Start MongoDb server


```
C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe                                    —  □
2018-02-20T19:28:31.635-0800 I CONTROL  [initandlisten] **           Read and write access to data and configuration
nrestricted.
2018-02-20T19:28:31.635-0800 I CONTROL  [initandlisten]
2018-02-20T19:28:31.635-0800 I CONTROL  [initandlisten] ** WARNING: This server is bound to localhost.
2018-02-20T19:28:31.636-0800 I CONTROL  [initandlisten] **           Remote systems will be unable to connect to this
ver.
2018-02-20T19:28:31.636-0800 I CONTROL  [initandlisten] **           Start the server with --bind_ip <address> to spe
 which IP
2018-02-20T19:28:31.637-0800 I CONTROL  [initandlisten] **           addresses it should serve responses from, or wit
bind_ip_all to
2018-02-20T19:28:31.637-0800 I CONTROL  [initandlisten] **           bind to all interfaces. If this behavior is desi
 start the
2018-02-20T19:28:31.638-0800 I CONTROL  [initandlisten] **           server with --bind_ip 127.0.0.1 to disable this
ing.
2018-02-20T19:28:31.638-0800 I CONTROL  [initandlisten]
2018-02-20T19:28:31.638-0800 I CONTROL  [initandlisten]
2018-02-20T19:28:31.639-0800 I CONTROL  [initandlisten] ** WARNING: The file system cache of this machine is configu
to be greater than 40% of the total memory. This can lead to increased memory pressure and poor performance.
2018-02-20T19:28:31.640-0800 I CONTROL  [initandlisten] See http://dochub.mongodb.org/core/wt-windows-system-file-ca
2018-02-20T19:28:31.641-0800 I CONTROL  [initandlisten]
2018-02-20T22:28:31.889-0500 I FTDC     [initandlisten] Initializing full-time diagnostic data capture with directory
:/data/db/diagnostic.data'
2018-02-20T22:28:31.892-0500 I NETWORK  [initandlisten] waiting for connections on port 27017
```

Starting mongo shell that is not connected to any mongod


```
Microsoft Windows [Version 10.0.16299.248]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\mereddydivyareddy>cd C:\Program Files\MongoDB\Server\3.6\bin

C:\Program Files\MongoDB\Server\3.6\bin>start Mongod

C:\Program Files\MongoDB\Server\3.6\bin>start Mongod

C:\Program Files\MongoDB\Server\3.6\bin> mongo --nodb
MongoDB shell version v3.6.2
>
```

Create replica set by

replicaSet = new ReplSetTest({"nodes" : 3})

```
> replicaSet = new ReplSetTest({"nodes" : 3})
```

Start the replication processes :

```
      nodes  : [ ]
}
>  replicaSet.startSet()
```

```
ReplSetTest Starting....
Resetting db path '/data/db/__unknown_name__-0'
2018-02-20T23:02:52.206-0500 E QUERY    [thread1] Error: Caught std::exception of type class boost::filesystem::filesys
em_error: boost::filesystem::remove: The process cannot access the file because it is being used by another process: "/
ata/db/__unknown_name__-0\collection-0--2353415141256918807.wt" :
MongoRunner.runMongod@src/mongo/shell/servers.js:721:17
ReplSetTest/this.start@src/mongo/shell/replsettest.js:1784:20
ReplSetTest/this.startSet@src/mongo/shell/replsettest.js:500:24
@(shell):1:1
>
```

replicaSet.initiate()

```
"use strict";

        var master;
        assert.soonNoExcept(function() {
            return (master = self.getPrimary());
        }, "waiting for master", timeout);

        return master;
    },
        "name" : "__unknown_name__",
        "useHostName" : true,
        "host" : "DESKTOP-7F7GG1M",
        "oplogSize" : 40,
        "useSeedList" : false,
        "keyFile" : undefined,
        "protocolVersion" : undefined,
        "waitForKeys" : undefined,
        "nodeOptions" : {
                "n0" : undefined,
                "n1" : undefined,
                "n2" : undefined
        },
        "ports" : [
                20000,
                20001,
                20002
        ],
        "nodes" : [ ]
}
```

port Numbers are :

20000

20001

20002

```
> conn1 = new Mongo("localhost:20000")
connection to localhost:20000
```

```
> primaryDB = conn1.getDB("test")
test
```

```
> primaryDB.isMaster()
```

```
        "hosts" : [
                "Yellanna-PC:20000",
                "Yellanna-PC:20001",
                "Yellanna-PC:20002"
        ],
        "setName" : "__unknown_name__",
        "setVersion" : 2,
        "ismaster" : true,
        "secondary" : false,
        "primary" : "Yellanna-PC:20000",
        "me" : "Yellanna-PC:20000",
        "electionId" : ObjectId("7fffffff0000000000000001"),
```

Insertion:

```
> for (i=0; i<1000; i++) { primaryDB.coll.insert({count: i}) }
WriteResult({ "nInserted" : 1 })
> primaryDB.coll.count()
1000
```

Connecting to DB-port 2:

```
> conn2 = new Mongo("localhost:20001")
connection to localhost:20001
```

Test:

```
> secondaryDB = conn2.getDB("test")
test
```

Find:

```
> secondaryDB.coll.find()
Error: error: {
        "operationTime" : Timestamp(1519179265, 254),
        "ok" : 0,
        "errmsg" : "not master and slaveOk=false",
        "code" : 13435,
        "codeName" : "NotMasterNoSlaveOk",
        "$clusterTime" : {
                "clusterTime" : Timestamp(1519179265, 254),
                "signature" : {
                        "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
```

```
> conn2.setSlaveOk()
```

```
> secondaryDB.coll.find()
{ "_id" : ObjectId("5a8cd6000d28dd6b39694651"), "count" : 1 }
{ "_id" : ObjectId("5a8cd6000d28dd6b39694659"), "count" : 9 }
{ "_id" : ObjectId("5a8cd6000d28dd6b39694661"), "count" : 17 }
{ "_id" : ObjectId("5a8cd6000d28dd6b39694665"), "count" : 21 }
{ "_id" : ObjectId("5a8cd6000d28dd6b39694672"), "count" : 34 }
```

```
> secondaryDB.coll.count()
1000
>
> secondaryDB.coll.insert(({"count" : 1001}))
WriteResult({ "writeError" : { "code" : 10107, "errmsg" : "not master" } })
>
>
> secondaryDB.runCommand(({"getLastError" : 1}))
{
        "connectionId" : 7,
        "err" : "not master",
        "code" : 10107,
        "codeName" : "NotMaster",
        "n" : 0,
        "ok" : 1,
        "operationTime" : Timestamp(1519179265, 254),
        "$clusterTime" : {
                "clusterTime" : Timestamp(1519179265, 254),
                "signature" : {
                        "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
                        "keyId" : NumberLong(0)
                }
        }
}
```

```
> primaryDB.printReplicationInfo()
configured oplog size:   40MB
log length start to end: 121secs (0.03hrs)
oplog first event time:  Tue Feb 20 2018 22
oplog last event time:   Tue Feb 20 2018 22
now:                     Tue Feb 20 2018 22
```

```
> primaryDB.printSlaveReplicationInfo()
source: Yellanna-PC:20001
        syncedTo: Tue Feb 20 2018 22:12:02 GMT-0500 (Eastern Standard Time)
        0 secs (0 hrs) behind the primary
source: Yellanna-PC:20002
        syncedTo: Tue Feb 20 2018 22:12:02 GMT-0500 (Eastern Standard Time)
        0 secs (0 hrs) behind the primary
```

primary Shutdown and secondary becomes master:

```
>  primaryDB.adminCommand({"shutdown" : 1})
2018-02-20T21:39:32.725-0500 E QUERY    [thread1] Error: error doing query: failed: network error while attempting to run command 'shu
DB.prototype.runCommand@src/mongo/shell/db.js:168:1
DB.prototype.adminCommand@src/mongo/shell/db.js:186:16
@(shell):1:1
>
>
> secondaryDB.isMaster()
{
        "hosts" : [
                "Yellanna-PC:20000",
                "Yellanna-PC:20001",
                "Yellanna-PC:20002"
        ],
        "setName" : "__unknown_name__",
        "setVersion" : 2,
        "ismaster" : false,
        "secondary" : true,
        "primary" : "Yellanna-PC:20002",
        "me" : "Yellanna-PC:20001",
        "lastWrite" : {
                "opTime" : {
                        "ts" : Timestamp(1519180787, 1),
                        "t" : NumberLong(2)
                },
                "lastWriteDate" : ISODate("2018-02-21T02:39:47Z"),
```

replication information:

```
> secondaryDB.printReplicationInfo()
configured oplog size:    40MB
log length start to end: 1923secs (0
oplog first event time:  Tue Feb 20
oplog last event time:   Tue Feb 20
now:                     Tue Feb 20
>
> secondaryDB.printSlaveReplicationI
source: Yellanna-PC:20000
        syncedTo: Wed Dec 31 1969 19
        1519180787 secs (421994.66 h
source: Yellanna-PC:20001
        syncedTo: Tue Feb 20 2018 21
        0 secs (0 hrs) behind the pr
```

Stopping:

```
fused it.
d20002| 2018-02-20T23:01:12.708-0500 I ASIO     [NetworkInterfaceASIO-RS-0] Dropping all pooled connections to Yellanna-PC:20001 d
d20002| 2018-02-20T23:01:12.708-0500 I REPL     [rsBackgroundSync] Replication producer stopped after oplog fetcher finished retur
ries and re-evaluating our sync source.
d20002| 2018-02-20T23:01:12.709-0500 I REPL     [eventTerminate] Stopping replication storage threads
d20002| 2018-02-20T23:01:12.713-0500 D REPL_HB  [replexec-0] Received response to heartbeat (requestId: 8371) from Yellanna-PC:200
d20002| 2018-02-20T23:01:12.714-0500 D REPL_HB  [replexec-0] Received response to heartbeat (requestId: 8373) from Yellanna-PC:200
d20002| 2018-02-20T23:01:12.715-0500 I FTDC     [eventTerminate] Shutting down full-time diagnostic data capture
d20002| 2018-02-20T23:01:12.747-0500 I STORAGE  [WTOplogJournalThread] oplog journal thread loop shutting down
d20002| 2018-02-20T23:01:12.748-0500 I STORAGE  [eventTerminate] WiredTigerKVEngine shutting down
d20002| 2018-02-20T23:01:14.261-0500 I STORAGE  [eventTerminate] shutdown: removing fs lock...
d20002| 2018-02-20T23:01:14.262-0500 I CONTROL  [eventTerminate] now exiting
d20002| 2018-02-20T23:01:14.262-0500 I CONTROL  [eventTerminate] shutting down with code:0
2018-02-20T23:01:14.315-0500 I -        [thread1] shell: stopped mongo program on port 20002
ReplSetTest stop *** Mongod in port 20002 shutdown with code (0) ***
ReplSetTest stopSet deleting all dbpaths
ReplSetTest stopSet *** Shut down repl set - test worked ****
```