

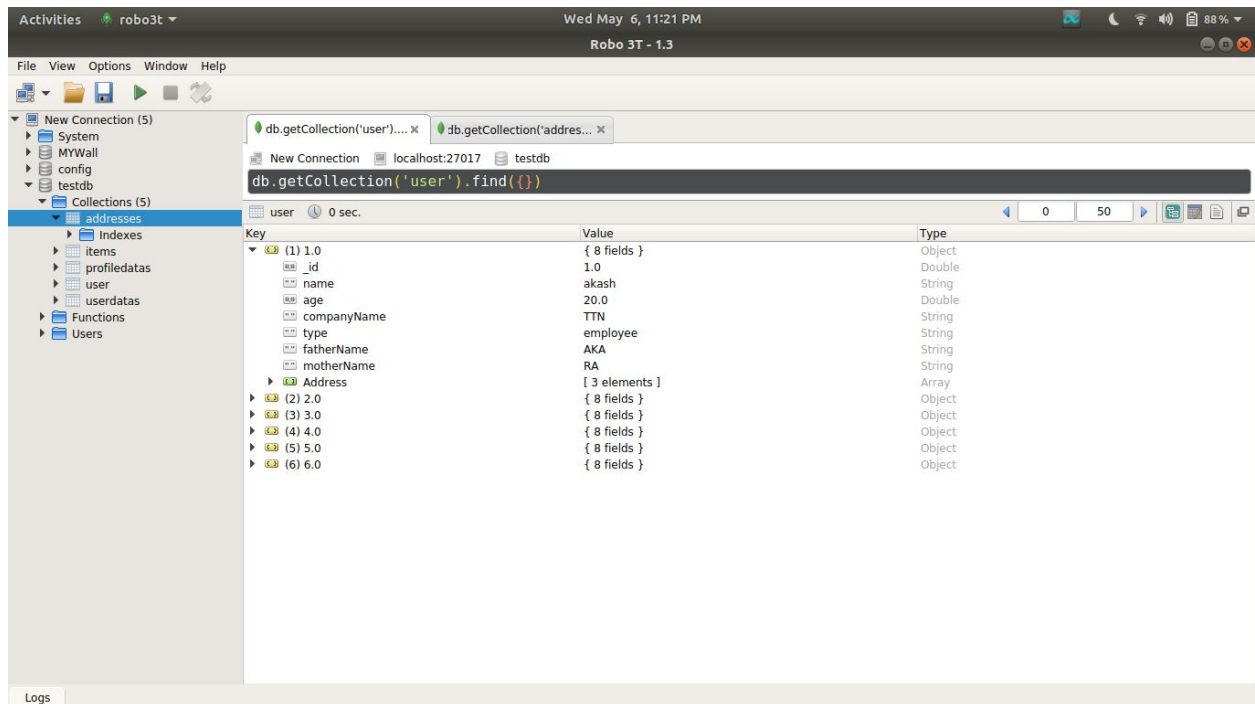
Exercise Answers

Add 50 User and 100 Address Collection, One User has multiple Address.

User properties - name, age, companyName, type, fatherName, motherName, Address ArrayIds as references.

Address Properties - houseNo, landmark, city, state, pincode.

User Data:



The screenshot shows the Robo 3T interface with a MongoDB connection to 'localhost:27017' and database 'testdb'. The 'user' collection is selected, and the query `db.getCollection('user').find({})` is executed. The results show 6 documents in the 'user' collection. The first document is expanded, showing its structure with fields like `_id`, `name`, `age`, `companyName`, `type`, `fatherName`, `motherName`, and an `Address` array containing 3 elements. The interface includes a sidebar with a tree view of collections and indexes, a top menu bar, and a status bar at the bottom.

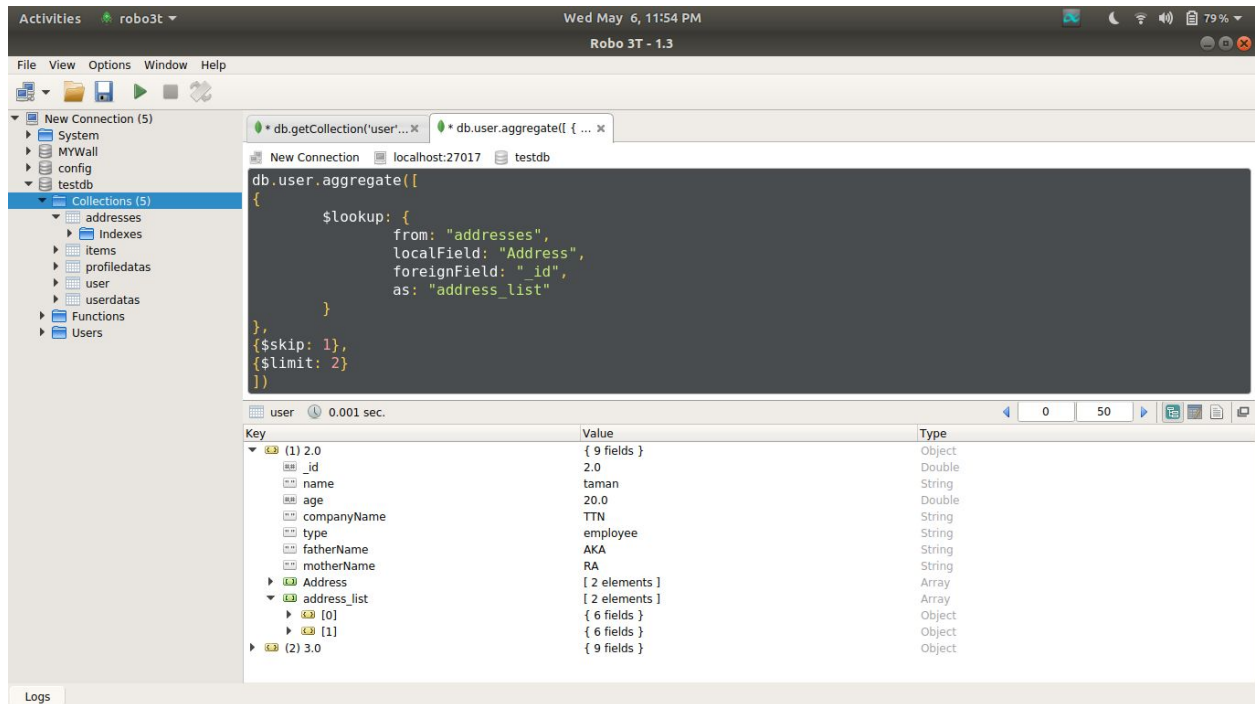
Key	Value	Type
(1) 1.0	{ 8 fields }	Object
_id	1.0	Double
name	akash	String
age	20.0	Double
companyName	TTN	String
type	employee	String
fatherName	AKA	String
motherName	RA	String
Address	[3 elements]	Array
(2) 2.0	{ 8 fields }	Object
(3) 3.0	{ 8 fields }	Object
(4) 4.0	{ 8 fields }	Object
(5) 5.0	{ 8 fields }	Object
(6) 6.0	{ 8 fields }	Object

Address Data:

The screenshot shows the Robo 3T interface with a MongoDB connection to localhost:27017, testdb. The 'addresses' collection is selected, and the query `db.getCollection('addresses').find({})` is executed. The results are displayed in a table with columns for Key, Value, and Type.

Key	Value	Type
(1) 1.0	{ 6 fields }	Object
id	1.0	Double
houseNo	1.0	Double
landmark	abcd	String
city	new delhi	String
state	delhi	String
pincode	110049	String
(2) 2.0	{ 6 fields }	Object
(3) 3.0	{ 6 fields }	Object

Write a query to fetch all address data from addresses in users. Limit should be 10 and skipping first five.



Start MongoDB without authentication

Create Database ttn and insert some data into user collection.

Setup the replication with Artiber and verify that ttn database is replicated into secondary.

Add 2 more members to a replica set with

`rs.add();`

Try adding a member which already has data.

Creating 3 server instances running on different ports

```
Activities Terminal Thu May 7, 12:27 AM 70%
Terminal
File Edit View Search Terminal Help
akash@akash:replicaset-mongodb $ ls
rs1 rs2 rs3
akash@akash:replicaset-mongodb $ mongod --replSet akash --logpath "1.log" --dbpath rs1 --port 27017 &
[1] 5458
akash@akash:replicaset-mongodb $ mongod --replSet akash --logpath "2.log" --dbpath rs2 --port 27018 &
[2] 5501
akash@akash:replicaset-mongodb $ mongod --replSet akash --logpath "3.log" --dbpath rs3 --port 27019 &
[3] 5545
akash@akash:replicaset-mongodb $ ps -ef | grep mongo
akash 5458 4984 3 00:23 pts/0 00:00:00 mongod --replSet akash --logpath 1.log --dbpath rs1 --port 27017
akash 5501 4984 4 00:23 pts/0 00:00:00 mongod --replSet akash --logpath 2.log --dbpath rs2 --port 27018
akash 5545 4984 9 00:23 pts/0 00:00:00 mongod --replSet akash --logpath 3.log --dbpath rs3 --port 27019
akash 5587 4984 0 00:23 pts/0 00:00:00 grep --color=auto mongo
akash@akash:replicaset-mongodb $ mongo
MongoDB shell version v4.2.6
```

Adding config to initiate replication

```
Activities Terminal Thu May 7, 12:27 AM
File Edit View Search Terminal Help

> config = {
... _id: 'akash',
... members: [
...   { _id: 0, host: 'localhost:27017' },
...   { _id: 1, host: 'localhost:27018' },
...   { _id: 2, host: 'localhost:27019' },
... ]
... }
{
  "_id": "akash",
  "members": [
    {
      "_id": 0,
      "host": "localhost:27017"
    },
    {
      "_id": 1,
      "host": "localhost:27018"
    },
    {
      "_id": 2,
      "host": "localhost:27019"
    }
  ]
}
rs.initiate(config);
{
  "ok": 1,
  "$clusterTime": {
    "clusterTime": Timestamp(1588791401, 1),
    "signature": {
      "hash": BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId": NumberLong(0)
    }
  },
  "operationTime": Timestamp(1588791401, 1)
}
```

```
Activities Terminal Thu May 7, 12:27 AM
Terminal
File Edit View Search Terminal Help
}
akash:SECONDARY> rs.status()
{
  "set" : "akash",
  "date" : ISODate("2020-05-06T18:56:58.630Z"),
  "myState" : 1,
  "term" : NumberLong(1),
  "syncingTo" : "",
  "syncSourceHost" : "",
  "syncSourceId" : -1,
  "heartbeatIntervalMillis" : NumberLong(2000),
  "majorityVoteCount" : 2,
  "writeMajorityCount" : 2,
```

```
Activities Terminal Thu May 7, 12:28 AM
Terminal
File Edit View Search Terminal Help
}
  "members" : [
    {
      "_id" : 0,
      "name" : "localhost:27017",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 237,
      "optime" : {
        "ts" : Timestamp(1588791414, 1),
        "t" : NumberLong(1)
      },
      "optimeDate" : ISODate("2020-05-06T18:56:54Z"),
      "syncingTo" : "",
      "syncSourceHost" : "",
      "syncSourceId" : -1,
      "infoMessage" : "could not find member to sync from",
      "electionTime" : Timestamp(1588791412, 1),
      "electionDate" : ISODate("2020-05-06T18:56:52Z"),
      "configVersion" : 1,
      "self" : true,
      "lastHeartbeatMessage" : ""
    },
    {
      "_id" : 1,
      "name" : "localhost:27018",
      "health" : 1,
      "state" : 2,
      "stateStr" : "SECONDARY",
      "uptime" : 17,
      "optime" : {
        "ts" : Timestamp(1588791414, 1),
        "t" : NumberLong(1)
      },
      "optimeDurable" : {
        "ts" : Timestamp(1588791414, 1),
        "t" : NumberLong(1)
      }
    }
  ],
}
```

Adding data to primary server named ttn

The screenshot shows the Robo 3T interface with the following components:

- Top Bar:** Displays 'Activities', 'robo3t', the date 'Thu May 7, 12:35 AM', and the version 'Robo 3T - 1.3'.
- Left Panel (Database Explorer):** A tree view showing the database structure. The 'primary' database is selected, and the 'user' collection is highlighted under the 'Collections (1)' folder.
- Command Editor:** Contains the MongoDB query `db.getCollection('user').find({})`.
- Results Panel:** Shows the execution of the query on the 'primary' database at 'localhost:27017'. The results are displayed in a table with columns 'Key', 'Value', and 'Type'.

Key	Value	Type
(1) ObjectId("5eb309ad4879bb82b821353a")	{ 4 fields }	Object
- Bottom Panel:** A 'Logs' tab is visible at the bottom left.

Checking for synchronization of data in both secondary servers

The screenshot shows the Robo 3T interface with the 'secondary2 (4)' server selected in the left sidebar. The 'user' collection is expanded, and the 'Indexes' sub-item is selected. The main pane displays the command `db.getCollection('user').find({})` and the execution time `0.001 sec.`. The results table shows one document:

Key	Value	Type
(1) ObjectId("5eb309ad4879bb82b821353a")	{ 4 fields }	Object

The screenshot shows the Robo 3T interface with the 'secondary2 (4)' server selected in the left sidebar. The 'user' collection is expanded, and the 'Indexes' sub-item is selected. The main pane displays the command `db.getCollection('user').find({})` and the execution time `0.001 sec.`. The results table shows one document:

Key	Value	Type
(1) ObjectId("5eb309ad4879bb82b821353a")	{ 4 fields }	Object

Creating 4th instance

```
File Edit View Search Terminal Help
akash@akash:replicaset-mongodb $ mkdir rs4
akash@akash:replicaset-mongodb $ mongod --replSet akash --logpath "4.log" --dbpath rs4 --port 27020 &
[1] 6300
akash@akash:replicaset-mongodb $
```

Adding the fourth instance as a secondary

```
akash:PRIMARY> rs.add('localhost:27020');
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1588792181, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  },
  "operationTime" : Timestamp(1588792181, 1)
}
```

Checking status

```
Activities Terminal Thu May 7, 12:41 AM
File Edit View Search Terminal Help

rs.add('localhost:27020');
{
  "ok" : 1,
  "$clusterTime" : {
    "clusterTime" : Timestamp(1588792181, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  },
  "operationTime" : Timestamp(1588792181, 1)
}

rs.status()
{
  "set" : "akash",
  "date" : ISODate("2020-05-06T19:10:57.543Z"),
  "members" : [
    {
      "_id" : 0,
      "name" : "localhost:27017",
      "health" : 1,
      "state" : 2,
      "stateStr" : "SECONDARY",
      "uptime" : 77,
      "optime" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDurable" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
      "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
      "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
      "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
      "pingMs" : NumberLong(0),
      "lastHeartbeatMessage" : "",
      "syncingTo" : "localhost:27019",
      "syncSourceHost" : "localhost:27019",
      "syncSourceId" : 2,
      "infoMessage" : "",
      "configVersion" : 2
    },
    {
      "_id" : 1,
      "name" : "localhost:27017",
      "health" : 1,
      "state" : 2,
      "stateStr" : "SECONDARY",
      "uptime" : 77,
      "optime" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDurable" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
      "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
      "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
      "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
      "pingMs" : NumberLong(0),
      "lastHeartbeatMessage" : "",
      "syncingTo" : "localhost:27019",
      "syncSourceHost" : "localhost:27019",
      "syncSourceId" : 2,
      "infoMessage" : "",
      "configVersion" : 2
    },
    {
      "_id" : 2,
      "name" : "localhost:27019",
      "health" : 1,
      "state" : 1,
      "stateStr" : "PRIMARY",
      "uptime" : 77,
      "optime" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDurable" : {
        "ts" : Timestamp(1588792252, 1),
        "t" : NumberLong(1)
      },
      "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
      "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
      "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
      "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
      "pingMs" : NumberLong(0),
      "lastHeartbeatMessage" : "",
      "syncingTo" : null,
      "syncSourceHost" : null,
      "syncSourceId" : null,
      "infoMessage" : "",
      "configVersion" : 2
    }
  ],
  "primary" : 2,
  "secondary" : 2,
  "secondaryPreferred" : 0,
  "available" : 2,
  "replSetGetStatus" : {
    "set" : "akash",
    "date" : ISODate("2020-05-06T19:10:57.543Z"),
    "members" : [
      {
        "_id" : 0,
        "name" : "localhost:27017",
        "health" : 1,
        "state" : 2,
        "stateStr" : "SECONDARY",
        "uptime" : 77,
        "optime" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDurable" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
        "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
        "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
        "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
        "pingMs" : NumberLong(0),
        "lastHeartbeatMessage" : "",
        "syncingTo" : "localhost:27019",
        "syncSourceHost" : "localhost:27019",
        "syncSourceId" : 2,
        "infoMessage" : "",
        "configVersion" : 2
      },
      {
        "_id" : 1,
        "name" : "localhost:27017",
        "health" : 1,
        "state" : 2,
        "stateStr" : "SECONDARY",
        "uptime" : 77,
        "optime" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDurable" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
        "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
        "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
        "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
        "pingMs" : NumberLong(0),
        "lastHeartbeatMessage" : "",
        "syncingTo" : "localhost:27019",
        "syncSourceHost" : "localhost:27019",
        "syncSourceId" : 2,
        "infoMessage" : "",
        "configVersion" : 2
      },
      {
        "_id" : 2,
        "name" : "localhost:27019",
        "health" : 1,
        "state" : 1,
        "stateStr" : "PRIMARY",
        "uptime" : 77,
        "optime" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDurable" : {
          "ts" : Timestamp(1588792252, 1),
          "t" : NumberLong(1)
        },
        "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
        "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
        "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
        "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
        "pingMs" : NumberLong(0),
        "lastHeartbeatMessage" : "",
        "syncingTo" : null,
        "syncSourceHost" : null,
        "syncSourceId" : null,
        "infoMessage" : "",
        "configVersion" : 2
      }
    ],
    "primary" : 2,
    "secondary" : 2,
    "secondaryPreferred" : 0,
    "available" : 2,
    "replSetGetStatus" : {
      "set" : "akash",
      "date" : ISODate("2020-05-06T19:10:57.543Z"),
      "members" : [
        {
          "_id" : 0,
          "name" : "localhost:27017",
          "health" : 1,
          "state" : 2,
          "stateStr" : "SECONDARY",
          "uptime" : 77,
          "optime" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDurable" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
          "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
          "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
          "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
          "pingMs" : NumberLong(0),
          "lastHeartbeatMessage" : "",
          "syncingTo" : "localhost:27019",
          "syncSourceHost" : "localhost:27019",
          "syncSourceId" : 2,
          "infoMessage" : "",
          "configVersion" : 2
        },
        {
          "_id" : 1,
          "name" : "localhost:27017",
          "health" : 1,
          "state" : 2,
          "stateStr" : "SECONDARY",
          "uptime" : 77,
          "optime" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDurable" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
          "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
          "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
          "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
          "pingMs" : NumberLong(0),
          "lastHeartbeatMessage" : "",
          "syncingTo" : "localhost:27019",
          "syncSourceHost" : "localhost:27019",
          "syncSourceId" : 2,
          "infoMessage" : "",
          "configVersion" : 2
        },
        {
          "_id" : 2,
          "name" : "localhost:27019",
          "health" : 1,
          "state" : 1,
          "stateStr" : "PRIMARY",
          "uptime" : 77,
          "optime" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDurable" : {
            "ts" : Timestamp(1588792252, 1),
            "t" : NumberLong(1)
          },
          "optimeDate" : ISODate("2020-05-06T19:10:52Z"),
          "optimeDurableDate" : ISODate("2020-05-06T19:10:52Z"),
          "lastHeartbeat" : ISODate("2020-05-06T19:10:57.543Z"),
          "lastHeartbeatRecv" : ISODate("2020-05-06T19:10:57.580Z"),
          "pingMs" : NumberLong(0),
          "lastHeartbeatMessage" : "",
          "syncingTo" : null,
          "syncSourceHost" : null,
          "syncSourceId" : null,
          "infoMessage" : "",
          "configVersion" : 2
        }
      ],
      "primary" : 2,
      "secondary" : 2,
      "secondaryPreferred" : 0,
      "available" : 2
    }
  }
}
```

Creating 5th instance

```
[1] 6560
akash@akash:replicaset-mongodb $ mkdir rs5
akash@akash:replicaset-mongodb $ mongod --replSet akash --logpath "5.log" --dbpath rs5 --port 27021 &
[2] 6572
akash@akash:replicaset-mongodb $
```