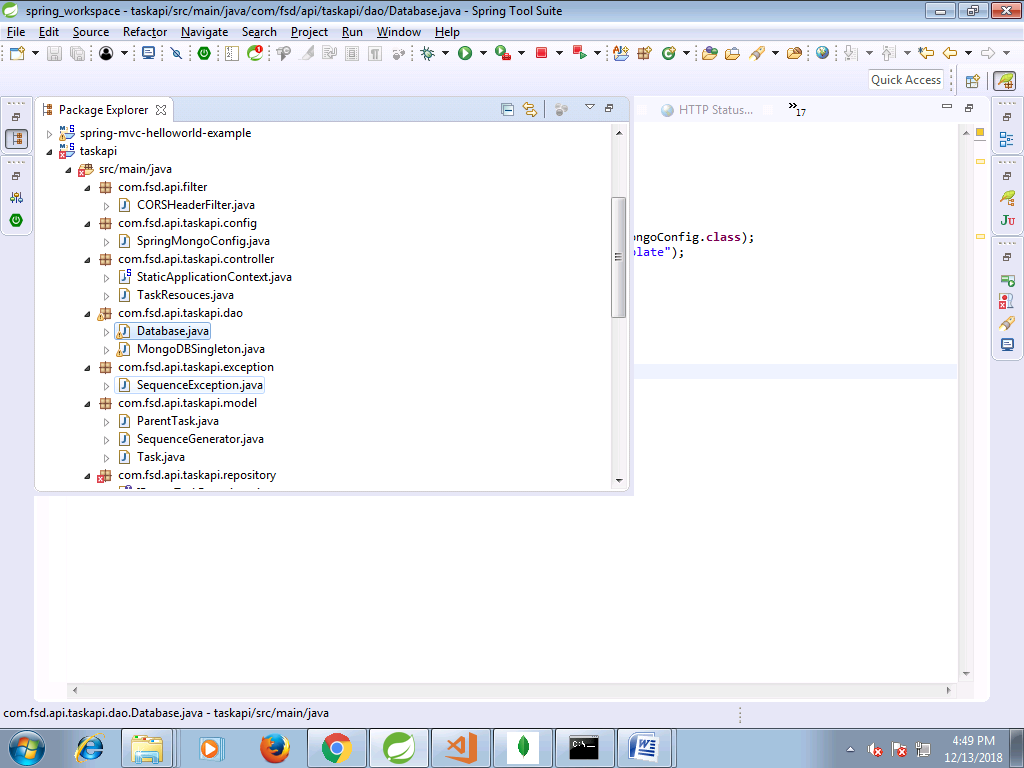
Project Structure



Database.java

# Database.java

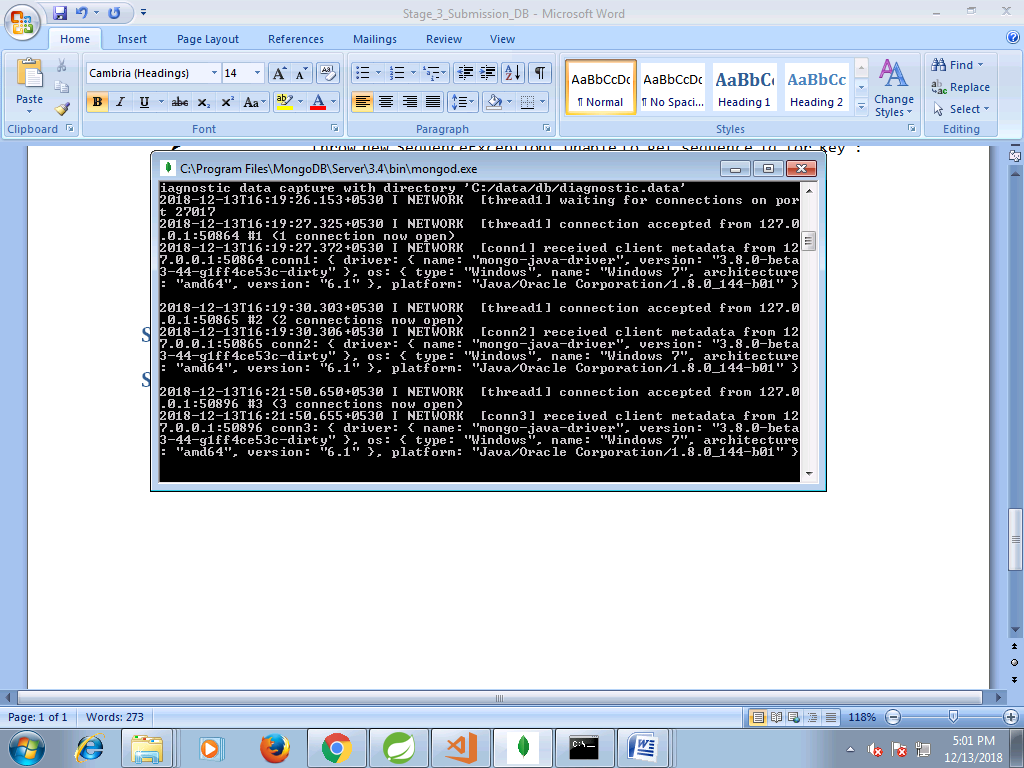
* package com.fsd.api.taskapi.dao;
* import java.util.Date;
* import org.springframework.context.ApplicationContext;
* import org.springframework.context.annotation.AnnotationConfigApplicationContext;
* import org.springframework.data.mongodb.core.MongoOperations;
* import org.springframework.data.mongodb.core.query.Criteria;
* import org.springframework.data.mongodb.core.query.Query;
* import com.fsd.api.taskapi.config.SpringMongoConfig;
* import com.fsd.api.taskapi.model.ParentTask;
* import com.fsd.api.taskapi.model.SequenceGenerator;
* import com.fsd.api.taskapi.model.Task;
* public class Database {
* public static void main(String[] args) {
* ApplicationContext ctx = new AnnotationConfigApplicationContext(SpringMongoConfig.class);
* MongoOperations mongoOperation = (MongoOperations)ctx.getBean("mongoTemplate");
* SequenceGenerator obj = new SequenceGenerator();
* obj.setCounter(1l);
* obj.setId("tbl\_task.taskid");
* // mongoOperation.save(obj);
* obj.setCounter(101l);
* obj.setId("tbl\_parenttask.parentid");
* mongoOperation.save(obj);
* }
* }

# SequenceDaoImpl.java

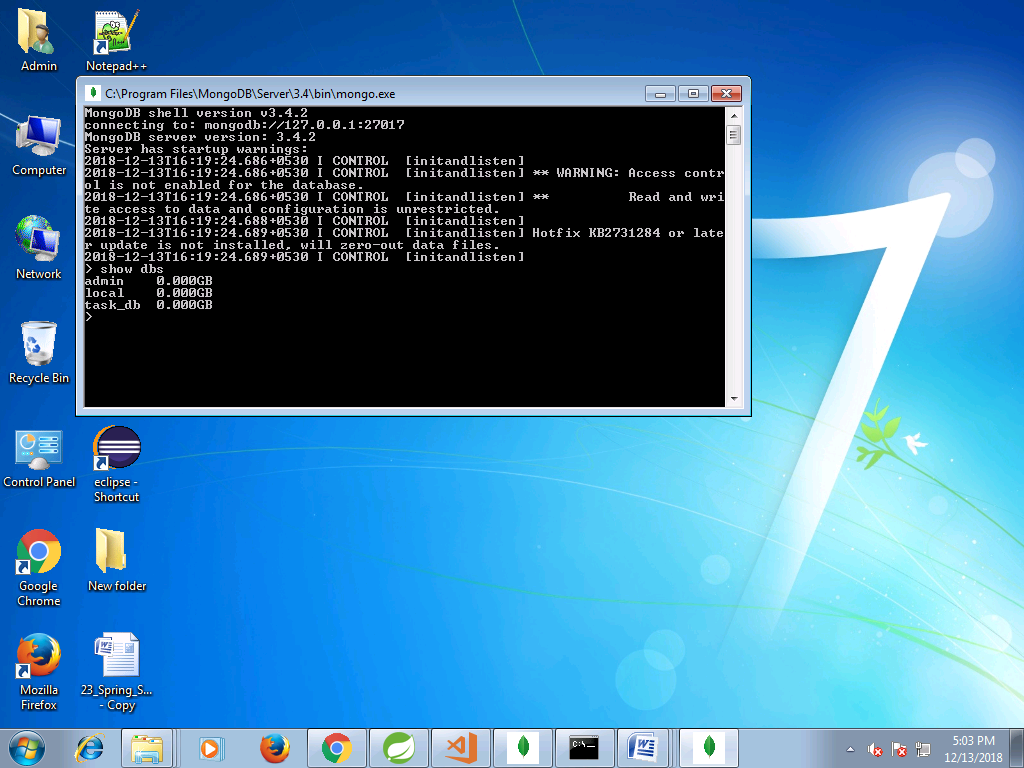
* package com.fsd.api.taskapi.repository;
* import org.springframework.beans.factory.annotation.Autowired;
* import org.springframework.data.mongodb.core.FindAndModifyOptions;
* import org.springframework.data.mongodb.core.MongoOperations;
* import org.springframework.data.mongodb.core.query.Criteria;
* import org.springframework.data.mongodb.core.query.Query;
* import org.springframework.data.mongodb.core.query.Update;
* import org.springframework.stereotype.Repository;
* import com.fsd.api.taskapi.exception.SequenceException;
* import com.fsd.api.taskapi.model.SequenceGenerator;
* @Repository
* public class SequenceDaoImpl implements ISequenceDao {
* @Autowired
* private MongoOperations mongoOperation;
* @Override
* public long getNextSequenceId(String key) throws SequenceException {
* //get sequence id
* Query query = new Query(Criteria.where("\_id").is(key));
* //increase sequence id by 1
* Update update = new Update();
* update.inc("counter", 1);
* //return new increased id
* FindAndModifyOptions options = new FindAndModifyOptions();
* options.returnNew(true);
* options.upsert(true);// To auto create sequence
* //this is the magic happened.
* SequenceGenerator seqId =
* mongoOperation.findAndModify(query, update, options, SequenceGenerator.class);
* //if no id, throws SequenceException
* //optional, just a way to tell user when the sequence id is failed to generate.
* if (seqId == null) {
* throw new SequenceException("Unable to get sequence id for key : " + key);
* }
* return seqId.getCounter();
* }
* }

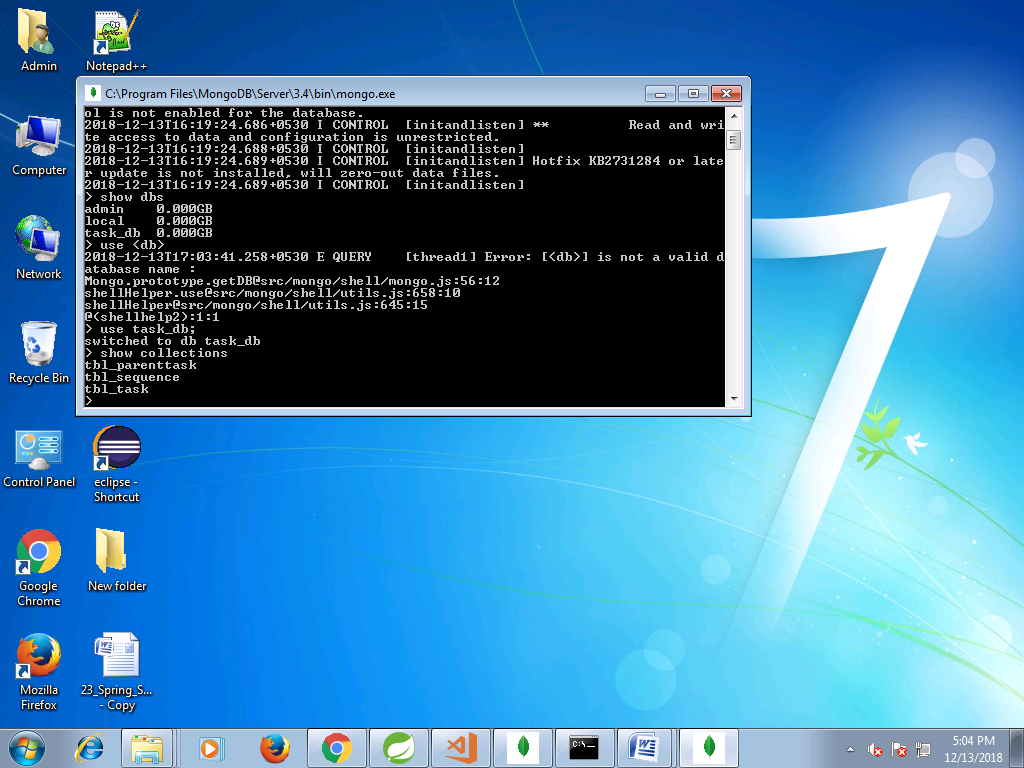
**Screenshots**

**Server start up**

****

**Database structure**

****

****