当前位置：[讨论区](http://www.oschina.net/question) » [技术分享](http://www.oschina.net/question?catalog=2)

窗体顶端



窗体底端

[红薯](http://my.oschina.net/javayou)

**[使用 Spring Data & MongoDB](http://www.oschina.net/question/12_26681)**

[红薯](http://my.oschina.net/javayou) 发表于 8-29 22:01 1个月前, [7](http://www.oschina.net/question/12_26681#answers)回/580阅, 最后回答: 1个月前 (*14*人收藏 ,[收藏](http://www.oschina.net/question/12_26681) |[举报](javascript:report('http://www.oschina.net/question/12_26681',26681,2)))

[**讨论区**](http://www.oschina.net/question)**»** [**技术分享**](http://www.oschina.net/question?catalog=2)

[顶](http://www.oschina.net/question/12_26681) 0 [踩](http://www.oschina.net/question/12_26681)

**这篇文章包含以下三个方面内容：**

1. 在 Windows 下安装 MongoDB
2. MongoDB Java 示例
3. Spring Data & MongoDB.

**首先，安装 MongoDB**

从官方网站上下载最新的 MongoDB ，地址：[MongoDB website](http://www.mongodb.org/downloads).

解压到任一目录，例如：“D:\mongodb-win32-x86\_64-1.8.3“.

MongoDB 并不会自动创建数据目录，我们需要手工创建，例如：“D:\mongodb-data“.

**mongod.exe** 是用来启动 Mongodb 的命令，使用下面的命令行参数来启动：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |
| --- | --- |
| 1 | D:\mongodb-win32-x86\_64-1.8.3\bin>mongod --dbpath "D:\mymongodb-data" |

程序运行后会有一些输出信息，告诉你 MongoDB 已经启动并启用了 Web 管理端口 28017.

你可以通过这个 URL 地址来访问 Web 管理端口：**http://your-ipaddress:28017.**

命令行方式访问 MongoDB 的方法是：D:\mongodb-win32-x86\_64-1.8.3\bin>mongo

如果你可以看到 Web 管理界面，并且可以通过 mongo.exe 连接到服务器，那么说明安装已经成功。

**紧接着，我们编写简单的 Java 示例程序**

需要用到的环境

1. MongoDB 1.8.3
2. [MongoDB-Java-Driver 2.6.5](https://github.com/mongodb/mongo-java-driver/downloads)
3. JDK 1.6

代码如下：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |  |
| --- | --- | --- |
| 01 | package edu.seua.mongo; | |
| 02 |  |

|  |  |  |
| --- | --- | --- |
| 03 | import java.net.UnknownHostException; | |
| 04 |  |

|  |  |  |
| --- | --- | --- |
| 05 | import com.mongodb.BasicDBObject; | |
| 06 | import com.mongodb.DB; |

|  |  |  |
| --- | --- | --- |
| 07 | import com.mongodb.DBCollection; | |
| 08 | import com.mongodb.DBCursor; |

|  |  |
| --- | --- |
| 09 | import com.mongodb.Mongo; |
| 10 | import com.mongodb.MongoException; | |

|  |  |
| --- | --- |
| 11 |  |
| 12 | /\*\* | |

|  |  |  |
| --- | --- | --- |
| 13 | \* Java MongoDB Example | |
| 14 | \* |

|  |  |
| --- | --- |
| 15 | \*/ |
| 16 | public class MongoApp { | |

|  |  |
| --- | --- |
| 17 |  |
| 18 | public static void main(String[] args) { | |

|  |  |
| --- | --- |
| 19 |  |
| 20 | try { | |

|  |  |
| --- | --- |
| 21 | // connect to mongoDB |
| 22 | Mongo mongo = new Mongo("localhost", 27017); | |

|  |  |
| --- | --- |
| 23 |  |
| 24 | // if database doesn't exists, mongoDB will create it | |

|  |  |  |
| --- | --- | --- |
| 25 | DB db = mongo.getDB("mydb"); | |
| 26 |  |

|  |  |
| --- | --- |
| 27 | // Get collection from MongoDB, database named "mydb" |
| 28 | // if collection doesn't exists, mongoDB will create it | |

|  |  |  |
| --- | --- | --- |
| 29 | DBCollection collection = db.getCollection("myCollection"); | |
| 30 |  |

|  |  |
| --- | --- |
| 31 | // create a document to store attributes |
| 32 | BasicDBObject document = new BasicDBObject(); | |

|  |  |
| --- | --- |
| 33 | document.put("id", 100); |
| 34 | document.put("name", "simple name"); | |

|  |  |  |
| --- | --- | --- |
| 35 | document.put("message", "simple message"); | |
| 36 |  |

|  |  |
| --- | --- |
| 37 |  |
| 38 | // save it into collection named "myCollection" | |

|  |  |  |
| --- | --- | --- |
| 39 | collection.insert(document); | |
| 40 |  |

|  |  |
| --- | --- |
| 41 | // search query |
| 42 | BasicDBObject searchQuery = new BasicDBObject(); | |

|  |  |  |
| --- | --- | --- |
| 43 | searchQuery.put("id", 100); | |
| 44 |  |

|  |  |  |
| --- | --- | --- |
| 45 | DBCursor cursor = collection.find(searchQuery); | |
| 46 |  |

|  |  |  |
| --- | --- | --- |
| 47 | // loop over the cursor and display the result | |
| 48 | while (cursor.hasNext()) { |

|  |  |  |
| --- | --- | --- |
| 49 | System.out.println(cursor.next()); | |
| 50 | } |

|  |  |
| --- | --- |
| 51 |  |
| 52 | } catch (UnknownHostException e) { | |

|  |  |  |
| --- | --- | --- |
| 53 | e.printStackTrace(); | |
| 54 | } catch (MongoException e) { |

|  |  |  |
| --- | --- | --- |
| 55 | e.printStackTrace(); | |
| 56 | } |

|  |  |  |
| --- | --- | --- |
| 57 | } | |
| 58 | } |

输出结构：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |
| --- | --- |
| 1 | { |
| 2 | "\_id" : { "$oid" : "4e5936ca602fbf0c50e0c9d0"} , | |

|  |  |  |
| --- | --- | --- |
| 3 | "id" : 100 , "name" : "simple name" , "message" : "simple message" | |
| 4 | } |

最后是 Spring Data & MongoDB

    很复杂，需要下面这么多的 jar 包:

1. MongoDB 1.8.3
2. JDK 1.6
3. mongo-2.6.5.jar
4. org.springframework.context-3.0.5.RELEASE.jar
5. org.springframework.core-3.0.5.RELEASE.jar
6. org.springframework.beans-3.0.5.RELEASE.jar
7. commons-logging-1.1.1.jar
8. org.springframework.asm-3.0.5.RELEASE.jar
9. spring-data-mongodb-1.0.0.M2.jar
10. slf4j-api-1.6.1.jar
11. slf4j-jdk14-1.6.1.jar
12. spring-dao-2.0.8.jar
13. spring-expression-3.1.0.M2.jar
14. spring-data-commons-core-1.1.0.M1.jar
15. spring-tx-3.1.0.M1.jar
16. spring-data-document-core-1.0.0.M2.jar

配置文件(**mongo-config.xml**)：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |
| --- | --- |
| 01 | <?xml version="1.0" encoding="UTF-8"?> |
| 02 | <beans xmlns="http://www.springframework.org/schema/beans" | |

|  |  |
| --- | --- |
| 03 | xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" |
| 04 | xmlns:context="http://www.springframework.org/schema/context" | |

|  |  |  |
| --- | --- | --- |
| 05 | | xmlns:mongo="http://www.springframework.org/schema/data/mongo" |
| 06 | xsi:schemaLocation="http://www.springframework.org/schema/context | |

|  |  |  |  |
| --- | --- | --- | --- |
| 07 | http://www.springframework.org/schema/context/spring-context-3.0.xsd | | |
| 08 | | http://www.springframework.org/schema/data/mongo |

|  |  |  |  |
| --- | --- | --- | --- |
| 09 | http://www.springframework.org/schema/data/mongo/spring-mongo-1.0.xsd | | |
| 10 | | http://www.springframework.org/schema/beans |

|  |  |  |  |
| --- | --- | --- | --- |
| 11 | http://www.springframework.org/schema/beans/spring-beans-3.0.xsd"> | | |
| 12 | |  |

|  |  |  |
| --- | --- | --- |
| 13 | <mongo:mongo host="localhost" port="27017" /> | |
| 14 |  |

|  |  |  |
| --- | --- | --- |
| 15 | <bean id="mongoTemplate" class="org.springframework.data.document.mongodb.MongoTemplate"> | |
| 16 | <constructor-arg ref="mongo" /> |

|  |  |
| --- | --- |
| 17 | <constructor-arg name="databaseName" value="myNewdb" /> |
| 18 | <constructor-arg name="defaultCollectionName" value="myNewCollection" /> |

|  |  |  |
| --- | --- | --- |
| 19 | </bean> | |
| 20 |  |

|  |  |  |
| --- | --- | --- |
| 21 | <!-- To translate any MongoExceptions thrown in @Repository annotated classes --> | |
| 22 | <context:annotation-config /> |

|  |  |
| --- | --- |
| 23 |  |
| 24 | </beans> | |

Employee DTO:

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |  |
| --- | --- | --- |
| 01 | package edu.seua.mongo; | |
| 02 |  |

|  |  |  |
| --- | --- | --- |
| 03 | public class Employee { | |
| 04 |  |

|  |  |
| --- | --- |
| 05 | private String id; |
| 06 | private String firstname; | |

|  |  |  |
| --- | --- | --- |
| 07 | private String lastname; | |
| 08 | private int age; |

|  |  |
| --- | --- |
| 09 |  |
| 10 | public Employee() { | |

|  |  |  |
| --- | --- | --- |
| 11 | } | |
| 12 |  |

|  |  |  |
| --- | --- | --- |
| 13 | public Employee(String id, String firstname, String lastname, int age) { | |
| 14 | super(); |

|  |  |
| --- | --- |
| 15 | this.id = id; |
| 16 | this.firstname = firstname; | |

|  |  |  |
| --- | --- | --- |
| 17 | this.lastname = lastname; | |
| 18 | this.age = age; |

|  |  |  |
| --- | --- | --- |
| 19 | } | |
| 20 |  |

|  |  |  |
| --- | --- | --- |
| 21 | public String getId() { | |
| 22 | return id; |

|  |  |  |
| --- | --- | --- |
| 23 | } | |
| 24 |  |

|  |  |  |
| --- | --- | --- |
| 25 | public void setId(String id) { | |
| 26 | this.id = id; |

|  |  |  |
| --- | --- | --- |
| 27 | } | |
| 28 |  |

|  |  |  |
| --- | --- | --- |
| 29 | public String getFirstname() { | |
| 30 | return firstname; |

|  |  |  |
| --- | --- | --- |
| 31 | } | |
| 32 |  |

|  |  |  |
| --- | --- | --- |
| 33 | public void setFirstname(String firstname) { | |
| 34 | this.firstname = firstname; |

|  |  |  |
| --- | --- | --- |
| 35 | } | |
| 36 |  |

|  |  |  |
| --- | --- | --- |
| 37 | public String getLastname() { | |
| 38 | return lastname; |

|  |  |  |
| --- | --- | --- |
| 39 | } | |
| 40 |  |

|  |  |  |
| --- | --- | --- |
| 41 | public void setLastname(String lastname) { | |
| 42 | this.lastname = lastname; |

|  |  |  |
| --- | --- | --- |
| 43 | } | |
| 44 |  |

|  |  |  |
| --- | --- | --- |
| 45 | public int getAge() { | |
| 46 | return age; |

|  |  |  |
| --- | --- | --- |
| 47 | } | |
| 48 |  |

|  |  |  |
| --- | --- | --- |
| 49 | public void setAge(int age) { | |
| 50 | this.age = age; |

|  |  |  |
| --- | --- | --- |
| 51 | } | |
| 52 |  |

|  |  |
| --- | --- |
| 53 | @Override |
| 54 | public String toString() { | |

|  |  |  |
| --- | --- | --- |
| 55 | return "Employee [age=" + age + ", firstname=" + firstname + ", id=" | |
| 56 | + id + ", lastname=" + lastname + "]"; |

|  |  |  |
| --- | --- | --- |
| 57 | } | |
| 58 |  |

|  |  |
| --- | --- |
| 59 | } |

完整示例代码：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |  |
| --- | --- | --- |
| 01 | package edu.seua.mongo; | |
| 02 |  |

|  |  |  |
| --- | --- | --- |
| 03 | import java.util.List; | |
| 04 |  |

|  |  |
| --- | --- |
| 05 | import org.springframework.context.ApplicationContext; |
| 06 | import org.springframework.context.support.ClassPathXmlApplicationContext; | |

|  |  |  |
| --- | --- | --- |
| 07 | import org.springframework.data.document.mongodb.MongoOperations; | |
| 08 | import org.springframework.data.document.mongodb.query.Criteria; |

|  |  |
| --- | --- |
| 09 | import org.springframework.data.document.mongodb.query.Query; |
| 10 | import org.springframework.data.document.mongodb.query.Update; | |

|  |  |
| --- | --- |
| 11 |  |
| 12 | /\*\* | |

|  |  |  |
| --- | --- | --- |
| 13 | \* Java Spring Data & MongoDB Example | |
| 14 | \* |

|  |  |
| --- | --- |
| 15 | \*/ |
| 16 | public class SpringDataMongoApp { | |

|  |  |
| --- | --- |
| 17 |  |
| 18 | public static void main(String[] args) { | |

|  |  |
| --- | --- |
| 19 |  |
| 20 | ApplicationContext ctx = new ClassPathXmlApplicationContext("edu/seua/mongo/mongo-config.xml"); | |

|  |  |
| --- | --- |
| 21 |  |
| 22 | MongoOperations mongoOperation = (MongoOperations) ctx.getBean("mongoTemplate"); | |

|  |  |
| --- | --- |
| 23 |  |
| 24 | Employee employee = new Employee("100", "firstName", "lastName", 23); | |

|  |  |
| --- | --- |
| 25 |  |
| 26 | // save | |

|  |  |  |
| --- | --- | --- |
| 27 | mongoOperation.save("employees", employee); | |
| 28 |  |

|  |  |
| --- | --- |
| 29 | // find |
| 30 | Employee savedEmployee = mongoOperation.findOne("employees", new Query(Criteria.where( | |

|  |  |  |
| --- | --- | --- |
| 31 | "id").is("100")), Employee.class); | |
| 32 |  |

|  |  |  |
| --- | --- | --- |
| 33 | System.out.println("Saved Employee: " + savedEmployee); | |
| 34 |  |

|  |  |
| --- | --- |
| 35 | // update |
| 36 | mongoOperation.updateFirst("employees", new Query(Criteria.where( | |

|  |  |  |
| --- | --- | --- |
| 37 | "firstname").is("firstName")), Update.update("lastname", | |
| 38 | "new lastName")); |

|  |  |
| --- | --- |
| 39 |  |
| 40 | // find | |

|  |  |  |
| --- | --- | --- |
| 41 | Employee updatedEmployee = mongoOperation.findOne("employees", new Query( | |
| 42 | Criteria.where("id").is("100")), Employee.class); |

|  |  |
| --- | --- |
| 43 |  |
| 44 | System.out.println("Updated Employee: " + updatedEmployee); | |

|  |  |
| --- | --- |
| 45 |  |
| 46 | // delete | |

|  |  |  |
| --- | --- | --- |
| 47 | mongoOperation.remove("employees", new Query(Criteria.where("id").is( | |
| 48 | "100")), Employee.class); |

|  |  |
| --- | --- |
| 49 |  |
| 50 | // List | |

|  |  |
| --- | --- |
| 51 | List<Employee> listEmployee = mongoOperation.getCollection("employees", Employee.class); |
| 52 | System.out.println("size of employees = " + listEmployee.size()); |

|  |  |
| --- | --- |
| 53 |  |
| 54 | } | |

|  |  |
| --- | --- |
| 55 | } |

输出信息：

[view source](http://www.oschina.net/question/12_26681#viewSource)



[print](http://www.oschina.net/question/12_26681#printSource)[?](http://www.oschina.net/question/12_26681#about)

|  |  |
| --- | --- |
| 1 | Saved Employee: Employee [age=23, firstname=firstName, id=100, lastname=lastName] |
| 2 | Updated Employee: Employee [age=23, firstname=firstName, id=100, lastname=new lastName] |

|  |  |
| --- | --- |
| 3 | size of employees = 0 |