



Spring Data MongoDB hello world example

By mkyong (<http://www.mkyong.com/author/mkyong/>) | May 16, 2011 | Updated : June 2, 2013 | Viewed : 336,629 times +424 pv/w

In this tutorial, we show you how to use “SpringData for MongoDB (<http://www.springsource.org/spring-data/mongodb>)” framework, to perform CRUD operations in MongoDB, via Spring’s annotation and XML schema.

Updated on 1/04/2013

Article is updated to use latest SpringData v 1.2.0.RELEASE, it was v1.0.0.M2.

Tools and technologies used :

1. Spring Data MongoDB – 1.2.0.RELEASE
2. Spring Core – 3.2.2.RELEASE
3. Java Mongo Driver – 2.11.0
4. Eclipse – 4.2
5. JDK – 1.6
6. Maven – 3.0.3

P.S Spring Data requires JDK 6.0 and above, and Spring Framework 3.0.x and above.

1. Project Structure

A classic Maven’s style Java project directory structure.



Java APIs for Developers

Ad Office formats and Native Java Code Free support
and free evaluation

aspose.com

[Learn more](#)

2. Dependency

The following libraries are required :

~~spring-data-mongodb~~

~~Currently, the "spring-data-mongodb" jar is only available in "<http://maven.springframework.org/milestone>", so, you have to declare this repository also:~~

Updated on 13/09/2012

spring-data-mongodb is available at the Maven central repository, Spring repository is no longer required.

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mkyong.core</groupId>
  <artifactId>SpringMongoDBExample</artifactId>
  <packaging>jar</packaging>
  <version>1.0</version>
  <name>SpringMongoExample</name>
  <url>http://maven.apache.org</url>

  <dependencies>

    <!-- Spring framework -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-core</artifactId>
      <version>3.2.2.RELEASE</version>
    </dependency>

    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>3.2.2.RELEASE</version>
    </dependency>

    <!-- mongodb java driver -->
    <dependency>
      <groupId>org.mongodb</groupId>
      <artifactId>mongo-java-driver</artifactId>
      <version>2.11.0</version>
    </dependency>

    <!-- Spring data mongodb -->
    <dependency>
      <groupId>org.springframework.data</groupId>
      <artifactId>spring-data-mongodb</artifactId>
      <version>1.2.0.RELEASE</version>
    </dependency>

    <dependency>
      <groupId>cglib</groupId>
      <artifactId>cglib</artifactId>
      <version>2.2.2</version>
    </dependency>

  </dependencies>

  <build>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.0</version>
        <configuration>
          <source>1.6</source>
          <target>1.6</target>
        </configuration>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-eclipse-plugin</artifactId>
        <version>2.9</version>
        <configuration>
          <downloadSources>true</downloadSources>
          <downloadJavadocs>true</downloadJavadocs>
        </configuration>
      </plugin>
    </plugins>
  </build>

</project>
```



Standards in English

Latest Technical, Safety
& Design Specs. Buy
Now!

3. Spring Configuration, Annotation and XML

Here, we show you two ways to configure Spring data and connect to MongoDB, via annotation and XML schema.

Note

Refer to this official reference Connecting to MongoDB with Spring (<http://static.springsource.org/spring-data/data-mongodb/docs/current/reference/html/mongo.core.html#mongodb-connectors>).

3.1 Annotation

Extends the `AbstractMongoConfiguration` is the fastest way, it helps to configure everything you need to start, like `mongoTemplate` object.

SpringMongoConfig.java

```
package com.mkyong.config;

import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.data.mongodb.config.AbstractMongoConfiguration;

import com.mongodb.Mongo;
import com.mongodb.MongoClient;

@Configuration
public class SpringMongoConfig extends AbstractMongoConfiguration {

    @Override
    public String getDatabaseName() {
        return "yourdb";
    }

    @Override
    @Bean
    public Mongo mongo() throws Exception {
        return new MongoClient("127.0.0.1");
    }
}
```

Alternatively, I prefer this one, more flexible to configure everything.

SpringMongoConfig1.java

```

package com.mkyong.config;

import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.data.mongodb.MongoDbFactory;
import org.springframework.data.mongodb.core.MongoTemplate;
import org.springframework.data.mongodb.core.SimpleMongoDbFactory;

import com.mongodb.MongoClient;

@Configuration
public class SpringMongoConfig1 {

    public @Bean
    MongoDbFactory mongoDbFactory() throws Exception {
        return new SimpleMongoDbFactory(new MongoClient(), "yourdb");
    }

    public @Bean
    MongoTemplate mongoTemplate() throws Exception {

        MongoTemplate mongoTemplate = new MongoTemplate(mongoDbFactory());

        return mongoTemplate;
    }

}

```

And load it with AnnotationConfigApplicationContext:

```

ApplicationContext ctx = new AnnotationConfigApplicationContext(SpringMongoConfig.class);
MongoOperations mongoOperation = (MongoOperations)ctx.getBean("mongoTemplate");

```

3.2 XML Schema

SpringConfig.xml

```

<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:mongo="http://www.springframework.org/schema/data/mongo"
    xsi:schemaLocation="http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context-3.0.xsd
        http://www.springframework.org/schema/data/mongo
        http://www.springframework.org/schema/data/mongo/spring-mongo-1.0.xsd
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">

    <mongo:mongo host="127.0.0.1" port="27017" />
    <mongo:db-factory dbname="yourdb" />

    <bean id="mongoTemplate" class="org.springframework.data.mongodb.core.MongoTemplate">
        <constructor-arg name="mongoDbFactory" ref="mongoDbFactory" />
    </bean>

</beans>

```

And include it with Spring's GenericXmlApplicationContext:

```

ApplicationContext ctx = new GenericXmlApplicationContext("SpringConfig.xml");
MongoOperations mongoOperation = (MongoOperations)ctx.getBean("mongoTemplate");

```

So, XML or Annotation?

Actually, both are doing the same thing, it's just based on personal preferences. Personally, I like XML to configure things.

4. User Model

An User object, annotated @Document – which collection to save. Later, we show you how to use Spring data to bind this object to / from MongoDB.

User.java

```
package com.mkyong.model;

import org.springframework.data.annotation.Id;
import org.springframework.data.mongodb.core.mapping.Document;

@Document(collection = "users")
public class User {

    @Id
    private String id;

    String username;

    String password;

    //getter, setter, toString, Constructors
}
```

5. Demo – CRUD Operations

Full example to show you how to use Spring data to perform CRUD operations in MongoDB. The Spring data APIs are quite clean and should be self-explanatory.

App.java

```
package com.mkyong.core;

import java.util.List;

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 org.springframework.data.mongodb.core.query.Update;

import com.mkyong.config.SpringMongoConfig;
import com.mkyong.model.User;
//import org.springframework.context.support.GenericXmlApplicationContext;

public class App {

    public static void main(String[] args) {

        // For XML
        //ApplicationContext ctx = new GenericXmlApplicationContext("SpringConfig.xml");

        // For Annotation
        ApplicationContext ctx =
            new AnnotationConfigApplicationContext(SpringMongoConfig.class);
        MongoOperations mongoOperation = (MongoOperations) ctx.getBean("mongoTemplate");

        User user = new User("mkyong", "password123");

        // save
        mongoOperation.save(user);

        // now user object got the created id.
        System.out.println("1. user : " + user);

        // query to search user
        Query searchUserQuery = new Query(Criteria.where("username").is("mkyong"));

        // find the saved user again.
        User savedUser = mongoOperation.findOne(searchUserQuery, User.class);
        System.out.println("2. find - savedUser : " + savedUser);

        // update password
        mongoOperation.updateFirst(searchUserQuery,
            Update.update("password", "new password"), User.class);

        // find the updated user object
        User updatedUser = mongoOperation.findOne(searchUserQuery, User.class);

        System.out.println("3. updatedUser : " + updatedUser);

        // delete
        mongoOperation.remove(searchUserQuery, User.class);

        // List, it should be empty now.
        List<User> listUser = mongoOperation.findAll(User.class);
        System.out.println("4. Number of user = " + listUser.size());

    }

}
```

Output

```
1. user : User [id=516627653004953049d9ddf0, username=mkyong, password=password123]
2. find - savedUser : User [id=516627653004953049d9ddf0, username=mkyong, password=password123]
3. updatedUser : User [id=516627653004953049d9ddf0, username=mkyong, password=new password]
4. Number of user = 0
```

Download Source Code

Download it – SpringMongoDB-HelloWorld-Example.zip (<http://www.mkyong.com/wp-content/uploads/2011/05/SpringMongoDB-HelloWorld-Example.zip>) (24 KB)

References

1. Spring data for MongoDB (<http://www.springsource.org/spring-data/mongodb>)
2. Connecting to MongoDB with Spring (<http://static.springsource.org/spring-data/data-mongodb/docs/current/reference/html/mongo.core.html#mongodb-connectors>)
3. O'Reilly Spring data mongodb tutorial (http://ofps.oreilly.com/titles/9781449323950/id2318796_07-mongodb.html)
4. Another good Spring data mongodb tutorial (<http://krams915.blogspot.com/2011/02/spring-data-mongodb-tutorial.html>)

Tags : [hello world](http://www.mkyong.com/tag/hello-world/) (<http://www.mkyong.com/tag/hello-world/>) [mongodb](http://www.mkyong.com/tag/mongodb/) (<http://www.mkyong.com/tag/mongodb/>) [spring-data](http://www.mkyong.com/tag/spring-data/) (<http://www.mkyong.com/tag/spring-data/>)

Share this article on

Twitter ([https://twitter.com/intent/tweet?text=Spring Data MongoDB hello world example&url=http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/&via=mkyong](https://twitter.com/intent/tweet?text=Spring+Data+MongoDB+hello+world+example&url=http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/&via=mkyong)) Facebook (<https://www.facebook.com/sharer/sharer.php?u=http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/>) Google+ (<https://plus.google.com/share?url=http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/>)

ARE YOU AS TECH SAVVY AS YOU THINK?

Who collects every public tweet ever sent? **15**

▼

A Donald Trump

B Bill Gates

C Library of Congress

D The British Parliament

Get mortgage help.
This is why.

MAKING HOME AFFORDABLE.gov
888-995-HOPE

Get mortgage help

MAKING HOME AFFORDABLE.gov
888-995-HOPE

About the Author



mkyong

Founder of Mkyong.com (<http://mkyong.com>), love Java and open source stuff. Follow him on Twitter (<https://twitter.com/mkyong>), or befriend him on Facebook (<http://www.facebook.com/java.tutorial>) or Google Plus (<https://plus.google.com/110948163568945735692?rel=author>). If you like my tutorials, consider make a donation to these charities (<http://www.mkyong.com/blog/donate-to-charity/>).

Related Posts

- | | |
|-----|---|
| 12k | Due to limitations of the BasicDBObject, you can't add a second 'Sand' (/java/due-to-limitations-of-the-basicdbobject-you-cant-add-a-second-and/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=0) |
| 76k | Spring Data MongoDB : Insert document (/mongodb/spring-data-mongodb-insert-document/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=1) |
| 71k | MongoDB hello world example (/mongodb/mongodb-hello-world-example/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=2) |
| 47k | Spring Data MongoDB : Save binary file, GridFS example (/mongodb/spring-data-mongodb-save-binary-file-gridfs-example/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=3) |
| 24k | Spring Data MongoDB remove _class column (/mongodb/spring-data-mongodb-remove-_class-column/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=4) |

38k	Spring Data MongoDB : get last modified records (date sorting) (/mongodb/spring-data-mongodb-get-last-modified-records-date-sorting/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=5)
25k	Spring Data MongoDB - Select fields to return (/mongodb/spring-data-mongodb-select-fields-to-return/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=6)
58k	Spring Data MongoDB - Aggregation Grouping Example (/mongodb/spring-data-mongodb-aggregation-grouping-example/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=7)
63k	Spring Boot + Spring Data MongoDB example (/spring-boot/spring-boot-spring-data-mongodb-example/?utm_source=mkyong&utm_medium=author&utm_campaign=related-post&utm_content=8)

Popular Posts

731k	Spring Security Custom Login Form Example (/spring-security/spring-security-form-login-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=1)
1.2m	JAXB hello world example (/java/jaxb-hello-world-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=2)
1.9m	How to convert Java object to / from JSON (Jackson) (/java/how-to-convert-java-object-to-from-json-jackson/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=3)
616k	Java - Read a file from resources folder (/java/java-read-a-file-from-resources-folder/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=4)
1.2m	Spring Security Tutorial (/tutorials/spring-security-tutorials/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=5)
1.3m	How to write to file in Java - BufferedWriter (/java/how-to-write-to-file-in-java-bufferedwriter-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=6)
538k	Top 5 free Java ebooks (/featured/top-5-free-java-ebooks/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=7)
509k	SunCertPathBuilderException: unable to find valid certification path to requested target (/webservicess/jax-ws/suncertpathbuilderexception-unable-to-find-valid-certification-path-to-requested-target/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=8)
729k	How to compare dates in Java (/java/how-to-compare-dates-in-java/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=9)
751k	how to convert Byte[] array to String in Java (/java/how-do-convert-byte-array-to-string-in-java/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=10)
569k	How to determine a prime number in Java (/java/how-to-determine-a-prime-number-in-java/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=11)
684k	Maven + Spring hello world example (/spring/quick-start-maven-spring-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=12)
1.4m	JAX-RS Tutorial (/tutorials/jax-rs-tutorials/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=13)
1.2m	Android spinner (drop down list) example (/android/android-spinner-drop-down-list-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=14)
897k	Struts 2 Tutorial (/tutorials/struts-2-tutorials/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=15)
995k	Java Date and Calendar examples (/java/java-date-and-calendar-examples/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=16)
758k	How to read XML file in Java - (SAX Parser) (/java/how-to-read-xml-file-in-java-sax-parser/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=17)
632k	How to write to file in Java - FileOutputStream (/java/how-to-write-to-file-in-java-fileoutputstream-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=18)
532k	Java MongoDB Tutorial (/tutorials/java-mongodb-tutorials/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=19)
516k	JAX-WS : wsimport tool example (/webservicess/jax-ws/jax-ws-wsimport-tool-example/?utm_source=mkyong&utm_medium=author&utm_campaign=top-pv&utm_content=20)

Leave a Reply



Join the discussion

Sort by: newest | oldest | [most voted](#)



Flat R

gratitude to Mkyong – The only working Spring-MongoDB example I could find.

Guest

0 [REPLY](#)

🕒 8 months 18 days ago



User

Hi do you have an example with jndi mongodb with java config? I would like to keep mongo db details outside project in tomcat.

Guest

0 [REPLY](#)

🕒 1 year 3 months ago ^



User

forgot to mention I want spring java rest application where I am using mongodb.

Guest

0 [REPLY](#)

🕒 1 year 3 months ago



Carol

nice!

Guest

0 [REPLY](#)

🕒 1 year 6 months ago



Charlie (<http://www.leosmartapp.com/>)

Thanks for this!

One thing I noticed, there is a typo in the xml configuration.

The host and port are not wired into the mongoTemplate.

It's only an issue if your mongo db is not running locally as the template uses the default host and port for mongo.

Guest

0 [REPLY](#)

🕒 1 year 8 months ago



Shrivats Rajagopalan

Could you please add a sample for finding distinct values?

Guest

0 [REPLY](#)

🕒 1 year 8 months ago



Ved Prakash Mishra

hello mkyong,

would you please share spring data with spring mvc application ?

Guest

0 [REPLY](#)

🕒 1 year 10 months ago



Asvin Kumar

Is it possible to handle concurrency? Throw an exception when multiple users trying to update same?

Guest

0 [REPLY](#)

🕒 1 year 11 months ago



Maxi Wu

this is really confusing, SpringMongoConfig1.java do not specify any connection parameter, should at least put "127.0.0.1" to clarify thing.

Guest



🕒 2 years 7 days ago



Maxi Wu

This somehow seems anti pattern to me. Since one of the most important feature using MongoDB is to free from predefined schema. We should be able to CRUD like JSONObject to MongoDB without any POJO class, right?

Guest



🕒 2 years 20 days ago



rookie

How can we enable SSL connection to a MONGO DB through Spring, can you please explain Mr. Mkyong

Guest



🕒 2 years 2 months ago



Hari

how do I make this connection singleton? I'm seeing tons of connections from my app to the mongo db and they never close.

Guest



🕒 2 years 3 months ago ^



Hari

found the issue. Do not load the mongoConfig with AnnotationConfigApplicationContext as shown in the example. Instead use @Autowired to let spring inject it for you. AnnotationConfigApplicationContext creates a new virtual container within and hence it keeps creating a new connection to mongodb which leaves lots of mongodb connections lying around. Autowiring solves this problem by reusing the connections. Good luck!

```
@Autowired
```

```
private MongoDBConfig mongoConfig;
```

```
public MongoApplication mongoAppData (String appName) {
```

```
try {
```

```
    MongoOperations mongoOperation = mongoConfig.mongoTemplate();
```

```
    logger.info(appName);
```

```
    Query searchAppQuery = new Query(Criteria.where("canonical_name").is(appName));
```

```
    MongoApplication newapp = mongoOperation.findOne(searchAppQuery, MongoApplication.class);
```

```
    return newapp;
```

```
} catch (Exception e) {
```

```
    logger.error("Error retrieving app data");
```

```
    return null;
```

```
}
```

```
}
```



🕒 2 years 3 months ago



Rajendra Prasad K

When we validate SpringConfig.xml, it is throwing errors by the validation, and it is stoping me to proceed further.

Guest



🕒 3 years 3 months ago ^



Madan

Disable XML validation in your eclipse

Guest

<http://stackoverflow.com/questions/7489510/disable-xml-validation-in-eclipse>
<http://stackoverflow.com/questions/7489510/disable-xml-validation-in-eclipse>



2 years 11 months ago



Guest

Gulshan

HTTP Status 500 – Servlet.init() for servlet dispatcher threw exception

type Exception report

message Servlet.init() for servlet dispatcher threw exception

description The server encountered an internal error that prevented it from fulfilling this request.

exception

```
javax.servlet.ServletException: Servlet.init() for servlet dispatcher threw exception
org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:472)
org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:99)
org.apache.catalina.valves.AccessLogValve.invoke(AccessLogValve.java:929)
org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:407)
org.apache.coyote.http11.AbstractHttp11Processor.process(AbstractHttp11Processor.java:1002)
org.apache.coyote.http11.AbstractHttp11Processor.process(AbstractHttp11Processor.java:585)
org.apache.coyote.tomcat.util.net.NioEndpoint$SocketProcessor.run(NioEndpoint.java:312)
java.util.concurrent.ThreadPoolExecutor$Worker.runTask(ThreadPoolExecutor.java:885)
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:907)
java.lang.Thread.run(Thread.java:619)
```

root cause

```
java.lang.NoClassDefFoundError: com/mongodb/ReadPreference
java.lang.Class.getDeclaredMethods0(Native Method)
java.lang.Class.privateGetDeclaredMethods(Class.java:2427)
java.lang.Class.getDeclaredMethod(Class.java:1935)
org.springframework.core.LocalVariableTableParameterNameDiscoverer$LocalVariableTableVisitor.resolveMember(LocalVariableTableParameterNameDiscoverer.java:236)
org.springframework.core.LocalVariableTableParameterNameDiscoverer$LocalVariableTableVisitor.visitEnd(LocalVariableTableParameterNameDiscoverer.java:219)
org.springframework.asm.ClassReader.accept(Unknown Source)
org.springframework.asm.ClassReader.accept(Unknown Source)
org.springframework.core.LocalVariableTableParameterNameDiscoverer.inspectClass(LocalVariableTableParameterNameDiscoverer.java:114)
org.springframework.core.LocalVariableTableParameterNameDiscoverer.getParameterNames(LocalVariableTableParameterNameDiscoverer.java:86)
org.springframework.beans.factory.support.ConstructorResolver.autowireConstructor(ConstructorResolver.java:193)
org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.autowireConstructor(AbstractAutowireCapableBeanFactory.java:1035)
org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.createBeanInstance(AbstractAutowireCapableBeanFactory.java:939)
org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.doCreateBean(AbstractAutowireCapableBeanFactory.java:485)
org.springframework.beans.factory.support.AbstractAutowireCapableBeanFactory.createBean(AbstractAutowireCapableBeanFactory.java:456)
org.springframework.beans.factory.support.AbstractBeanFactory$1.getObject(AbstractBeanFactory.java:294)
org.springframework.beans.factory.support.DefaultSingletonBeanRegistry.getSingleton(DefaultSingletonBeanRegistry.java:225)
org.springframework.beans.factory.support.AbstractBeanFactory.doGetBean(AbstractBeanFactory.java:291)
org.springframework.beans.factory.support.AbstractBeanFactory.getBean(AbstractBeanFactory.java:193)
org.springframework.beans.factory.support.DefaultListableBeanFactory.preInstantiateSingletons(DefaultListableBeanFactory.java:585)
org.springframework.context.support.AbstractApplicationContext.finishBeanFactoryInitialization(AbstractApplicationContext.java:913)
org.springframework.context.support.AbstractApplicationContext.refresh(AbstractApplicationContext.java:464)
org.springframework.web.servlet.FrameworkServlet.configureAndRefreshWebApplicationContext(FrameworkServlet.java:631)
org.springframework.web.servlet.FrameworkServlet.createWebApplicationContext(FrameworkServlet.java:588)
org.springframework.web.servlet.FrameworkServlet.createWebApplicationContext(FrameworkServlet.java:645)
org.springframework.web.servlet.FrameworkServlet.initWebApplicationContext(FrameworkServlet.java:508)
org.springframework.web.servlet.FrameworkServlet.initServletBean(FrameworkServlet.java:449)
org.springframework.web.servlet.HttpServletBean.init(HttpServletBean.java:133)
javax.servlet.GenericServlet.init(GenericServlet.java:160)
org.apache.catalina.authenticator.AuthenticatorBase.invoke(AuthenticatorBase.java:472)
org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:99)
```

```
org.apache.catalina.valves.AccessLogValve.invoke(AccessLogValve.java:929)
org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:407)
org.apache.coyote.http11.AbstractHttp11Processor.process(AbstractHttp11Processor.java:1002)
org.apache.coyote.AbstractProtocol$AbstractConnectionHandler.process(AbstractProtocol.java:585)
org.apache.tomcat.util.net.JIoEndpoint$SocketProcessor.run(JIoEndpoint.java:312)
java.util.concurrent.ThreadPoolExecutor$Worker.runTask(ThreadPoolExecutor.java:885)
java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:907)
java.lang.Thread.run(Thread.java:619)
```

note The full stack trace of the root cause is available in the Apache Tomcat/7.0.30 logs.

Apache Tomcat/7.0.30



4 years 10 days ago



spring data mongodb 1.0.2.RELEASE | Just do

it. (<http://dev118.com/archives/461>)

[...] Helloworld? <http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/> (<http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/>) [...]



4 years 15 days ago



Veer

Hello,

Guest

I am new to MongoDB.

I am getting Error -> The method `getBean(String)` is undefined for the type `ApplicationContext`. for the following

```
MongoOperations mongoOperation = (MongoOperations) ctx.getBean("mongoTemplate");
```

Please let me know which version of JAR do I need to use?



4 years 1 month ago



Greg

I get `org.springframework.dao.InvalidDataAccessApiUsageException`. Isn't this a relic of Spring 2?

Guest



4 years 5 months ago



Maxim

seems this is invalid:

there is no `'mongoDbFactory'`

Guest



4 years 6 months ago



JJ Zabkar

FYI, the link in section 3 is broken ("http").

Guest



4 years 7 months ago



mkyong (<http://www.mkyong.com>)

Fixed, thanks.

(<http://www.mkyong.com/author/mkyong/>)

Author



4 years 7 months ago



Java MongoDB Tutorial

(<http://www.mkyong.com/tutorials/java-mongodb-tutorials/>)

framework. [...]

[...] Spring Data MongoDB hello world example Configure (both XML and annotation) and perform CRUD operations with "Spring Data for MongoDB"



🕒 4 years 8 months ago



Spring Data MongoDB : Save binary file,
GridFS example

(<http://www.mkyong.com/mongodb/spring-data-mongodb-save-binary-file-gridfs-example/>) [...] Spring Data MongoDB Hello World Example [...]



🕒 4 years 8 months ago



Spring Data MongoDB : Update document

(<http://www.mkyong.com/mongodb/spring-data-mongodb-update-document/>) [...] Spring Data MongoDB Hello World Example [...]



🕒 4 years 9 months ago



Spring Data MongoDB : Insert document

(<http://www.mkyong.com/mongodb/spring-data-mongodb-insert-document/>) [...] Spring Data MongoDB Hello World Example [...]



🕒 4 years 9 months ago



Haris

Guest

Thank you for the sample,

It is simple and well understandable.



🕒 4 years 10 months ago



Guest

Srinivasa HP (<http://www.mkyong.com/java/jce-encryption-data-encryption-standard-des-tutorial/>)

Pls let us know how to close the mongoDB connection.
FYI – we are using Spring Data to connect to mongoDB



🕒 5 years 14 days ago ^



mkyong (<http://www.mkyong.com>)

If not mistake, SpringData will handle it, no need to close or release the MongoDB connection manually.
(<http://www.mkyong.com/author/mkyong/>)

Author



🕒 4 years 8 months ago



Guest

Hans Herrera Bravo

(<http://www.facebook.com/hans.herrerabravo>) persist data? with implementation DataNucleus?



🕒 5 years 19 days ago



Guest

Vincent

Can you show us how to close the mongoDB connection in this example ?



🕒 5 years 1 month ago



Guest

Bob Marks

The Spring XML config isn't quite correct – its missing a "mongo-ref" attribute. For example, if your Mongo database is on another server.
i.e.

It won't connect (always defaults to localhost) so you have to update the

XML to look like the following: –

Nice example all the same!

0

REPLY

5 years 2 months ago



spring data mongodb 1.0.2.RELEASE | Just do it. (<http://coder365.com/archives/1244>) [...] Helloworld?<http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/> (<http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/>) [...]

0

REPLY

5 years 4 months ago



Converting to MongoDB » Agile Mobile Developer
(<http://www.agilemobiledeveloper.com/2012/06/10/converting-to-mongodb/>) [...] <http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/> (<http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/>) [...]

0

REPLY

5 years 6 months ago



mark

what about updating multiple fields at once? I can't seem to find an example of this anywhere with spring data.

Guest

0

REPLY

5 years 8 months ago ^



mkyong (<http://www.mkyong.com>)

Refer to this Spring Data MongoDB : Update Document (<http://www.mkyong.com/mongodb/spring-data-mongodb-update-document/>) (<http://www.mkyong.com/author/mkyong/>)

Author

0

REPLY

4 years 8 months ago

Load More Comments (<http://www.mkyong.com/mongodb/spring-data-mongodb-hello-world-example/#!parentId=82236>)



Хватит сидеть дома



Насладитесь жизнью, приходите играть, кричать и смеяться. Активная Игра в реальности +6



Token with monthly income



Make the crypto trading great again.
Proven efficiency based on a real financial model.



Mkyong
55,036 likes

Like Page

Sign Up

Be the first of your friends to like this



Авторские кожаные сумки



Я шью оригинальные сумки и портфели из кожи. На заказ от 2 дней. Также есть готовые вещи.

Медленный сайт

убивает

SEO!

A woman in a white tank top with a black swirl design, black shorts, and black boots is holding a large black rifle. She is standing next to a black mannequin lying on the ground. The mannequin has 'SEO' written on its back. A white arrow points from the text 'убивает SEO!' to the mannequin.

WSR ускорит сайт

в 5 раз уже сегодня!

Спасти SEO