

TUGAS 4 BASIS DATA LANJUT

REPLACE MONGODB

Nama : Mega Aprilia Santoso

NIM : 17050623026

Langkah-langkah :

1. Pertama

```
> replicaSet = new ReplSetTest({"nodes" : 3})
Starting new replica set __unknown_name__
{
  "kDefaultTimeoutMS" : 600000,
  "getReadConcernMajorityOpTimeOrThrow" : function (conn) {
"use strict";

    const majorityOpTime = _getReadConcernMajorityOpTime(conn);
    if (friendlyEqual(majorityOpTime, {ts: Timestamp(0, 0), t: NumberLong(0)})) {
      throw new Error("readConcern majority optime not available");
    }
    return majorityOpTime;
  },
  "nodeList" : function () {
"use strict";

    var list = [];
    for (var i = 0; i < this.ports.length; i++) {
      list.push(this.host + ":" + this.ports[i]);
    }

    return list;
  },
  "getNodeId" : function (node) {
"use strict";

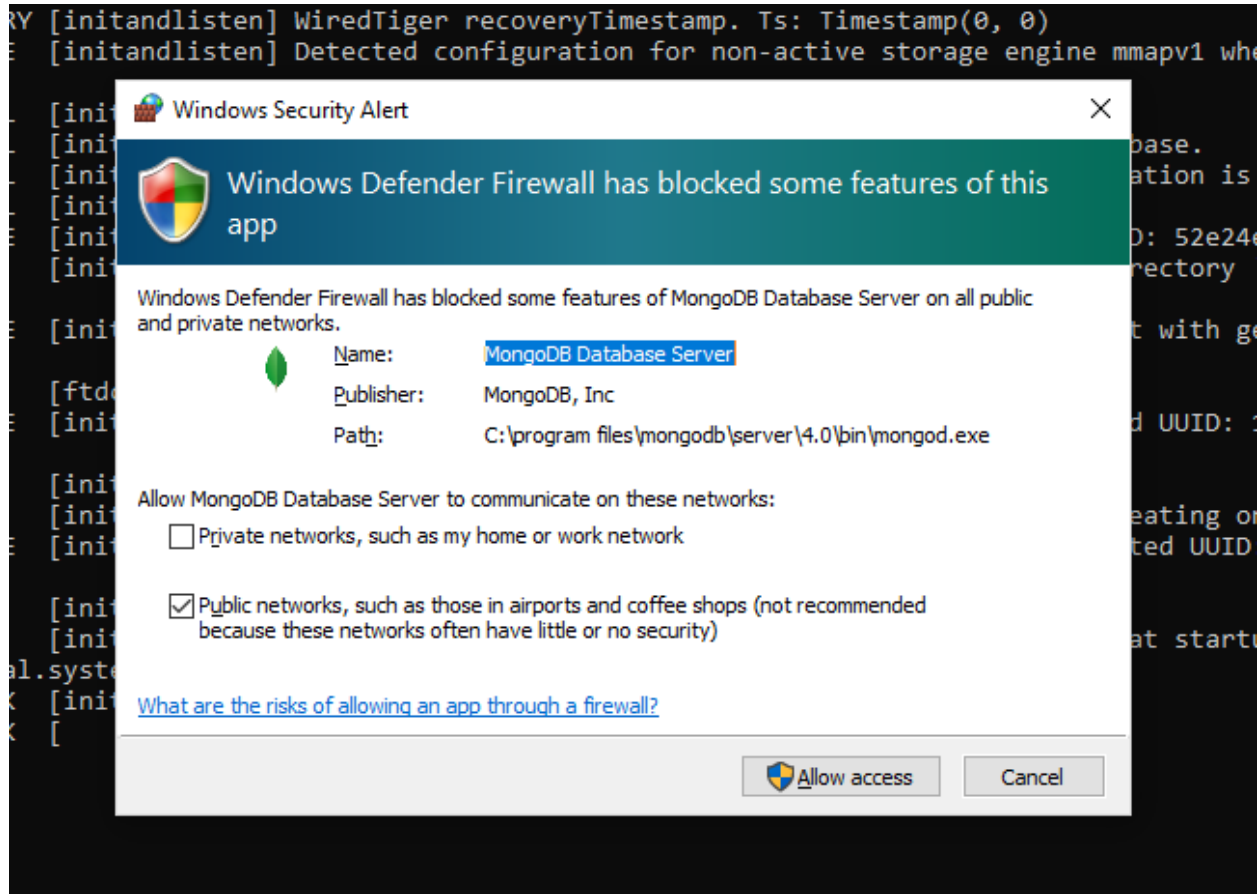
    if (node.toFixed) {
      return parseInt(node);
    }

    for (var i = 0; i < this.nodes.length; i++) {
```

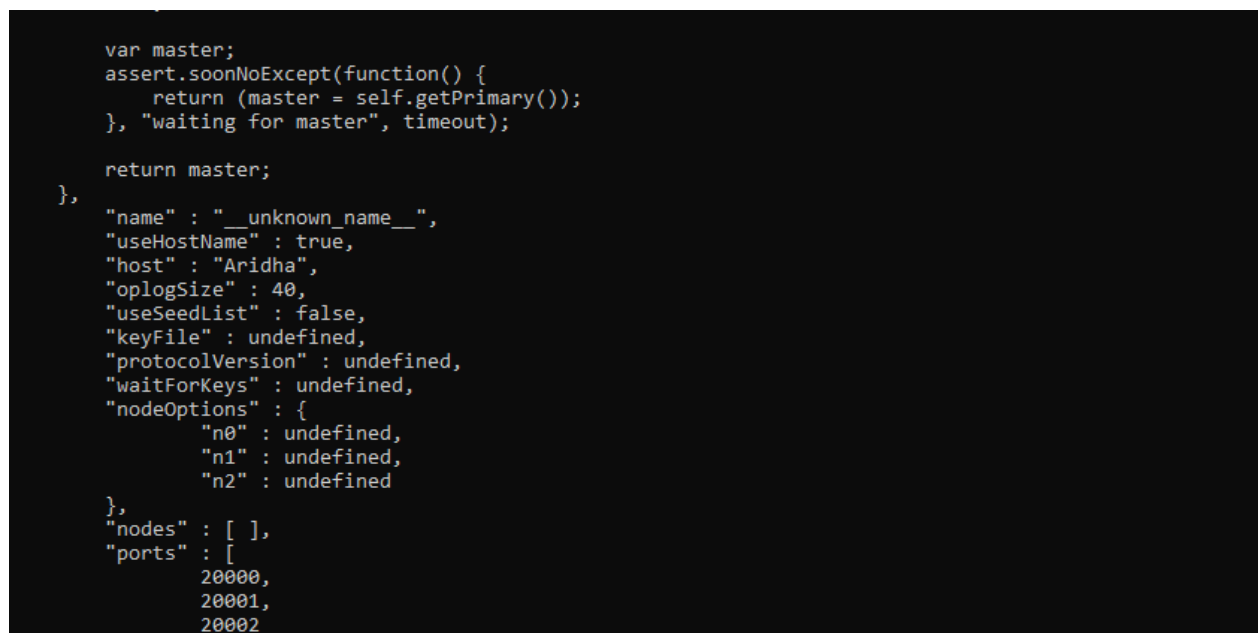
2. Kedua,

```
> replicaSet.startSet()
ReplSetTest starting set
ReplSetTest n is : 0
{
  "useHostName" : true,
  "oplogSize" : 40,
  "keyFile" : undefined,
  "port" : 28000,
  "noprealloc" : "",
  "smallfiles" : "",
  "replSet" : "__unknown_name__",
  "dbpath" : "$set-$node",
  "restart" : undefined,
  "pathOpts" : {
    "node" : 0,
    "set" : "__unknown_name__"
  },
  "setParameter" : {
    "writePeriodicNoops" : false,
    "numInitialSyncAttempts" : 1,
    "numInitialSyncConnectAttempts" : 60
  }
}
ReplSetTest Starting....
Resetting db path '/data/db/__unknown_name__-0'
2019-04-25T11:52:41.194+0700 I - [js] shell: started program (sh6564): C:\Program Files\MongoDB\Server\4.0\bin\mongod.exe --oplogSize 40 --port 28000 --
```

3. ketiga,



4. keempat,



5. Selanjutnya,

```
> replicaSet.initiate()
{
  "replSetInitiate": {
    "_id": "unknown_name_",
    "protocolVersion": 1,
    "members": [
      {
        "_id": 0,
        "host": "Aridha:20000"
      }
    ]
  }
}
d20000 2019-04-25T12:02:10.322+0700 I REPL [conn1] replSetInitiate admin command received from client
d20000 2019-04-25T12:02:10.340+0700 I REPL [conn1] replSetInitiate config object with 1 members parses ok
d20000 2019-04-25T12:02:10.354+0700 I REPL [conn1] *****
d20000 2019-04-25T12:02:10.354+0700 I REPL [conn1] creating replication oplog of size: 40MB...
d20000 2019-04-25T12:02:10.354+0700 I STORAGE [conn1] createCollection: local.oplog.rs with generated UUID: 40725cc7-6554-4e5a-9cc0-83d744e96c8d
d20000 2019-04-25T12:02:10.403+0700 I STORAGE [conn1] Starting OplogTruncaterThread local.oplog.rs
d20000 2019-04-25T12:02:10.404+0700 I STORAGE [conn1] The size storer reports that the oplog contains 0 records totaling to 0 bytes
d20000 2019-04-25T12:02:10.404+0700 I STORAGE [conn1] Scanning the oplog to determine where to place markers for truncation
d20000 2019-04-25T12:02:10.580+0700 I REPL [conn1] *****
d20000 2019-04-25T12:02:10.580+0700 I STORAGE [conn1] createCollection: local.system.replset with generated UUID: 6de3a410-eff1-4afb-9377-37d79c7a09b1
d20000 2019-04-25T12:02:10.581+0700 I COMMAND [conn1] monitoring keys for HMAC command admin.system.keys command: find { find: "system.keys", filter: { purpose: "HM
AC", expiresAt: { $gt: Timestamp(0, 0) } }, sort: { expiresAt: 1 }, $readPreference: { mode: "nearest", tags: [] }, $db: "admin" } planSummary: EOF keysExamined
:0 docsExamined:0 cursorExhausted:1 numYields:0 nreturned:0 reslen:231 locks:{ Global: { acquireCount: { r: 2 }, acquireWaitCount: { r: 1 }, timeAcquiringMicros
```

6. Kemudian.

```
> conn1 = new Mongo("localhost:20000")
d20000 2019-04-25T12:44:11.262+0700 I NETWORK [listener] connection accepted from 127.0.0.1:53405 #19 (9 connections now open)
d20000 2019-04-25T12:44:11.262+0700 I NETWORK [conn19] received client metadata from 127.0.0.1:53405 conn19: { application: { name: "MongoDB Shell" }, driver: { name: "MongoDB Internal Client", version: "4.0.6" }, os: { type: "Windows", name: "Microsoft Windows 10", architecture: "x86_64" }, version: "10.0 (build 17134)" }
connection to localhost:20000
> testReplSet:PRIMARY
2019-04-25T12:44:35.395+0700 E QUERY [js] ReferenceError: PRIMARY is not defined :
@(@shell):1:13
[main] automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'
> testReplSet:PRIMARY> [initandlisten] MongoDB starting : pid=7840 port=27017 dbpath=C:\data\db\ 64-bit host=DESKTOP-KPS9R2U
... testReplSet:PRIMARY> primaryDB = conn1.getDB("test")
2019-04-25T12:45:21.712+0700 E QUERY [js] SyntaxError: missing ; before statement @(@shell):2:11
> testReplSet:PRIMARY> testReplSet:PRIMARY> primaryDB = conn1.getDB("test")
2019-04-25T12:45:32.072+0700 E QUERY [js] SyntaxError: missing ; before statement @(@shell):1:32
> testReplSet:PRIMARY> primaryDB = conn1.getDB("test")
2019-04-25T12:46:28.713+0700 E QUERY [js] ReferenceError: invalid assignment left-hand side @(@shell):1:12
> primaryDB = conn1.getDB("test")
[initandlisten] distarch: x86_64
[initandlisten] target_arch: x86_64
> primaryDB.isMaster()
[initandlisten] options: {}
[initandlisten] Detected data files in C:\data\db\ created by the 'wiredTiger' storage engine, so setting the active storage
engine to 'wiredTiger'
[initandlisten] {
  "storage": {
    "engine": "wiredTiger"
  },
  "hosts": [
    {
      "name": "DESKTOP-KPS9R2U:20000",
      "wiredtiger_open_config": "create,cache_size=1470M,session_max=20000,eviction=(threads_min=4,threads_max=4),comp
ressor=(fast),statistics=(fast),log=(enabled=true,path=journal,compressor=snappy),file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=
0"
    },
    {
      "name": "DESKTOP-KPS9R2U:20002"
    }
  ],
  "setName": "rs0",
  "setVersion": 2,
  "ismaster": true,
  "secondary": false,
  "primary": "DESKTOP-KPS9R2U:20000",
  "me": "DESKTOP-KPS9R2U:20000",
  "electionId": "7fffffff00000000000000000001",
  "lastWrite": {
    "opTime": {
      "ts": Timestamp(1556170024, 230279),
      "t": NumberLong(1)
    },
    "majorityWriteDate": ISODate("2019-04-25T05:31:43Z")
  },
  "majorityOpTime": {
    "ts": Timestamp(1556170024, 230279),
    "t": NumberLong(1)
  },
  "majorityWriteDate": ISODate("2019-04-25T05:31:43Z")
}
[initandlisten] WiredTiger message [1556170024:590353][7840:140703613930576], txn-recover: Main recovery loop: starting at 3
[initandlisten] WiredTiger message [1556170024:590353][7840:140703613930576], txn-recover: Recovering log 3 through 4
[initandlisten] WiredTiger message [1556170025:43623][7840:140703613930576], txn-recover: Recovering log 4 through 4
[initandlisten] WiredTiger message [1556170025:187662][7840:140703613930576], txn-recover: Set global recovery timestamp: 0
[initandlisten] WiredTiger recoveryTimestamp: Ts: Timestamp(0, 0)
[initandlisten] ** WARNING: Access control is not enabled for the database.
[initandlisten] Read and write access to data and configuration is unrestricted.
[initandlisten] ** WARNING: This server is bound to localhost.
[initandlisten] Remote systems will be unable to connect to this server.
[initandlisten] To allow this server to be accessible, pass --bind_ip <address> to specify which IP
[initandlisten] addresses it should serve responses from, or with --bind_ip_all to
[initandlisten] bind to all interfaces. If this behavior is desired, start the
[initandlisten] server with --bind_ip 127.0.0.1 to disable this warning.
[initandlisten]
[initandlisten] diagnostic data capture with directory 'C:\data\db\diagnostic.data'
```

7. lalu,

```
> use('Admin')
> for (i=0; i<1000; i++) { primaryDB.coll.insert({count: 1}) }
d20000 2019-04-25T12:55:34.363+0700 I STORAGE [conn19] createCollection: test.coll with generated UUID: aec8e6ee-3ba4-460d-9f9f-5247f62a2baa
d20000 2019-04-25T12:55:34.637+0700 I COMMAND [conn19] command test.coll appName: "MongoDB Shell" command: Insert { insert: "coll", ordered: true, lsid: { id: UUID(
"05a5809d-5793-47de-b1db-ae1ef90526c9") }, $clusterTime: { clusterTime: Timestamp(1556170303, 1), signature: { hash: BinData(0, 00000000000000000000000000000000
00), keyId: 0 } }, $db: "test" }, ninserted:1 keysInserted:0 numYields:0 reslen:230 locks:{ Global: { acquireCount: { r: 3, w: 3 } }, Database: { acquireCount: { w: 2,
r: 1 } }, Collection: { acquireCount: { w: 2 } } } protocol:op_msg 274ms
d20001 2019-04-25T12:55:34.730+0700 I STORAGE [repl writer worker 10] createCollection: test.coll with provided UUID: aec8e6ee-3ba4-460d-9f9f-5247f62a2baa
d20002 2019-04-25T12:55:34.730+0700 I STORAGE [repl writer worker 10] createCollection: test.coll with provided UUID: aec8e6ee-3ba4-460d-9f9f-5247f62a2baa
d20001 2019-04-25T12:55:35.142+0700 I REPL [repl writer worker 10] applied op: command { ts: Timestamp(1556171734, 1), t: 1, h: 3867649606046479784, v: 2, op: "c
", ns: "test.$cmd", ui: UUID("aec8e6ee-3ba4-460d-9f9f-5247f62a2baa"), wall: new Date(1556171734636), o: { create: "coll", idIndex: { v: 2, key: { _id: 1 }, name: "_id
", ns: "test.coll" } } }, took 414ms [initandlisten] allocator: tcmalloc
d20002 2019-04-25T12:55:35.274+0700 I REPL [repl writer worker 10] applied op: command { ts: Timestamp(1556171734, 1), t: 1, h: 3867649606046479784, v: 2, op: "c
", ns: "test.$cmd", ui: UUID("aec8e6ee-3ba4-460d-9f9f-5247f62a2baa"), wall: new Date(1556171734636), o: { create: "coll", idIndex: { v: 2, key: { _id: 1 }, name: "_id
", ns: "test.coll" } } }, took 547ms [initandlisten] distmod: 2008plus-ssl
d20000 2019-04-25T12:55:35.469+0700 I NETWORK [conn18] end connection 127.0.0.1:53399 (8 connections now open)
writeResult({ "inserted": 1 }) [initandlisten] target arch: x86_64
> 19-04-24T22:27:03.457-0700 I CONTROL [initandlisten] options: {}
> 19-04-24T22:27:03.557-0700 I STORAGE [initandlisten] Detected data files in C:\data\db\ created by the 'wiredtiger' storage engine, so setting the active storage e
> use('test')
> use('wiredtiger')
> primaryDB.coll.count() 700 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=1476M,session_max=20000,eviction=(threads_min=4,threads_max=4),con
10000,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=1
> query progress)
> 19-04-24T22:27:04.230-0700 I STORAGE [initandlisten] WiredTiger message [1556170024:230279][7840:140703613930576], txn-recover: Main recovery loop: starting at 3/
144 to 4/150
```

8. kedelapan,

```
> conn2 = new Mongo("localhost:20002")
d20002 2019-04-25T13:16:54.607+0700 I NETWORK [listener] connection accepted from 127.0.0.1:53412 #14 (4 connections now open)
d20002 2019-04-25T13:16:54.608+0700 I NETWORK [conn14] received client metadata from 127.0.0.1:53412 conn14: { application: { name: "MongoDB Sh
ell" }, driver: { name: "MongoDB Internal Client", version: "4.0.6" }, os: { type: "Windows", name: "Microsoft Windows 10", architecture: "x86_64
", version: "10.0 (build 17134)" } }
connection to localhost:20002 [initandlisten] wiredtiger_open config: create,cache_size=1476M,session_max=20000,eviction=(threads_min=4,threads_max=4),e
hive=true,path=journal,compressor=snappy,file_manager=(close_idle_time=100000),statistics_log=(wait=0),verbose=1
> secondaryDB = conn2.getDB("test")
2019-04-25T13:17:17.993+0700 E QUERY [js] ReferenceError: Conn2 is not defined :
@(:shell):1:1
> secondaryDB = conn2.getDB("test")
test [initandlisten] WiredTiger message [1556170024:590353][7840:140703613930576], txn-recover: Recovering log 3 through 4
> secondaryDB.coll.find() [initandlisten] WiredTiger message [1556170025:43623][7840:140703613930576], txn-recover: Recovering log 4 through 4
Error: error: {
  "code": 13435,
  "operationTime": Timestamp(1556171736, 650),
  "ok": 0,
  "errmsg": "not master and slaveOk=false",
  "code": 13435,
  "codeName": "NotMasterNoSlaveOk",
  "$clusterTime": {
    "clusterTime": Timestamp(1556171736, 650),
    "signature": {
      "hash": BinData(0, "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId": NumberLong(0)
    }
  }
} [initandlisten] WARNING: This server is bound to localhost.
systems will be unable to connect to this server.
Start the server with --bind_ip <address> to specify which IP
address to bind to, or with --bind_ip_all to bind to all interfaces. If this behavior is desired, start the
server with --bind_ip 127.0.0.1 to disable this warning.
> 17-063+0700 I FTDC [initandlisten] Initializing full-time diagnostic data capture with directory 'C:\data\db\diagnostic.data'
> 17-162+0700 I NETWORK [initandlisten] waiting for connections on port 27017
```


9. terakhir, syntax "find"

```
> conn2.setSlaveOK()
3.457-8788 I CONTROL [initandlisten] target_arch: x86_64
> secondaryDB.coll.find()
{"_id" : ObjectId("5cc14bd6faef356afab69fd4"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fcc"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fcb"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fce"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd0"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd6"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd5"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fcd"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd2"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fca"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd1"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd9"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fda"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd3"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fcf"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd7"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fd8"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fdf"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69feb"), "count" : 1 }
{"_id" : ObjectId("5cc14bd6faef356afab69fee"), "count" : 1 }
Type "it" for more
> secondaryDB.coll.count()
1000
07.162+0700 I NETWORK [initandlisten] Initializing full-time diagnostic data capture with directory ./diaglog
07.162+0700 I NETWORK [initandlisten] waiting for connections on port 27017
```