How to setup MongoDB - Four sharded replicas in Windows 10 on Single machine/Laptop

------------ USE Following steps sequentially in windows cmd prompt.

1) setup config servers

Open windows cmd prompt

```
cd \
mkdir data\config1
mkdir data\config2
mkdir data\config3
mkdir data\logs
```

Open multiple windows cmd prompt’s as needed to run each of these individually

```
mongod --configsvr --dbpath data/config1 --port 2011 --replSet rs0  --bind_ip localhost
mongod --configsvr --dbpath data/config2 --port 2012 --replSet rs0  --bind_ip localhost
mongod --configsvr --dbpath data/config3 --port 2013 --replSet rs0  --bind_ip localhost
```

---connect to config1 server

Open windows cmd prompt

```
mongo --port 2011

rs.initiate(
  {
    _id: "rs0",
    configsvr: true,
    members: [
      { _id : 0, host : "localhost:2011" },
      { _id : 1, host : "localhost:2012" },
      { _id : 2, host : "localhost:2013" }
    ]
  }
)
```

2) setup shard1

Open windows cmd prompt

```
cd \
mkdir data\shard1\rs2
mkdir data\shard1\rs2
mkdir data\shard1\rs3
```

Open multiple windows cmd prompt’s as needed to run each of these individually cmd prompt

```
mongod --shardsvr --replSet shard1rs --port 20010 --dbpath data/shard1/rs1   --bind_ip localhost
mongod --shardsvr --replSet shard1rs --port 20011 --dbpath data/shard1/rs2    --bind_ip localhost
mongod --shardsvr --replSet shard1rs --port 20012 --dbpath data/shard1/rs3   --bind_ip localhost
```

---connect to shard1 host

Open windows cmd prompt

```
mongo --port 20010

rs.initiate(
  {
    _id: "shard1rs",
    members: [
      { _id : 0, host : "localhost:20010" },
      { _id : 1, host : "localhost:20011" },
      { _id : 2, host : "localhost:20012" }
    ]
  }
)
```

3) setup shard2

Open windows cmd prompt

```
cd \
mkdir data\shard2\rs2
mkdir data\shard2\rs2
mkdir data\shard2\rs3
```

Open multiple windows cmd prompt’s as needed to run each of these individually cmd prompt

```
mongod --shardsvr --replSet shard2rs --port 20013 --dbpath data/shard2/rs1
mongod --shardsvr --replSet shard2rs --port 20014 --dbpath data/shard2/rs2
mongod --shardsvr --replSet shard2rs --port 20015 --dbpath data/shard2/rs3
```

Open windows cmd prompt

```
mongo --port 20013
rs.initiate(
 {
   _id: "shard2rs",
   members: [
     { _id : 0, host : "localhost:20013" },
     { _id : 1, host : "localhost:20014" },
     { _id : 2, host : "localhost:20015" }
   ]
 }
)
```

4) setup shard3

Open windows cmd prompt

```
cd \
mkdir data\shard3\rs2
mkdir data\shard3\rs2
mkdir data\shard3\rs3
```

Open multiple windows cmd prompt’s as needed to run each of these individually cmd prompt

```
mongod --shardsvr --replSet shard3rs --port 20016--dbpath data/shard3/rs1   --bind_ip
localhost
mongod --shardsvr --replSet shard3rs --port 20017 --dbpath data/shard3/rs2     --bind_ip
localhost
mongod --shardsvr --replSet shard3rs --port 20018 --dbpath data/shard3/rs3  --bind_ip
localhost
```

Open windows cmd prompt

```
mongo --port 20016
rs.initiate(
 {
   _id: "shard3rs",
   members: [
     { _id : 0, host : "localhost:20016" },
     { _id : 1, host : "localhost:20017" },
     { _id : 2, host : "localhost:20018" }
   ]
 }
)
```

5) setup shard4

Open windows cmd prompt

```
cd \
mkdir data\shard4\rs2
mkdir data\shard4\rs2
mkdir data\shard4\rs3
```

Open multiple windows cmd prompt’s as needed to run each of these individually cmd prompt

```
mongod --shardsvr --replSet shard4rs --port 20019 --dbpath data/shard4/rs1   --bind_ip
localhost
mongod --shardsvr --replSet shard4rs --port 20020 --dbpath data/shard4/rs2    --bind_ip
localhost
mongod --shardsvr --replSet shard4rs --port 20021  --dbpath data/shard4/rs3  --bind_ip
localhost
```

Open windows cmd prompt

```
mongo --port 20019
rs.initiate(
  {
    _id: "shard4rs",
    members: [
      { _id : 0, host : "localhost:20019" },
      { _id : 1, host : "localhost:20020" },
      { _id : 2, host : "localhost:20021" }
    ]
  }
)
```

6) setup mongos and add shards

Open windows cmd prompt

```
cd \
mongos --configdb "rs0/localhost:2011,localhost:2012,localhost:2013" --logpath
data/logs/log.mongos0 --port 27200
```

--connect to mongos and add shards:

Open windows cmd prompt

```
mongo --port 27200
```

```
mongos>  use config
mongos> sh.addShard("shard1rs/localhost:20010")
mongos> sh.addShard("shard2rs/localhost:20013")
mongos> sh.addShard("shard3rs/localhost:20016")
mongos> sh.addShard("shard4rs/localhost:20019")
```

```
mongos> sh.status()
```

```
db.shards.find()
```

7)enable sharding on mongoMart and at collection level

```
use mongoMart
sh.enableSharding("mongoMart")
```

---enable sharding on collection level

```
sh.shardCollection("mongoMart.shop",{"_id":"hashed"})
```

```
for (var i =1;i<=100;i++) db.shop.insert({x:i})
```

```
mongos> db.shop.find().count()
100
```

verfiy distribution and enjoy:

```
mongos> db.shop.getShardDistribution()
```

connect to each shards primary and verify counts

Create additional collections:

```
sh.shardCollection("mongoMart.restaurants",{"_id":"hashed"})
```

```
mongoimport --port 27200   --db mongoMart  --collection restaurants  --file  restaurants.json
```

```
2021-05-24T11:32:59.200-0400    connected to: mongodb://localhost:27200/
2021-05-24T11:33:02.200-0400    [#.......................] mongoMart.restaurants       7.56MB/144MB
(5.3%)
2021-05-24T11:33:05.200-0400    [##......................] mongoMart.restaurants
13.8MB/144MB (9.6%)
2021-05-24T11:33:08.200-0400    [##......................] mongoMart.restaurants
16.1MB/144MB (11.2%)
```

```
2021-05-24T11:33:11.200-0400    [###.....................] mongoMart.restaurants
21.9MB/144MB (15.2%)
2021-05-24T11:33:14.201-0400    [####....................] mongoMart.restaurants
26.9MB/144MB (18.7%)
2021-05-24T11:33:17.200-0400    [#####...................] mongoMart.restaurants
34.7MB/144MB (24.1%)
2021-05-24T11:33:20.201-0400    [#######.................] mongoMart.restaurants
42.6MB/144MB (29.7%)
2021-05-24T11:33:23.201-0400    [########................] mongoMart.restaurants
50.0MB/144MB (34.8%)
2021-05-24T11:33:26.200-0400    [#########...............] mongoMart.restaurants
55.0MB/144MB (38.3%)
2021-05-24T11:33:29.200-0400    [##########..............] mongoMart.restaurants
61.5MB/144MB (42.8%)
2021-05-24T11:33:32.201-0400    [###########.............] mongoMart.restaurants
67.0MB/144MB (46.6%)
2021-05-24T11:33:35.200-0400    [############............] mongoMart.restaurants
74.4MB/144MB (51.8%)
2021-05-24T11:33:38.201-0400    [#############...........] mongoMart.restaurants
82.3MB/144MB (57.3%)
2021-05-24T11:33:41.202-0400    [###############.........] mongoMart.restaurants
90.3MB/144MB (62.8%)
2021-05-24T11:33:44.201-0400    [################........] mongoMart.restaurants
98.2MB/144MB (68.3%)
2021-05-24T11:33:47.200-0400    [#################.......] mongoMart.restaurants
104MB/144MB (72.6%)
2021-05-24T11:33:50.201-0400    [##################......] mongoMart.restaurants
110MB/144MB (76.4%)
2021-05-24T11:33:53.200-0400    [###################.....] mongoMart.restaurants
115MB/144MB (79.8%)
2021-05-24T11:33:56.200-0400    [####################....] mongoMart.restaurants
120MB/144MB (83.4%)
2021-05-24T11:33:59.201-0400    [#####################...] mongoMart.restaurants
126MB/144MB (87.8%)
2021-05-24T11:34:02.200-0400    [######################..] mongoMart.restaurants
134MB/144MB (93.3%)
2021-05-24T11:34:05.200-0400    [#######################.] mongoMart.restaurants
140MB/144MB (97.7%)
2021-05-24T11:34:07.325-0400    [########################] mongoMart.restaurants
144MB/144MB (100.0%)
2021-05-24T11:34:07.325-0400    1000000 document(s) imported successfully. 0 document(s)
failed to import
```

```
mongos> db.restaurants.getShardDistribution()
```

```
mongoimport --port 27200   --db mongoMart  --collection restaurants2  --file restaurants2.json
```

```
mongos> db.restaurants2.find({cuisine: "Bakery", name: /^Morris/})
{ "_id" : ObjectId("60ac5dd87f9558d81958acb7"), "address" : { "building" : "1007", "coord" : [ -73.856077, 40.848447 ], "street" : "Morris Park Ave", "zipcode" : "10462" }, "borough" : "Bronx", "cuisine" : "Bakery", "grades" : [ { "date" : ISODate("2014-03-03T00:00:00Z"), "grade" : "A", "score" : 2 }, { "date" : ISODate("2013-09-11T00:00:00Z"), "grade" : "A", "score" : 6 }, { "date" : ISODate("2013-01-24T00:00:00Z"), "grade" : "A", "score" : 10 }, { "date" : ISODate("2011-11-23T00:00:00Z"), "grade" : "A", "score" : 9 }, { "date" : ISODate("2011-03-10T00:00:00Z"), "grade" : "B", "score" : 14 } ], "name" : "Morris Park Bake Shop", "restaurant_id" : "30075445" }
```

Enabling Authentication of whole cluster

1) Create keyfile :

   ```
   openssl rand -base64 768  >  data/keys/shard_cluster_keyfile
   ```

2) Use start_mongo_2 Auth.bat file to start shard cluster and connect to mongos

   ```
   C:\>mongo --port 27200
   MongoDB shell version v4.4.6
   connecting to:
   mongodb://127.0.0.1:27200/?compressors=disabled&gssapiServiceName=mongodb
   Implicit session: session { "id" : UUID("3207df31-ac62-45bd-9fce-49b156dd2d7e") }
   MongoDB server version: 4.4.6
   mongos> show dbs
   mongos>
   ```

3) Create account for user administration :

```
      mongos> use admin
mongos> db.createUser( {user: "user1",   pwd: "secret_password", roles: [ { role: “userAdminAnyDatabase", db: "admin" }, "readWriteAnyDatabase" ]
...   }
... )
Successfully added user: {
        "user" : "user1",
```

```
    "roles" : [
        {
            "role" : "userAdminAnyDatabase",
            "db" : "admin"
        },
        "readWriteAnyDatabase"

mongos>db.auth({user:’user1’,pwd:’secret_password’})
1
```