

Implementation of Live Updates from MongoDB Server in Spring boot app

Step 1: Installation of MongoDB Server

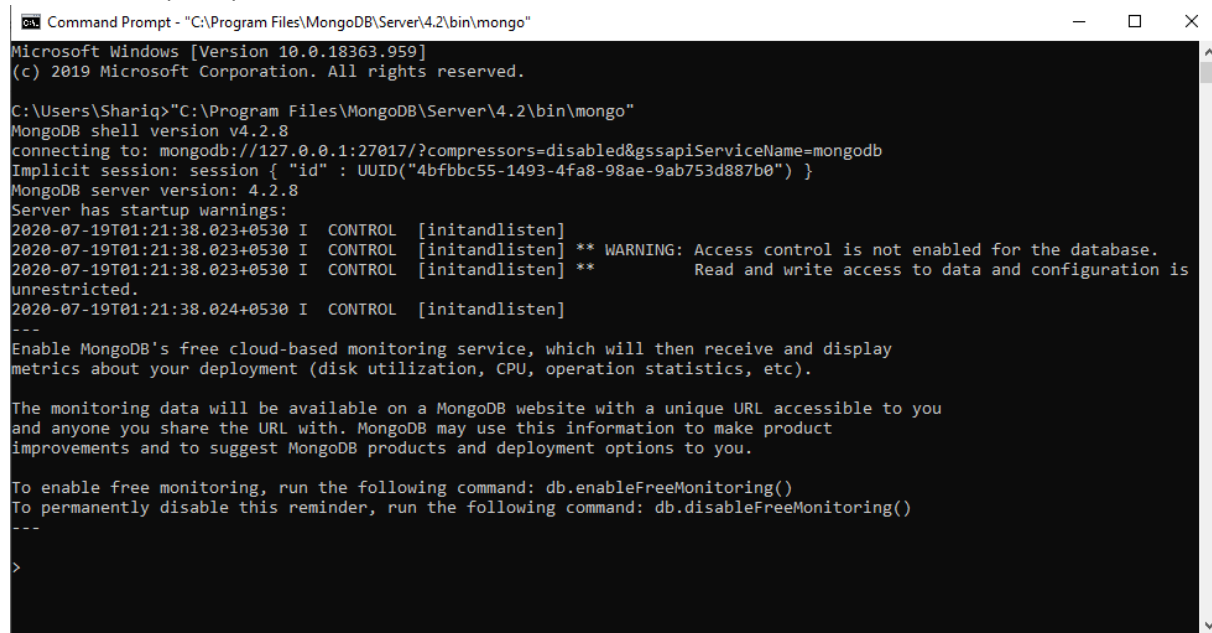
Go to <https://www.mongodb.com/try/download/community> and follow instructions.

Step 2: Start the server in normal mode

Assuming installation in C drive of Windows OS

Copy the path "C:\Program Files\MongoDB\Server\4.2\bin\mongo"

In command prompt like this



```
Command Prompt - "C:\Program Files\MongoDB\Server\4.2\bin\mongo"
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Shariq>"C:\Program Files\MongoDB\Server\4.2\bin\mongo"
MongoDB shell version v4.2.8
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("4bfbbc55-1493-4fa8-98ae-9ab753d887b0") }
MongoDB server version: 4.2.8
Server has startup warnings:
2020-07-19T01:21:38.023+0530 I CONTROL [initandlisten]
2020-07-19T01:21:38.023+0530 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2020-07-19T01:21:38.023+0530 I CONTROL [initandlisten] ** Read and write access to data and configuration is
unrestricted.
2020-07-19T01:21:38.024+0530 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
>
```

At this stage server is already started you can create db and collections as per your need.

Also to see visually install MongoDB Compass from here

<https://www.mongodb.com/try/download/compass>

Note: default port is 27017

<https://docs.mongodb.com/manual/reference/command/>

Follow this to learn commands

Step 3: The change stream

Change stream is a utility in MongoDB which provides live updates on certain DDL commands of the selected Collection or DB.

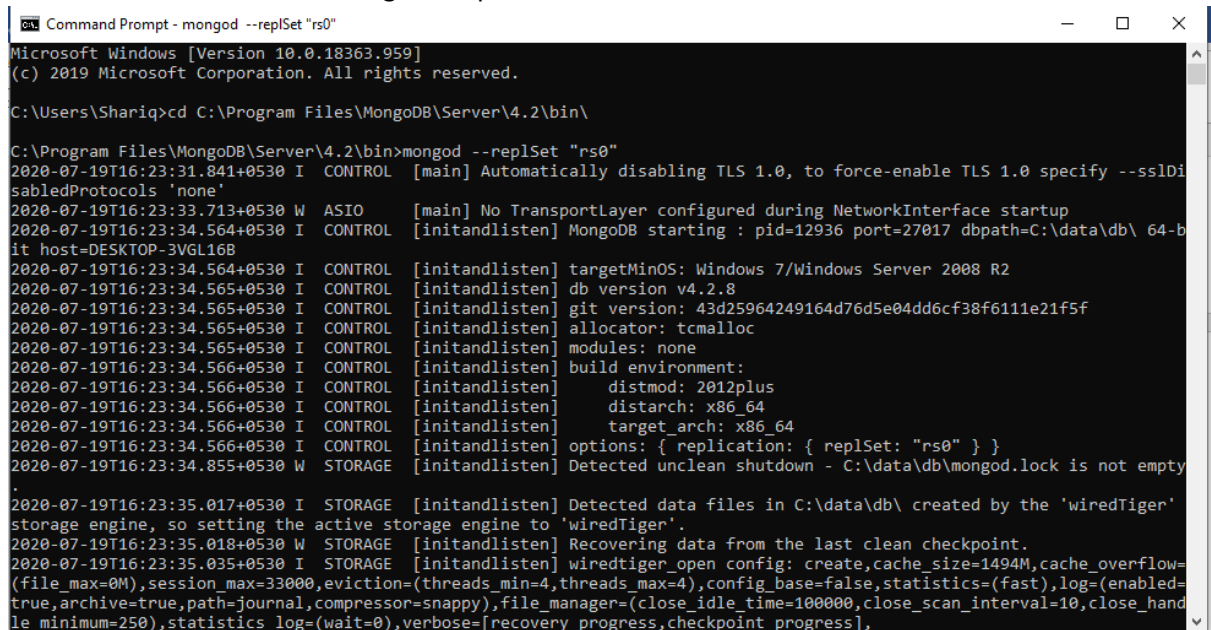
Change stream, required the Replica of primary node of the server to operate. Change stream reads the OpLog of the mongo operations for live updates.

To create replica:

<https://docs.mongodb.com/manual/tutorial/convert-standalone-to-replica-set/>

Other way:

1. Start the mongoDB server as in Step 2
2. Open a separate window of Command Prompt
And navigate to the bin folder of MongoDB
And enter this Command <mongod --replSet "rs0" >



```
Microsoft Windows [Version 10.0.18363.959]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Shariq>cd C:\Program Files\MongoDB\Server\4.2\bin\

C:\Program Files\MongoDB\Server\4.2\bin>mongod --replSet "rs0"
2020-07-19T16:23:31.841+0530 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'
2020-07-19T16:23:33.713+0530 W ASIO [main] No TransportLayer configured during NetworkInterface startup
2020-07-19T16:23:34.564+0530 I CONTROL [initandlisten] MongoDB starting : pid=12936 port=27017 dbpath=C:\data\db\ 64-bit host=DESKTOP-3VGL16B
2020-07-19T16:23:34.564+0530 I CONTROL [initandlisten] targetMinOS: Windows 7/Windows Server 2008 R2
2020-07-19T16:23:34.565+0530 I CONTROL [initandlisten] db version v4.2.8
2020-07-19T16:23:34.565+0530 I CONTROL [initandlisten] git version: 43d25964249164d76d5e04dd6cf38f611e21f5f
2020-07-19T16:23:34.565+0530 I CONTROL [initandlisten] allocator: tcmalloc
2020-07-19T16:23:34.565+0530 I CONTROL [initandlisten] modules: none
2020-07-19T16:23:34.566+0530 I CONTROL [initandlisten] build environment:
2020-07-19T16:23:34.566+0530 I CONTROL [initandlisten] distmod: 2012plus
2020-07-19T16:23:34.566+0530 I CONTROL [initandlisten] distarch: x86_64
2020-07-19T16:23:34.566+0530 I CONTROL [initandlisten] target_arch: x86_64
2020-07-19T16:23:34.566+0530 I CONTROL [initandlisten] options: { replication: { replSet: "rs0" } }
2020-07-19T16:23:34.855+0530 W STORAGE [initandlisten] Detected unclean shutdown - C:\data\db\mongod.lock is not empty
2020-07-19T16:23:35.017+0530 I STORAGE [initandlisten] Detected data files in C:\data\db\ created by the 'wiredTiger' storage engine, so setting the active storage engine to 'wiredTiger'.
2020-07-19T16:23:35.018+0530 W STORAGE [initandlisten] Recovering data from the last clean checkpoint.
2020-07-19T16:23:35.035+0530 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=1494M,cache_overflow=(file_max=0M),session_max=33000,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file_manager=(close_idle_time=100000,close_scan_interval=10,close_handle_minimum=250),statistics_log=(wait=0),verbose=[recovery progress,checkpoint progress],
```

If server doesnot replicate do this:

1. In the server 1st window of CMD enter this command:
db.adminCommand({shutdown:1})
it will start restart the server and replication window will show positive oup

So far this is what I have learnt and some coding in Spring boot too.

1. Code for Changestream live updates:

```
import java.lang.reflect.Array;
import java.util.Arrays;
import java.util.List;

import org.bson.Document;
import org.bson.conversions.Bson;

import com.mongodb.Block;
import com.mongodb.client.ChangeStreamIterable;
import com.mongodb.client.MongoClient;
import com.mongodb.client.MongoClients;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.client.model.Aggregates;
import com.mongodb.client.model.Filters;
import com.mongodb.client.model.changestream.ChangeStreamDocument;
```

```

import com.mongodb.client.model.changestream.FullDocument;

public class MondoChangeStream {

    public static void main(String[] args) {
        MongoClient mongoClient =
MongoClients.create("mongodb://localhost:27017");
        MongoDB database = mongoClient.getDatabase("userdb");
        MongoCollection<Document> collection =
database.getCollection("tasks");

        //          Block<ChangeStreamDocument<Document>> printBlock = new
Block<ChangeStreamDocument<Document>>() {
        //              @Override
        //              public void apply(final ChangeStreamDocument<Document>
changeStreamDocument) {
        //                  System.out.println(changeStreamDocument);
        //              }
        //          };

        ChangeStreamIterable<Document> changes =
collection.watch(Arrays.asList(
            Aggregates.match( Filters.and( Arrays.asList(
                Filters.in("operationType",
Arrays.asList("insert", "update", "replace", "delete")),
                Filters.eq("fullDocument.even", 1L)))
            )));

        //          changes.forEach(new Block<ChangeStreamDocument<Document>>() {
        //              @Override
        //              public void apply(ChangeStreamDocument<Document> t) {
        //                  System.out.println("received: " + t.getFullDocument());
        //              }
        //          });

        changes.iterator().forEachRemaining(
            change -> System.out.println("received: " +
change.getFullDocument())
        );
    }

}

```

Maven Dependencies:

```

<dependency>
<groupId>org.mongodb</groupId>
<artifactId>mongodb-driver-legacy</artifactId>
<version>4.1.0-rc0</version>
</dependency>

```

