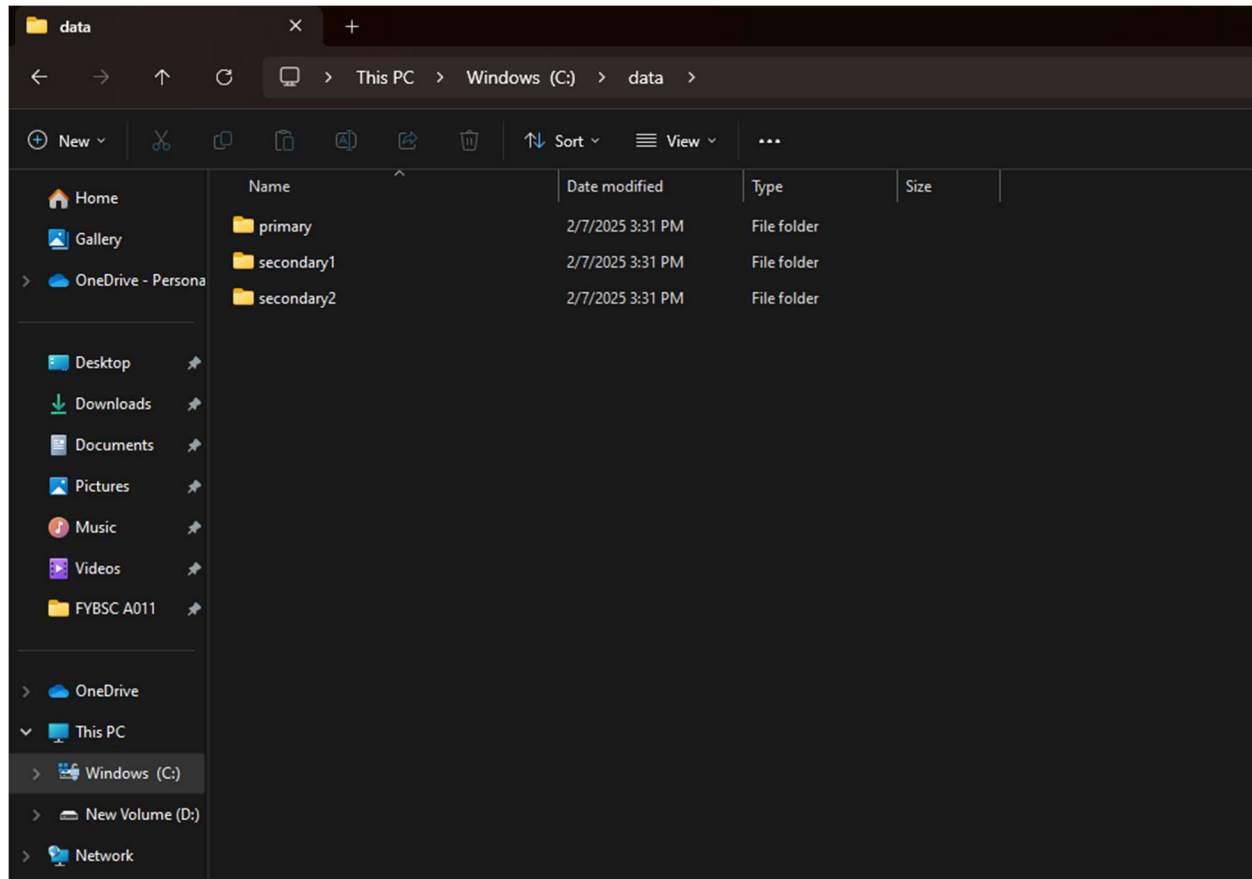


PRACTICAL 6

REPLICATION USING MONGODB



```
PS C:\Users\Admin> mongod --port=2717 --dbpath="C:\data\primary" --replSet="test-replica-set"
{"t":{"$date":"2025-02-07T15:21:45.955+05:30"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"thread1", "msg":"Initialize
d wire specification", "attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":21},"incomingInterna
lClient":{"minWireVersion":0,"maxWireVersion":21},"outgoing":{"minWireVersion":6,"maxWireVersion":21},"isInternalClient"
:true}}}}
{"t":{"$date":"2025-02-07T15:21:45.955+05:30"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"thread1", "msg":"Automatica
lly disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2025-02-07T15:21:45.957+05:30"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"thread1", "msg":"Implicit T
CP FastOpen in use."}
{"t":{"$date":"2025-02-07T15:21:45.968+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successful
ly registered PrimaryOnlyService", "attr":{"service":"TenantMigrationDonorService", "namespace":"config.tenantMigrationDon
ors"}}}
{"t":{"$date":"2025-02-07T15:21:45.969+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successful
ly registered PrimaryOnlyService", "attr":{"service":"TenantMigrationRecipientService", "namespace":"config.tenantMigratio
nRecipients"}}}
{"t":{"$date":"2025-02-07T15:21:45.969+05:30"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"thread1", "msg":"Multi thre
ading initialized"}
{"t":{"$date":"2025-02-07T15:21:45.969+05:30"},"s":"I", "c":"TENANT_M", "id":7091600, "ctx":"thread1", "msg":"Starting T
enantMigrationAccessBlockerRegistry"}
{"t":{"$date":"2025-02-07T15:21:45.970+05:30"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten", "msg":"Mong
oDB starting", "attr":{"pid":5728, "port":2717, "dbPath":"C:/data/primary", "architecture":"64-bit", "host":"MUM2024COM17"}}
{"t":{"$date":"2025-02-07T15:21:45.970+05:30"},"s":"I", "c":"CONTROL", "id":23398, "ctx":"initandlisten", "msg":"Targ
et operating system minimum version", "attr":{"targetMinOS":"Windows 7/Windows Server 2008 R2/11}}
```

```
PS C:\Users\Admin> mongod --port=2727 --dbpath="C:\data\secondary1" --replSet="test-replica-set"
{"t":{"$date":"2025-02-07T15:24:55.647+05:30"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"thread1", "msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2025-02-07T15:24:55.650+05:30"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"thread1", "msg":"Initializing wire specification", "attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":21},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":21},"outgoing":{"minWireVersion":6,"maxWireVersion":21},"isInternalClient":true}}}
{"t":{"$date":"2025-02-07T15:24:55.651+05:30"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"thread1", "msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2025-02-07T15:24:55.652+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationDonorService", "namespace":"config.tenantMigrationDonors"}}
{"t":{"$date":"2025-02-07T15:24:55.652+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationRecipientService", "namespace":"config.tenantMigrationRecipients"}}
{"t":{"$date":"2025-02-07T15:24:55.652+05:30"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"thread1", "msg":"Multi threading initialized"}
{"t":{"$date":"2025-02-07T15:24:55.652+05:30"},"s":"I", "c":"TENANT_M", "id":7091600, "ctx":"thread1", "msg":"Starting TenantMigrationAccessBlockerRegistry"}
{"t":{"$date":"2025-02-07T15:24:55.653+05:30"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten", "msg":"MongoDB starting", "attr":{"pid":13412, "port":2727, "dbPath":"C:/data/secondary1", "architecture":"64-bit", "host":"MUM2024COM17"}}
```

```
PS C:\Users\Admin> mongod --port=2737 --dbpath="C:\data\secondary2" --replSet="test-replica-set"
{"t":{"$date":"2025-02-07T15:26:40.029+05:30"},"s":"I", "c":"NETWORK", "id":4915701, "ctx":"thread1", "msg":"Initializing wire specification", "attr":{"spec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":21},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":21},"outgoing":{"minWireVersion":6,"maxWireVersion":21},"isInternalClient":true}}}
{"t":{"$date":"2025-02-07T15:26:41.039+05:30"},"s":"I", "c":"CONTROL", "id":23285, "ctx":"thread1", "msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2025-02-07T15:26:41.041+05:30"},"s":"I", "c":"NETWORK", "id":4648602, "ctx":"thread1", "msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2025-02-07T15:26:41.043+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationDonorService", "namespace":"config.tenantMigrationDonors"}}
{"t":{"$date":"2025-02-07T15:26:41.043+05:30"},"s":"I", "c":"REPL", "id":5123008, "ctx":"thread1", "msg":"Successfully registered PrimaryOnlyService", "attr":{"service":"TenantMigrationRecipientService", "namespace":"config.tenantMigrationRecipients"}}
{"t":{"$date":"2025-02-07T15:26:41.043+05:30"},"s":"I", "c":"CONTROL", "id":5945603, "ctx":"thread1", "msg":"Multi threading initialized"}
{"t":{"$date":"2025-02-07T15:26:41.043+05:30"},"s":"I", "c":"TENANT_M", "id":7091600, "ctx":"thread1", "msg":"Starting TenantMigrationAccessBlockerRegistry"}
{"t":{"$date":"2025-02-07T15:26:41.045+05:30"},"s":"I", "c":"CONTROL", "id":4615611, "ctx":"initandlisten", "msg":"MongoDB starting", "attr":{"pid":11520, "port":2737, "dbPath":"C:/data/secondary2", "architecture":"64-bit", "host":"MUM2024COM17"}}
```

```
PS C:\Users\Admin> mongosh --host="localhost:2717"
Current Mongosh Log ID: 67a5daf245d0faa991a75a7a
Connecting to: mongod://localhost:2717/?directConnection=true&appName=mongosh+2.1.4
Using MongoDB: 7.0.12
Using Mongosh: 2.1.4
mongosh 2.3.8 is available for download: https://www.mongodb.com/try/download/shell
```

For mongosh info see: <https://docs.mongodb.com/mongodbs-shell/>

The server generated these startup warnings when booting
 2025-02-07T15:21:46.017+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
 2025-02-07T15:21:46.017+05:30: This server is bound to localhost. Remote systems will be unable to connect to this server. Start the server with --bind_ip <address> to specify which IP addresses it should serve responses from, 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

```
test> rs.initiate()
{
  info2: 'no configuration specified. Using a default configuration for the set',
  me: 'localhost:2717',
  ok: 1
}
```

```
test-replica-set [direct: primary] test> rs.add({host:"localhost:2727"})
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1738923195, i: 2 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1738923195, i: 2 })
}
```

```
test-replica-set [direct: primary] test> rs.add({host:"localhost:2737"})
{
  ok: 1,
  '$clusterTime': {
    clusterTime: Timestamp({ t: 1738923218, i: 1 }),
    signature: {
      hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0),
      keyId: Long('0')
    }
  },
  operationTime: Timestamp({ t: 1738923218, i: 1 })
}
```

To check if both the ports are available

```
test-replica-set [direct: primary] test> rs.status()
{
  set: 'test-replica-set',
  date: ISODate('2025-02-07T10:16:21.417Z'),
  myState: 1,
  term: Long('1'),
  syncSourceHost: '',
  syncSourceId: -1,
  heartbeatIntervalMillis: Long('2000'),
  majorityVoteCount: 2,
  writeMajorityCount: 2,
  votingMembersCount: 3,
  writableVotingMembersCount: 3,
  optimes: {
    lastCommittedOpTime: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
    lastCommittedWallTime: ISODate('2025-02-07T10:16:15.054Z'),
    readConcernMajorityOpTime: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
    appliedOpTime: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
    durableOpTime: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
    lastAppliedWallTime: ISODate('2025-02-07T10:16:15.054Z'),
    lastDurableWallTime: ISODate('2025-02-07T10:16:15.054Z')
  },
}
```

```

{
  _id: 2,
  name: 'localhost:2737',
  health: 1,
  state: 2,
  stateStr: 'SECONDARY',
  uptime: 163,
  optime: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
  optimeDurable: { ts: Timestamp({ t: 1738923375, i: 1 }), t: Long('1') },
  optimeDate: ISODate('2025-02-07T10:16:15.000Z'),
  optimeDurableDate: ISODate('2025-02-07T10:16:15.000Z'),
  lastAppliedWallTime: ISODate('2025-02-07T10:16:15.054Z'),
  lastDurableWallTime: ISODate('2025-02-07T10:16:15.054Z'),
  lastHeartbeat: ISODate('2025-02-07T10:16:20.453Z'),
  lastHeartbeatRecv: ISODate('2025-02-07T10:16:20.902Z'),
  pingMs: Long('0'),
  lastHeartbeatMessage: '',
  syncSourceHost: 'localhost:2727',
  syncSourceId: 1,
  infoMessage: '',
  configVersion: 5,
  configTerm: 1
},
ok: 1,
'$clusterTime': {
  clusterTime: Timestamp({ t: 1738923375, i: 1 }),
  signature: {
    hash: Binary.createFromBase64('AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA=', 0)
  }
}

```

Create instances for secondary1 and secondary2

```

PS C:\Users\Admin> mongosh --host="localhost:2727"
Current Mongosh Log ID: 67a5dd42a26885d4f476d4a9
Connecting to:      mongodb://localhost:2727/?directConnection=true&appName=mongosh+2.1.4
Using MongoDB:      7.0.12
Using Mongosh:       2.1.4
mongosh 2.3.9 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://docs.mongodb.com/mongosh-shell/

The server generated these startup warnings when booting
2025-02-07T15:24:55.692+05:30: Access control is not enabled for the database. Read and write access to
figuration is unrestricted
2025-02-07T15:24:55.692+05:30: This server is bound to localhost. Remote systems will be unable to connect
to this server. Start the server with --bind_ip <address> to specify which IP addresses it should serve responses from
or --bind_ip_all to bind to all interfaces. If this behavior is desired, start the server with --bind_ip 127.0.0.0
to disable this warning

test-replica-set [direct: secondary] test> |

```



```

PS C:\Users\Admin> mongosh --host="localhost:2737"
Current Mongosh Log ID: 67a5df2180dc0b0068cff333
Connecting to:      mongodb://localhost:2737/?directConnection=true&appName=mongosh+2.1.4
Using MongoDB:      7.0.12
Using Mongosh:       2.1.4
mongosh 2.3.9 is available for download: https://www.mongodb.com/try/download/shell

For mongosh info see: https://docs.mongodb.com/mongodb-shell/



---


The server generated these startup warnings when booting
2025-02-07T15:26:41.083+05:30: Access control is not enabled for the database. Read and write acc
figuration is unrestricted
2025-02-07T15:26:41.084+05:30: This server is bound to localhost. Remote systems will be unable t
rver. Start the server with --bind_ip <address> to specify which IP addresses it should serve respon
bind_ip_all to bind to all interfaces. If this behavior is desired, start the server with --bind_ip
this warning

```

Perform Crud operations:

1.insert

```

test-replica-set [direct: primary] test> show dbs;
admin      80.00 KiB
config    212.00 KiB
local     404.00 KiB
test-replica-set [direct: primary] test> use practice;
switched to db practice
test-replica-set [direct: primary] practice> db.users.insertMany([{"name":"Swati",age:28},{"name":"Prachi",age:17}])
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('67a5dfe595ca045275b5a522'),
    '1': ObjectId('67a5dfe595ca045275b5a523')
  }
}

```

```

test-replica-set [direct: secondary] test> use practice;
switched to db practice
test-replica-set [direct: secondary] practice> db.users.find();
MongoServerError[NotPrimaryNoSecondaryOk]: not primary - consider using db.getMongo()
the connection string
test-replica-set [direct: secondary] practice> db.getMongo().setReadPref("secondary")
test-replica-set [direct: secondary] practice> db.users.find();
[
  {
    _id: ObjectId('67a5dfe595ca045275b5a523'),
    name: 'Prachi',
    age: 17
  },
  { _id: ObjectId('67a5dfe595ca045275b5a522'), name: 'Swati', age: 28 }
]

```

2.update

```
test-replica-set [direct: primary] practice> db.users.updateOne({name:"Swati"},{$set:{age:25}});
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 1,
  modifiedCount: 1,
  upsertedCount: 0
}
```

```
test-replica-set [direct: secondary] practice> db.users.find();
[
  {
    _id: ObjectId('67a5dfe595ca045275b5a523'),
    name: 'Prachi',
    age: 17
  },
  { _id: ObjectId('67a5dfe595ca045275b5a522'), name: 'Swati', age: 25 }
]
```

3.delete

```
test-replica-set [direct: primary] practice> db.users.deleteOne({name:"Prachi"});
{ acknowledged: true, deletedCount: 1 }
```

```
test-replica-set [direct: secondary] practice> db.users.find();
[
  { _id: ObjectId('67a5dfe595ca045275b5a522'), name: 'Swati', age: 25 }
]
```