

Java + MongoDB hello world example

By [mkyong](#) | May 2, 2011 | Updated : March 8, 2013 | Viewed : 406,589 | +638 pv/w



A simple Java + MongoDB hello world example – how to connect, create database, collection and document, save, update, remove, get and display document (data).

Tools and technologies used :

1. MongoDB 2.2.3
2. MongoDB-Java-Driver 2.10.1
3. JDK 1.6
4. Maven 3.0.3
5. Eclipse 4.2

P.S Maven and Eclipse are both optional, just my personal favorite development tool.

1. Create a Java Project

Create a [simple Java project](#) with Maven.

```
mvn archetype:generate -DgroupId=com.mkyong.core -DartifactId=mongodb
-DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
```

2. Get Mongo Java Driver

Download mongo-java driver from [github](#). For Maven users, declares mongo-java driver in `pom.xml`.

pom.xml

```
<project ...>
  <dependencies>

    <dependency>
      <groupId>org.mongodb</groupId>
      <artifactId>mongo-java-driver</artifactId>
      <version>2.10.1</version>
    </dependency>

  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>2.3.1</version>
        <configuration>
          <source>1.6</source>
          <target>1.6</target>
        </configuration>
      </plugin>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-eclipse-plugin</artifactId>
        <configuration>
          <downloadSources>true</downloadSources>
          <downloadJavadocs>true</downloadJavadocs>
        </configuration>
      </plugin>
    </plugins>
  </build>

</project>
```

3. Mongo Connection

Connect to MongoDB server. For MongoDB version \geq 2.10.0, uses `MongoClient`.

```
// Old version, uses Mongo
Mongo mongo = new Mongo("localhost", 27017);

// Since 2.10.0, uses MongoClient
MongoClient mongo = new MongoClient( "localhost" , 27017 );
```

If MongoDB in secure mode, authentication is required.

```
MongoClient mongoClient = new MongoClient();
DB db = mongoClient.getDB("database name");
boolean auth = db.authenticate("username", "password".toCharArray());
```

4. Mongo Database

Get database. If the database doesn't exist, MongoDB will create it for you.

```
DB db = mongo.getDB("database name");
```

Display all databases.

```
List<String> dbs = mongo.getDatabaseNames();
for(String db : dbs){
    System.out.println(db);
}
```

5. Mongo Collection

Get collection / table.

```
DB db = mongo.getDB("testdb");
DBCollection table = db.getCollection("user");
```

Display all collections from selected database.

```
DB db = mongo.getDB("testdb");
Set<String> tables = db.getCollectionNames();

for(String coll : tables){
    System.out.println(coll);
}
```

Note

In RDBMS, collection is equal to table.

6. Save example

Save a document (data) into a collection (table) named “user”.

```
DBCollection table = db.getCollection("user");
BasicDBObject document = new BasicDBObject();
document.put("name", "mkyong");
document.put("age", 30);
document.put("createdDate", new Date());
table.insert(document);
```

Refer to this [Java MongoDB insert example](#).

7. Update example

Update a document where “name=mkyong”.

```
DBCollection table = db.getCollection("user");

BasicDBObject query = new BasicDBObject();
query.put("name", "mkyong");

BasicDBObject newDocument = new BasicDBObject();
newDocument.put("name", "mkyong-updated");

BasicDBObject updateObj = new BasicDBObject();
updateObj.put("$set", newDocument);

table.update(query, updateObj);
```

[Copy](#)

Refer to this [Java MongoDB update example](#).

8. Find example

Find document where “name=mkyong”, and display it with DBCursor

```
DBCollection table = db.getCollection("user");

BasicDBObject searchQuery = new BasicDBObject();
searchQuery.put("name", "mkyong");

DBCursor cursor = table.find(searchQuery);

while (cursor.hasNext()) {
    System.out.println(cursor.next());
}
```

Refer to this [Java MongoDB search query example](#).

9. Delete example

Find document where "name=mkyong", and delete it.

```
DBCollection table = db.getCollection("user");

BasicDBObject searchQuery = new BasicDBObject();
searchQuery.put("name", "mkyong");

table.remove(searchQuery);
```

Refer to this [Java MongoDB delete example](#).

10. Hello World

Let review a complete Java + MongoDB example, see comments for self-explanatory.

App.java

```
package com.mkyong.core;

import java.net.UnknownHostException;
import java.util.Date;
import com.mongodb.BasicDBObject;
import com.mongodb.DB;
import com.mongodb.DBCollection;
import com.mongodb.DBCursor;
import com.mongodb.MongoClient;
import com.mongodb.MongoException;

/**
 * Java + MongoDB Hello world Example
 *
 */
public class App {
    public static void main(String[] args) {

        try {

            /*** Connect to MongoDB ***/
            // Since 2.10.0, uses MongoClient
            MongoClient mongo = new MongoClient("localhost", 27017);

            /*** Get database ***/
            // if database doesn't exists, MongoDB will create it for you
            DB db = mongo.getDB("testdb");

            /*** Get collection / table from 'testdb' ***/
            // if collection doesn't exists, MongoDB will create it for you
            DBCollection table = db.getCollection("user");

            /*** Insert ***/
            // create a document to store key and value
            BasicDBObject document = new BasicDBObject();
            document.put("name", "mkyong");
            document.put("age", 30);
            document.put("createdDate", new Date());
            table.insert(document);

            /*** Find and display ***/
            BasicDBObject searchQuery = new BasicDBObject();
            searchQuery.put("name", "mkyong");

            DBCursor cursor = table.find(searchQuery);

            while (cursor.hasNext()) {
                System.out.println(cursor.next());
            }

            /*** Update ***/
            // search document where name="mkyong" and update it with new values
            BasicDBObject query = new BasicDBObject();
            query.put("name", "mkyong");

            BasicDBObject newDocument = new BasicDBObject();
            newDocument.put("name", "mkyong-updated");

            BasicDBObject updateObj = new BasicDBObject();
            updateObj.put("$set", newDocument);

            table.update(query, updateObj);

            /*** Find and display ***/
            BasicDBObject searchQuery2
                = new BasicDBObject().append("name", "mkyong-updated");

            DBCursor cursor2 = table.find(searchQuery2);

            while (cursor2.hasNext()) {
                System.out.println(cursor2.next());
            }
        }
    }
}
```

```
    }

    /**** Done ****/
    System.out.println("Done");

    } catch (UnknownHostException e) {
    e.printStackTrace();
    } catch (MongoException e) {
    e.printStackTrace();
    }

    }
}
```

Output...

```
{ "_id" : { "$oid" : "51398e6e30044a944cc23e2e" } , "name" : "mkyong" , "age" : 30 , "createdAt" : { "$date" : "2013-03-08T07:08:30.168Z" } , "
{ "_id" : { "$oid" : "51398e6e30044a944cc23e2e" } , "age" : 30 , "createdAt" : { "$date" : "2013-03-08T07:08:30.168Z" } , "
Done
```

Let use `mongo` console to check the created database “testdb”, collection “user”, and document.

```
$ mongo
MongoDB shell version: 2.2.3
connecting to: test

> show dbs
testdb  0.203125GB

> use testdb
switched to db testdb

> show collections
system.indexes
user
> db.user.find()
{ "_id" : ObjectId("51398e6e30044a944cc23e2e"), "age" : 30, "createdAt" : ISODate("2013-03-08T07:08:30.168Z"), "name" : "mkyong" }
```

Download Source Code

Download it – [Java-mongodb-hello-world-example.zip](#) (13KB)

References

- 1. [Getting started with Java driver](#)
- 2. [Java-MongoDB driver](#)

java

mongodb

About the Author



mkyong

Founder of [Mkyong.com](#), love Java and open source stuff. Follow him on [Twitter](#), or befriend him on [Facebook](#) or [Google Plus](#). If you like my tutorials, consider make a donation to [these charities](#).

Comments

Leave a Reply



Join the discussion

▲ newest ▲ oldest ▲ **most voted**



Shubham Aggarwal

"com.mongodb.DB" is now deprecated.

We can use "MongoDatabase database = MongoClient.getDatabase("myMongoDb");" to obtain the database now but with that, many other methods will change, like to get a collection etc.

+ 4 - Reply

🕒 1 year ago



Malik

That's what I want.

Thank you,

+ 0 - Reply

🕒 8 months ago



Carol

Thanks, good article.

+ 0 - Reply

🕒 9 months ago



Youssef

Why when i do tout in android studio it bloc

+ 0 - Reply

🕒 1 year ago



Dmitry

Thanks mkyong :D

+ 0 - Reply

🕒 1 year ago



Kishan Kumar Verma

hi any bady is know how to automate Mongoddb Quarry

+ 0 - Reply

🕒 1 year ago



roman

you can prepare a simple scenario using Cucumber and your favorite pr.lang (e.g. java or c#)

Guest

+ 0 - Reply

🕒 2 months ago



Ashok

After downloading MongoDriver 2.9.1, 2.10.1 and so on using maven, I cannot find MongoClient. Can you help me with this?

+ 0 - Reply

🕒 2 years ago



Ashok

Never mind.. I got it after series of debugging and internet searches.. Thanks and the articles are really good for a newbie like me

Guest

+ 0 -

Reply

🕒 2 years ago

👍



Brija

Can you please share how did you do it?

Guest

+ 0 -

Reply

🕒 1 year ago



Sankar

Hi I did know how to set the data type as objectId in mongodb using java. Example "accountId" : ObjectId("56c3fc7fbb82eeb4039106b7"). Give solution

Guest

+ 0 -

Reply

🕒 2 years ago

👍



Sankar

It's very urgent

Guest

+ 0 -

Reply

🕒 2 years ago



Suresh

i am trying to insert form data from a jsp page to the mongodb but i am not succeed plz give me any example of jsp+mongodb with form submission please.....

Guest

+ 0 -

Reply

🕒 2 years ago



Amol Jore

Thanks Mkyong.. helped me a lot.

Guest

+ 0 -

Reply

🕒 3 years ago



Dennis Wanjama

I was following this tutorial
this is my error: the method put(String, Object) in the type BasicDBObject is not applicable for the arguments (String, int)
mycode:
BasicDBObject doc = new BasicDBObject();
doc.put("age", 20);

Guest

+ 0 -

Reply

🕒 3 years ago



Dennis Wanjama

hi i was following your tutorial <http://www.mkyong.com/mongodb/java-mongodb-hello-world-example/>
this is my error: the method put(String, Object) in the type BasicDBObject is not applicable for the arguments (String, int)
mycode:
BasicDBObject doc = new BasicDBObject();
doc.put("age", 20);

Guest

+ 0 -

Reply

🕒 3 years ago



Rashmi Venkatesh

Hello, I am getting this error when trying to connect to mongodb server through Java code. I am able to connect and run commands in mongo shell. Please help INFO: Exception in monitor thread while connecting to server xxxxxx:cccc
com.mongodb.MongoSocketReadException: Exception receiving message at
com.mongodb.connection.InternalStreamConnection.translateReadException(InternalStreamConnection.java:480) at
com.mongodb.connection.InternalStreamConnection.receiveMessage(InternalStreamConnection.java:225) at
com.mongodb.connection.CommandHelper.receiveReply(CommandHelper.java:134) at
com.mongodb.connection.CommandHelper.receiveCommandResult(CommandHelper.java:121) at
com.mongodb.connection.CommandHelper.executeCommand(CommandHelper.java:32) at
com.mongodb.connection.InternalStreamConnectionInitializer.initializeConnectionDescription(InternalStreamConnectionInitializer.java:83) at
com.mongodb.connection.InternalStreamConnectionInitializer.initialize(InternalStreamConnectionInitializer.java:43) at
com.mongodb.connection.InternalStreamConnection.open(InternalStreamConnection.java:115) at

Guest

com.mongodb.connection.DefaultServerMonitor\$ServerMonitorRunnable.run(DefaultServerMonitor.java:127) at java.lang.Thread.run(Thread.java:662) Caused by: java.net.SocketException: Connection reset at java.net.SocketInputStream.read(SocketInputStream.java:168) at com.mongodb.connection.SocketStream.read(SocketStream.java:85) at com.mongodb.connection.InternalStreamConnection.receiveResponseBuffers(InternalStreamConnection.java:491) at com.mongodb.connection.InternalStreamConnection.receiveMessage(InternalStreamConnection.java:221) ... 8 more com.mongodb.MongoTimeoutException: Timed out after 30000 ms while waiting for a server that matches ReadPreferenceServerSelector{readPreference=primary}. Client view of cluster state is {type=UNKNOWN,... [Read more »](#)

+

0

–

[Reply](#)

3 years ago



michael

getDB(String) is deprecated. Use getDatabase(String);

+

0

–

[Reply](#)

3 years ago



Scott

When I try to run I am getting:
"Exception in thread "main" java.lang.NoClassDefFoundError: com/mongodb/MongoException"
I simplified the code way down and am still getting it, also ran the original code (attached archive) and got the same thing.
Is this an issue with how I'm linking mongo to the project or something else?

+

0

–

[Reply](#)

3 years ago



Tom.TH Lin

Thanks for the convenient example, it works perfect!!

+

0

–

[Reply](#)

3 years ago



Nerdcoresteve

No matter what I do, I keep getting package com.mongodb does not exist

+

0

–

[Reply](#)

3 years ago



Nerdcoresteve

Ah, never mind. I was putting the dependency in the wrong place. Not too familiar with maven. Thanks for this tutorial. I use your tutorials all the time. :)

+

0

–

[Reply](#)

3 years ago



idir

thanks for your job

+

0

–

[Reply](#)

3 years ago



Shahul

Even after adding the org.mongodb dependency, I am not seeing the mongodb java drivers under the external libraries. Please help

+

0

–

[Reply](#)

3 years ago



Wojciech Bogucki

How do you convert the result to String, can you use toString() method?

+

0

–

[Reply](#)

4 years ago



Niharika Saxena

if table has duplicate clolum value after extecuting below script twice.

Guest

db.users.insert({username:"niharika",password:"123456"})
db.users.insert({username:"niharika",password:"123456"})

db.users.update({username:"niharika"},{\$set:{username:"mkyoung"}})

When we try to updtae the cloumn name it updtae only 1 row. ideally it should update all the rows any suggesstion?


{ "_id" : { "\$oid" : "53b2f692d28717367aa9ce28" } , "username" : "niharika" , "password" : "123456" }
{ "_id" : { "\$oid" : "53b2fd6ed28717367aa9ce29" } , "username" : "mkyoung" , "password" : "123456" }

+ 0 -

Reply

🕒 4 years ago

👍



tan


public WriteResult update(DBObject q , DBObject o , boolean upsert , boolean multi);

+ 0 -

Reply

🕒 3 years ago

👍



Umesh


I am fresher in development .
Really this is nice project/example to beginner..
Thank you very much.....

+ 0 -

Reply

🕒 4 years ago

👍



saurav jain


Also found more elaboration on how to create collections more explicitly i.e capped collections.
<http://lotusmediacentre.com/mongodb-java-hello-world-example/>

+ 0 -

Reply

🕒 4 years ago

👍



Kundan Dere


When we run example code . After successful execution of code ... whey the java code is not terminating ?
how should i set timeout for same example ?

+ 0 -

Reply

🕒 4 years ago

👍



paul


dont you want to close the connections?

+ 0 -

Reply

🕒 4 years ago

👍



kranthi

Hi mkyong,


Thanks for your sharing your knowledge with us.They are really helpful for us

+ 0 -

Reply

🕒 4 years ago

👍



Mediha


Instead of creating a maven project are there any jars that we can import and then import them?

+ 0 -

Reply

🕒 5 years ago

👍



Mediha


Instead of creating a maven project are there any jars that we can import and then use those instead of pom.xml dependencies?

+ 0 -

Reply

🕒 5 years ago

👍



a


Why would you ever want to do that? Maven has made everything so easy.

Guest

+ 0 -

Reply

🕒 4 years ago



Mateen

i just did tried that there are 3 jars listed below that i used, and copied the above sample program it worked just fine


1. mongo-2.10.1.jar
2. mongoconnector.jar
3. mongo-java-driver-2.10.1-javadoc.jar

and you will be good to go

+ 0 -

Reply

🕒 4 years ago



nirmal


Some more examples

http://www.pretechsol.com/p/nosql_20.html#.UeZpTRnZhaE

+ 0 -

Reply

🕒 5 years ago




ajit

For a beginner like me , your tutorials are too good. I appreciate the work that you do to make people understand all the concepts. Thanks a lot.

+ 0 -

Reply

🕒 5 years ago



dreamshutter

Do not need MongoClient.close() in example source?
use only upper code then mongo has too many client connection
then it works slower

+ 0 -

Reply

🕒 5 years ago

Load More Comments

Favorites Links

- [Android Getting Started](#)
- [Google App Engine – Java](#)
- [Spring 2.5.x Documentation](#)
- [Spring 3.2.x Documentation](#)
- [Spring 4.1.x Documentation](#)
- [Java EE 5 Tutorial](#)
- [Java EE 6 Tutorial](#)
- [Java EE 7 Tutorial](#)
- [Java 6 API](#)
- [Java 7 API](#)
- [Java 8 API](#)
- [Oracle J2SE Tutorials](#)
- [JSF Home Page](#)
- [JSP Home Page](#)
- [Maven Central Repository](#)
- [Gradle User Guide](#)
- [Hibernate ORM](#)
- [JAX-WS Home Page](#)
- [JAX-RS Home Page \(Jersey\)](#)

Partners & Bookmarks

- [JAX London](#)
- [Java Code Geeks](#)

About Mkyong.com

Mkyong.com is for Java and J2EE developers, all examples are simple and easy to understand, and well tested in my development environment.

Mkyong.com is created, written by, and maintained by Yong Mook Kim, aka Mkyong. It is built on [WordPress](#), hosted by [Liquid Web](#), and the caches are served by CloudFlare CDN.

