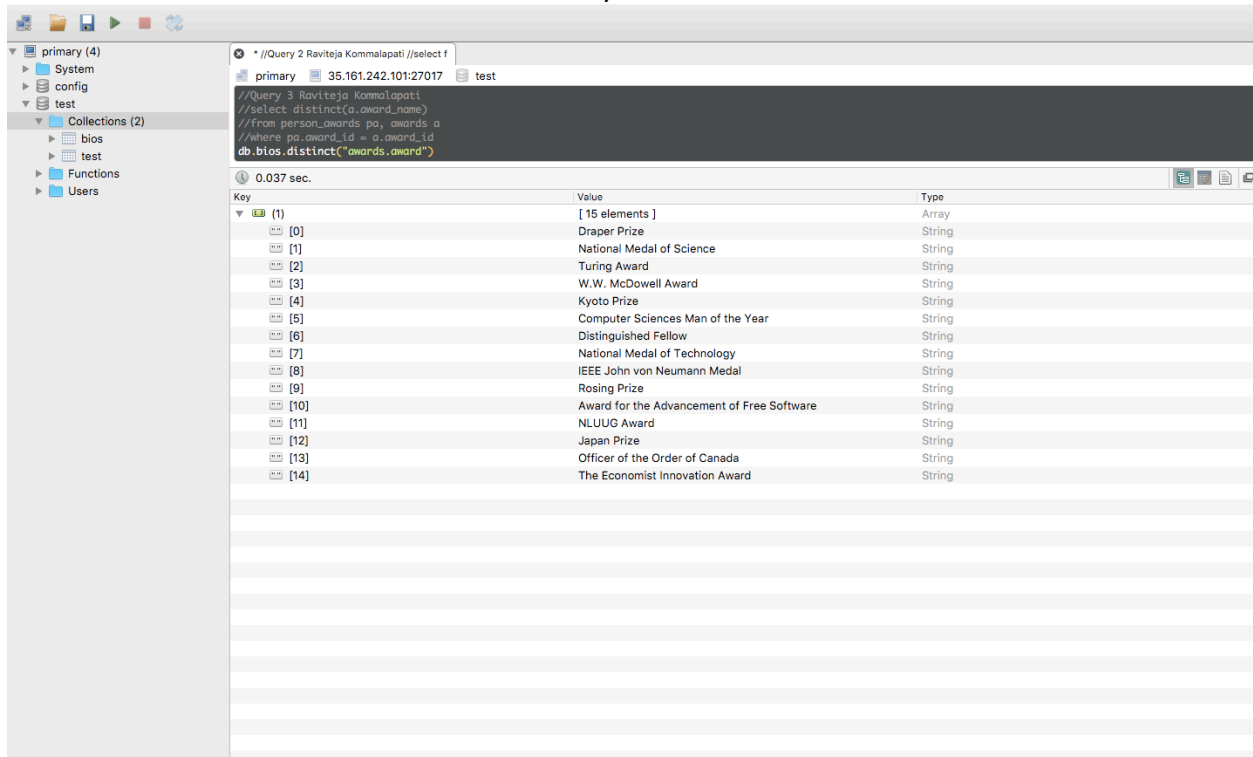
The screenshot shows the MongoDB Compass interface. On the left, the database structure is visible: 'primary (4)' containing 'System', 'config', 'test', and 'Collections (2)' which includes 'bios' and 'test'. The main panel shows the command `db.bios.count()` executed against the 'primary' database at `35.161.242.101:27017` in the 'test' namespace. The execution time was `0.037 sec.` and the result is `10`.

Query 2 Result

The screenshot shows the MongoDB Compass interface with a query executed on the 'bios' collection. The query is `db.bios.find({birth:{$lt:new ISODate("1950-01-01T20:00:00Z")}}, {name:1,birth:1,_id:0})`. The execution time was `0.037 sec.`. The result is a JSON array of 5 documents, each containing a 'name' object with 'first' and 'last' fields, and a 'birth' ISODate field.

```
/* 1 */
{
  "name" : {
    "first" : "John",
    "last" : "Backus"
  },
  "birth" : ISODate("1924-12-03T05:00:00.000Z")
}
/* 2 */
{
  "name" : {
    "first" : "John",
    "last" : "McCarthy"
  },
  "birth" : ISODate("1927-09-04T04:00:00.000Z")
}
/* 3 */
{
  "name" : {
    "first" : "Grace",
    "last" : "Hopper"
  },
  "birth" : ISODate("1906-12-09T05:00:00.000Z")
}
/* 4 */
{
  "name" : {
    "first" : "Kristen",
    "last" : "Nygaard"
  },
  "birth" : ISODate("1926-08-27T04:00:00.000Z")
}
/* 5 */
{
  "name" : {
    "first" : "Ole-Johan",
```

Query 3 Result



primary (4)
System
config
test
Collections (2)
bios
test
Functions
Users

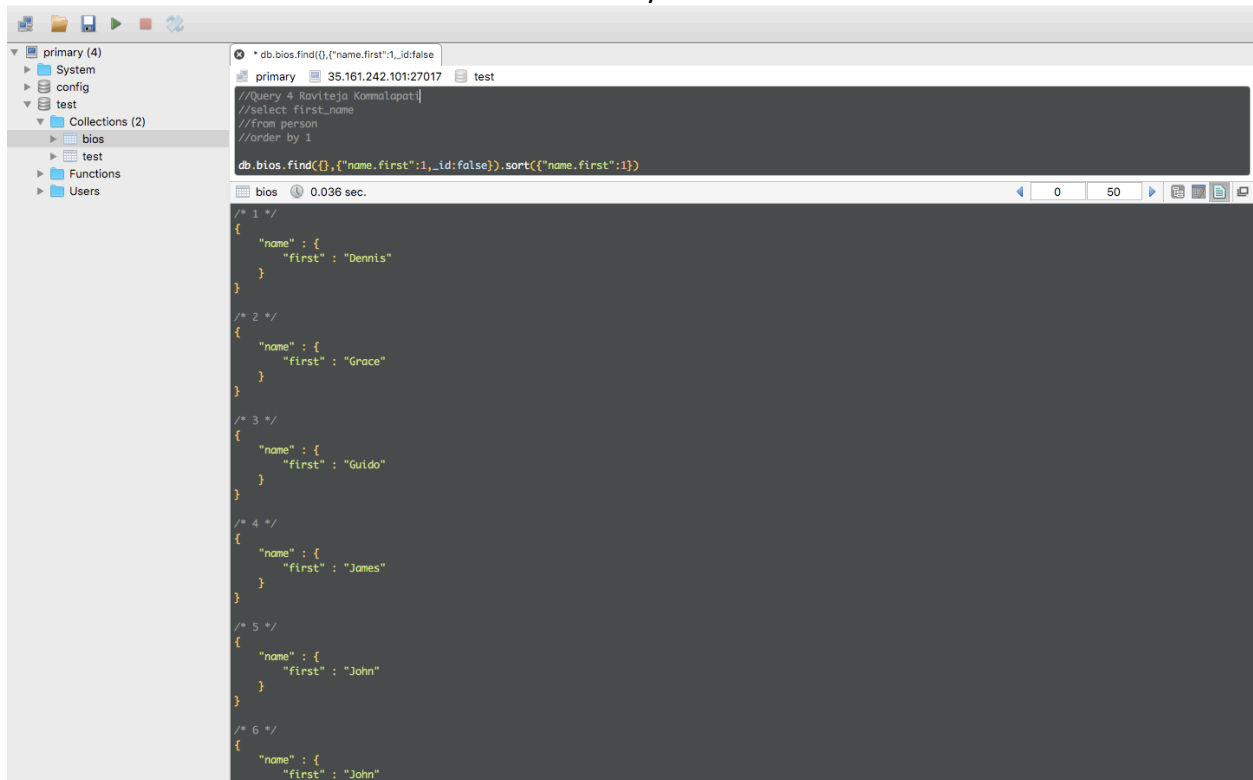
Query 3 Raviteja Kommalapati //select f
primary 35.161.242.101:27017 test

```
//Query 3 Raviteja Kommalapati  
//select distinct(a.award_name)  
//from person_awards pa, awards a  
//where pa.award_id = a.award_id  
db.bios.distinct("awards.award")
```

0.037 sec.

Key	Value	Type
(1)	[15 elements]	Array
[0]	Draper Prize	String
[1]	National Medal of Science	String
[2]	Turing Award	String
[3]	W.W. McDowell Award	String
[4]	Kyoto Prize	String
[5]	Computer Sciences Man of the Year	String
[6]	Distinguished Fellow	String
[7]	National Medal of Technology	String
[8]	IEEE John von Neumann Medal	String
[9]	Rosing Prize	String
[10]	Award for the Advancement of Free Software	String
[11]	NLUUG Award	String
[12]	Japan Prize	String
[13]	Officer of the Order of Canada	String
[14]	The Economist Innovation Award	String

Query 4



primary (4)
System
config
test
Collections (2)
bios
test
Functions
Users

Query 4 Raviteja Kommalapati
primary 35.161.242.101:27017 test

```
//Query 4 Raviteja Kommalapati  
//select first_name  
//from person  
//order by 1  
db.bios.find({"name.first":1,_id:false}).sort({"name.first":1})
```

bios 0.036 sec.

```
/* 1 */  
{  
  "name" : {  
    "first" : "Dennis"  
  }  
}  
/* 2 */  
{  
  "name" : {  
    "first" : "Grace"  
  }  
}  
/* 3 */  
{  
  "name" : {  
    "first" : "Guido"  
  }  
}  
/* 4 */  
{  
  "name" : {  
    "first" : "James"  
  }  
}  
/* 5 */  
{  
  "name" : {  
    "first" : "John"  
  }  
}  
/* 6 */  
{  
  "name" : {  
    "first" : "John"  
  }  
}
```

Query5 Result

The screenshot shows the MongoDB Compass interface. On the left, the database structure is visible: 'primary (4)' containing 'System', 'config', 'test', and 'Users'. Under 'test', there are 'Collections (2)' including 'bios' and 'test'. The 'bios' collection is selected. The main panel displays the query: `db.bios.find({}, {"name.first":1, "_id":false}).sort({"name.first":-1})`. The results are shown as a list of 6 documents, each with a 'name' field containing 'first' and 'last' names. The documents are sorted by 'first' name in descending order. The execution time is 0.058 sec.

```
//Query 4 Raviteja Kommalapati //select f
primary 35.161.242.101:27017 test

//Query 4 Raviteja Kommalapati
//select first_name
//from person
//order by 1 desc

db.bios.find({}, {"name.first":1, "_id":false}).sort({"name.first":-1})

bios 0.058 sec.

/* 1 */
{
  "name" : {
    "first" : "Yukihiro"
  }
}

/* 2 */
{
  "name" : {
    "first" : "Ole-Johan"
  }
}

/* 3 */
{
  "name" : {
    "first" : "Martin"
  }
}

/* 4 */
{
  "name" : {
    "first" : "Kristen"
  }
}

/* 5 */
{
  "name" : {
    "first" : "John"
  }
}

/* 6 */
{
  "name" : {
    "first" : "John"
  }
}
```

Query 6 Result

The screenshot shows the MongoDB Compass interface. The database structure is the same as in the previous screenshot. The 'bios' collection is selected. The main panel displays the query: `db.bios.count({awards:null}, {_id:false})`. The results show a single document with the value '1'. The execution time is 0.036 sec.

```
* //Query 4 Raviteja Kommalapati //select f
primary 35.161.242.101:27017 test

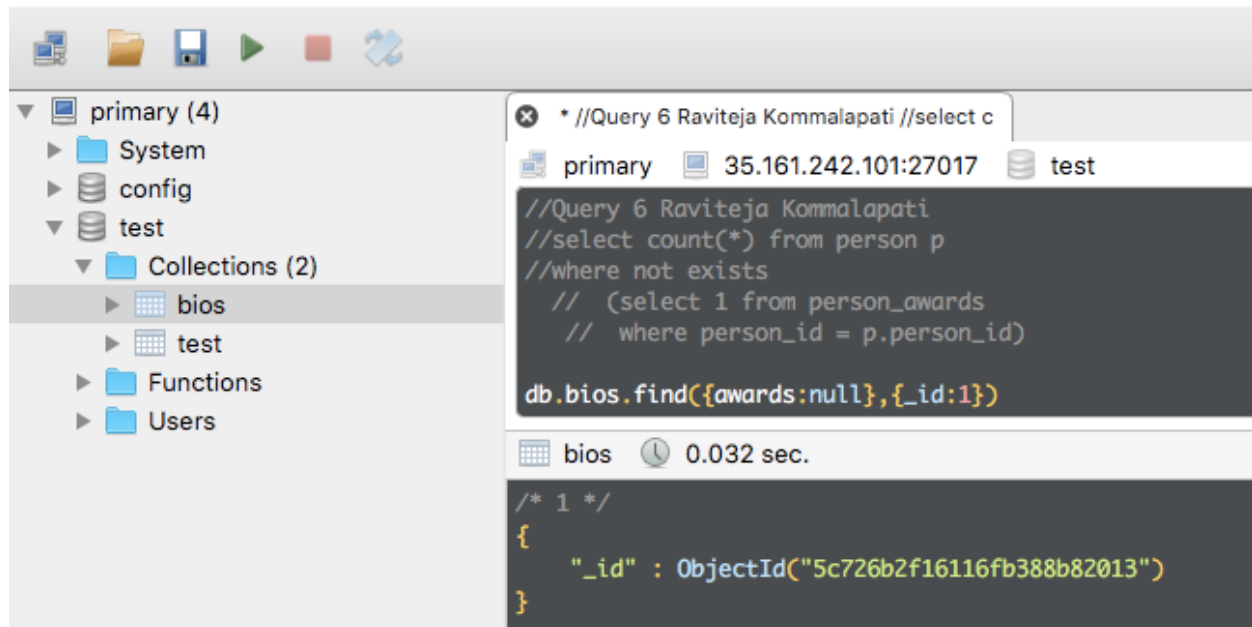
//Query 6 Raviteja Kommalapati
//select count(*) from person p
//where not exists
// (select 1 from person_awards
//  where person_id = p.person_id)

db.bios.count({awards:null}, {_id:false})

0.036 sec.

1
```

Query 7 Result



The screenshot shows the MongoDB Compass interface. On the left, the database structure is displayed: primary (4) > System > config > test > Collections (2) > bios. The main panel shows the query editor for Query 6. The query is a MongoDB find operation on the bios collection, filtering for documents where awards is null and _id is 1. The execution time is 0.032 seconds. The result is a single document with _id: ObjectId("5c726b2f16116fb388b82013").

```
primary 35.161.242.101:27017 test

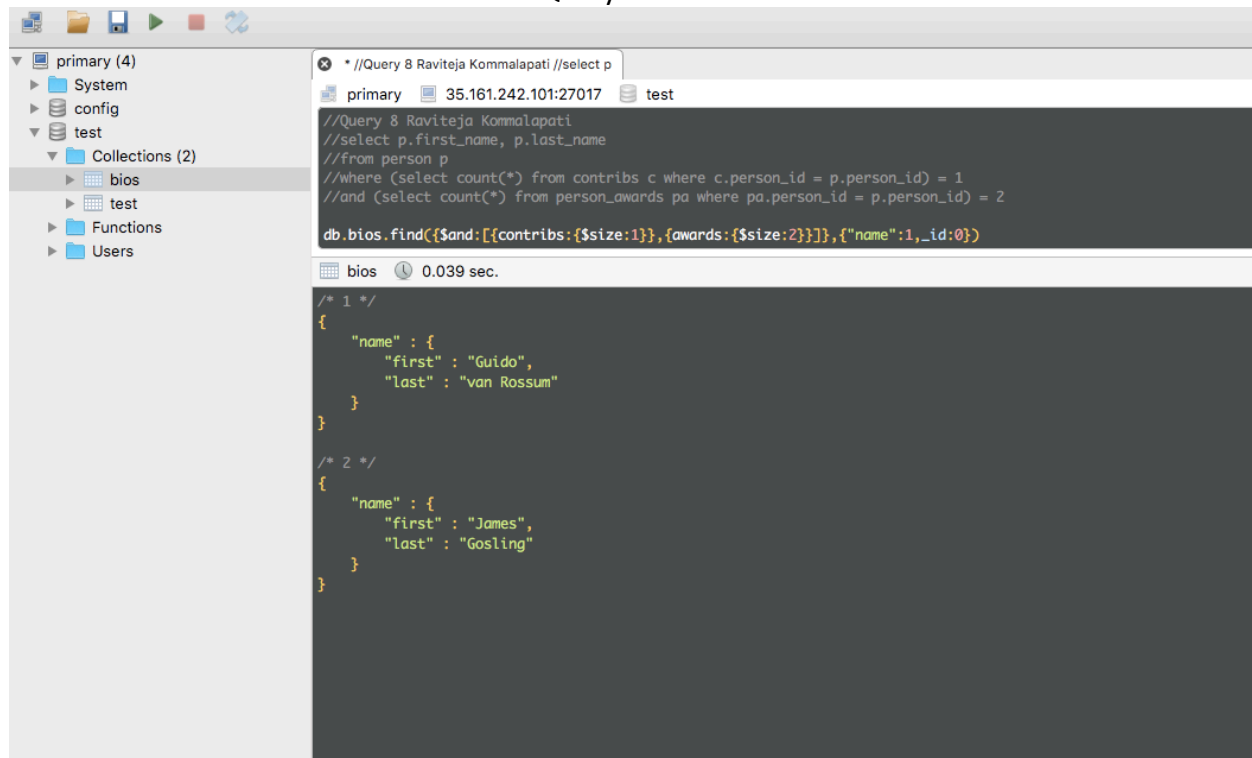
//Query 6 Raviteja Kommalapati
//select count(*) from person p
//where not exists
//  (select 1 from person_awards
//   where person_id = p.person_id)

db.bios.find({awards:null},{_id:1})

bios 0.032 sec.

/* 1 */
{
  "_id" : ObjectId("5c726b2f16116fb388b82013")
}
```

Query 8 Result



The screenshot shows the MongoDB Compass interface. On the left, the database structure is displayed: primary (4) > System > config > test > Collections (2) > bios. The main panel shows the query editor for Query 8. The query is a MongoDB find operation on the bios collection, filtering for documents where the name is Guido van Rossum and the awards array has 2 elements. The execution time is 0.039 seconds. The result is a single document with name: {first: "Guido", last: "van Rossum"}.

```
primary 35.161.242.101:27017 test

//Query 8 Raviteja Kommalapati
//select p.first_name, p.last_name
//from person p
//where (select count(*) from contribs c where c.person_id = p.person_id) = 1
//and (select count(*) from person_awards pa where pa.person_id = p.person_id) = 2

db.bios.find({$and:[{contribs:{size:1}},{awards:{size:2}}]},{name:1,_id:0})

bios 0.039 sec.

/* 1 */
{
  "name" : {
    "first" : "Guido",
    "last" : "van Rossum"
  }
}

/* 2 */
{
  "name" : {
    "first" : "James",
    "last" : "Gosling"
  }
}
```

Query 9 Result

The screenshot shows the MongoDB Compass interface. On the left, the database structure is visible: 'primary (4)' containing 'System', 'config', 'test', and 'Collections (2)' which includes 'bios', 'test', 'Functions', and 'Users'. The 'bios' collection is selected. The main pane displays the query and its results. The query is a MongoDB find operation with a complex filter. The results show four documents, each representing a person's bio with their name, first name, last name, and awards.

```
//Query 9 Raviteja Kommalapati //select p
primary 35.161.242.101:27017 test
//Query 9 Raviteja Kommalapati
//select p.first_name, p.last_name
//from person p
//where (select count(*) from contribs c where c.person_id = p.person_id) = 1
//or (select count(*) from person_awards pa where pa.person_id = p.person_id) = 2
db.bios.find({$or:[{contribs:{size:1}},{awards:{size:2}}]},{"name":1,_id:0})
```

bios 0.036 sec.

```
/* 1 */
{
  "name" : {
    "first" : "Guido",
    "last" : "van Rossum"
  }
}
/* 2 */
{
  "name" : {
    "first" : "Yukihiro",
    "aka" : "Matz",
    "last" : "Matsumoto"
  }
}
/* 3 */
{
  "name" : {
    "first" : "James",
    "last" : "Gosling"
  }
}
/* 4 */
{
  "name" : {
    "first" : "Martin",
    "last" : "Odersky"
  }
}
```

Query 10 Result

The screenshot shows the MongoDB Compass interface. On the left, the database structure is visible: 'primary (4)' containing 'System', 'config', 'test', and 'Collections (2)' which includes 'bios', 'test', 'Functions', and 'Users'. The 'bios' collection is selected. The main pane displays the query and its results. The query is a MongoDB find operation with a complex filter. The results show two documents, each representing a person's bio with their name, first name, last name, and awards.

```
//Query 8 Raviteja Kommalapati //select p
primary 35.161.242.101:27017 test
//Query 8 Raviteja Kommalapati
//select p.first_name, p.last_name, a.award_name
//from awards a, person_awards pa, person p
//where a.award_id = pa.award_id
//and p.person_id = pa.person_id
//and p.person_id = 1
db.bios.find({awards:{ne:null}},{"name":1,"awards.award":1,_id:0})
```

bios 0.043 sec.

```
/* 1 */
{
  "name" : {
    "first" : "John",
    "last" : "Backus"
  },
  "awards" : [
    {
      "award" : "W.W. McDowell Award"
    },
    {
      "award" : "National Medal of Science"
    },
    {
      "award" : "Turing Award"
    },
    {
      "award" : "Draper Prize"
    }
  ]
}
/* 2 */
{
  "name" : {
    "first" : "John",
    "last" : "McCarthy"
  },
  "awards" : [
    {
      "award" : "Turing Award"
    },
    {
      "award" : "Kyoto Prize"
    },
    {
      "award" : "National Medal of Science"
    }
  ]
}
```