勤务系统(QW)项目组

Web后端方案设计文档

|  |  |
| --- | --- |
| **文档编号** |  |
| **版本号** |  |
| **日期** | 2019-09-11 |

本文档模板由中华人民共和国总后勤保障部勤务系统项目组所有。

本文档未经总后书面许可，任何单位和个人不得以任何形式摘抄、复制本文档的部分或全体，并以任何形式传播。

中华人民共和国后勤保障部

**2018年09月**

**文档版本与控制信息**

|  |  |
| --- | --- |
| **拟制** | 宋全恒 |
| **审核** | XXX |
| **责任人** | XXX |
| **读者** | 勤务系统项目组 |

**文档修改记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **修改日期** | **修改人** | **修改内容简述** | **版本号** |
| 2017-09-11 | 宋全恒 | 新建 | V1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

勤务系统项目环境部署

# 简介

由于

# 材料

勤务系统部署的数据库为maridb数据库和mongo数据库，同时，在勤务系统项目运行时需要一些数据库表的存在，因此需要提供脚本文件sql来通过命令执行，因此所需的材料如0下

mariadb-10.3.8-winx64.msi

mongodb-win32-x86\_64-2008plus-ssl-4.0.1-signed.msi

liucheng2.sql

mongo4idea-0.12.0.zip 插件

cas.zip 单点登陆服务， 解压部署在Tomcat的webapps目录下

# 数据库部署

## 安装mariadb数据库

### 安装maridb数据库

安装mariadb数据库较为简单，因为安装程序为msi，所以安装过程是有向导界面的，在安装时按照默认即可，需要确保在安装时勾选可以通过网络可以访问数据库。

### 配置环境变量

由于是默认安装，安装路径为

MARIADB\_HOME: C:\Program Files\MariaDB 10.3

由于Program Files中间有空格，因此在Path中添加bin目录时如下所示：

%”MARIADB\_HOME“%\bin或者”C:\Program Files“\MariaDB 10.3\bin

### 验证

输入如下命令，如果有输出即表明mariadb环境配置OK。

C:\Users\全恒>mysql -u root -p

Enter password: \*\*\*\*

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 28

Server version: 10.3.8-MariaDB mariadb.org binary distribution

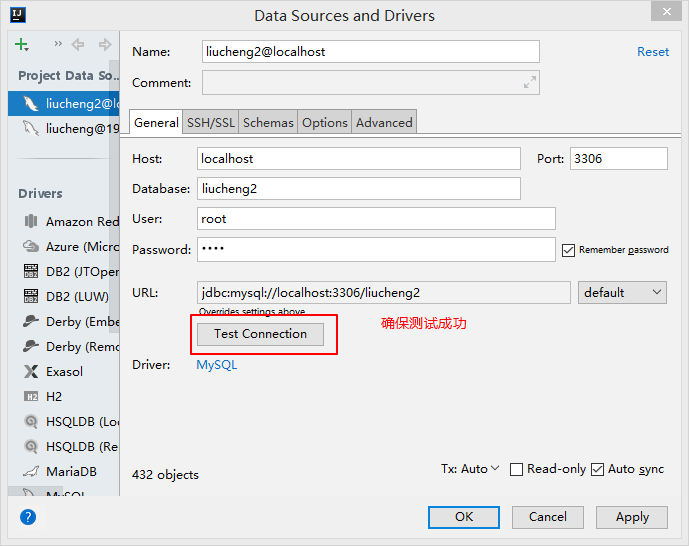
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>

## IDEA连接数据库

虽然数据库使用mariadb，但仍然可以使用mysql的驱动，mysql-connector-java。



连接过后，确保能够连接成功。

## 安装mongodb数据库

### 安装

运行mongodb-win32-x86\_64-2008plus-ssl-4.0.1-signed.msi向导程序，默认安装即可。默认安装位置为：

C:\Program Files\MongoDB

### 配置环境变量

新建系统变量MONGODB\_HOME,值如下：

C:\Program Files\MongoDB\Server\4.0

由于Program Files含有空格，因此在Path中引入bin目录时在末尾添加下值：

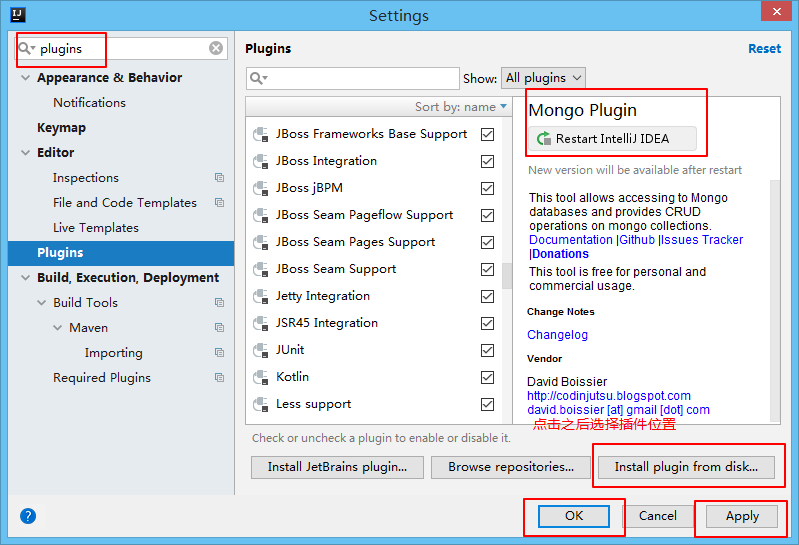
“%MONGODB\_HOME%”\bin

### 安装Intellij IDEA的mongoDB插件

可以在下述网站下载IDEA mongo插件

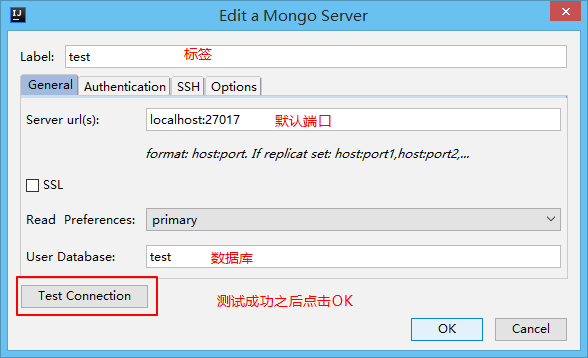
<https://plugins.jetbrains.com/plugin/7141-mongo-plugin>

在IDEA中打开设置，搜索plugins，选择Install from disk，如下图：



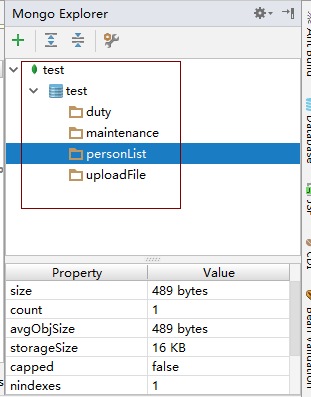
### 使用IDEA连接mongodb数据库

点击Apply->OK之后Restart IDEA，再次启动之后，选择菜单View->Tool Windows->Mongo Explorer，在弹出的Edit a Mongo Server对话框中输入Mongo Server属性



### 验证

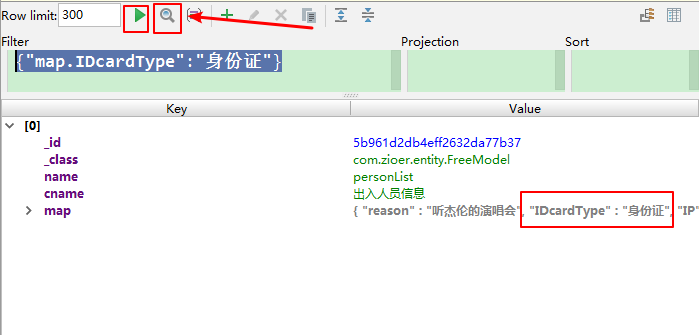
点击OK之后，在右侧弹出如下结果：



双击personList表，点击搜索图标，添加过滤器，输入如下内容：

{"map.IDcardType":"身份证"}

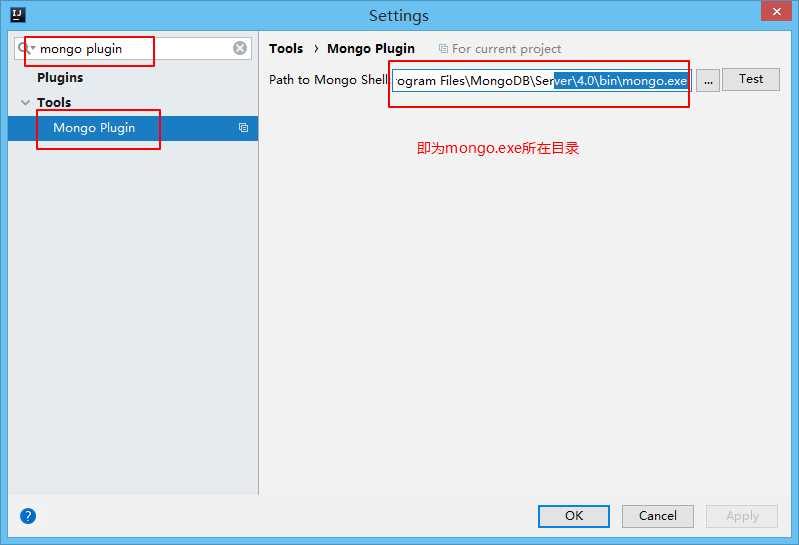
点击运行图标，即可实现过滤。



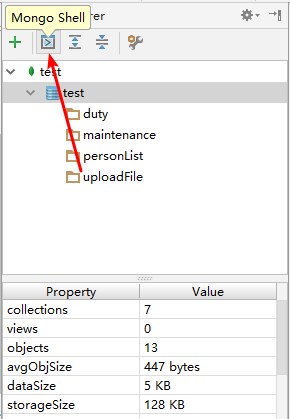
### Mongo Shell

IDEA同时可以结合使用Mongo Shell来操作数据库，只要把在Mongo Plugin中path to Mongo Shell路径配置为mono.exe的路径即可：

C:\Program Files\MongoDB\Server\4.0\bin\mongo.exe



点击右侧Test，通过即可。



点击Mongo Shell 即可运行Mongo命令如下：

> db.personList.find().count()

1

>db.personList.find()

{ "\_id" : ObjectId("5b961d2db4eff2632da77b37"), "\_class"…}

### MongoDB常用命令

help

db.help();

db.yourColl.help();

db.youColl.find().help();

show dbs;

show collections;

use db\_name;

rs.help();

db.userInfo.find();

db.userInfo.distinct("name");

db.userInfo.find({"age": 22});

db.userInfo.find({age: {$gt: 22}});

db.userInfo.find({age: {$gte: 23, $lte: 26}});

升序：db.userInfo.find().sort({age: 1});

降序：db.userInfo.find().sort({age: -1});

db.userInfo.find().limit(5);

db.userInfo.find().skip(10);

mogo导出数据库

mongodump -h dbhost -d dbname -o dbdirectory

常输入以上命令即可。

注意：在Mongo Shell中输入类似du.userInfo.find()的命令之后，点击Ctrl +Enter即可运行。

# 数据库配置文件修改

由于从北京压缩文件而生成的包，里面的数据库配置全部无效，而在自己的电脑安装了MongoDB和mariadb数据库，需要在配置文件中更换成自己的数据库配置。需要修改的文件如下：

## resources/dbconfig.properties

修改成如下内容：

url:jdbc:mysql://localhost:3306/liucheng2?useUnicode=true&characterEncoding=utf8&characterSetResults=utf8&serverTimezone=Hongkong

driverClassName:com.mysql.jdbc.Driver

username:root

password:root

## resources/mongo.properties

修改成如下内容：

mongo.host=127.0.0.1

mongo.port=27017

mongo.connectionsPerHost=8

mongo.threadsAllowedToBlockForConnectionMultiplier=4

#连接超时时间

mongo.connectTimeout=1000

#等待时间

mongo.maxWaitTime=1500

mongo.autoConnectRetry=true

mongo.socketKeepAlive=true

#socket超时时间

mongo.socketTimeout=1500

mongo.slaveOK=true

mongo.writeconcern=safe

由于Mongo服务器的默认端口为27017，沿用即可，但勤务系统把此端口修改为20058，不明所以。

由于在该文件中仅仅指定了Mongodb访问的主机名和端口，而并没有指出要访问的数据库名，这一部分在mongodb-context.xml进行指定。

## resources/mongodb-context.xml

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:mongo="http://www.springframework.org/schema/data/mongo"

xsi:schemaLocation="http://www.springframework.org/schema/data/mongo http://www.springframework.org/schema/data/mongo/spring-mongo.xsd">

<mongo:mongo host="${mongo.host}" port="${mongo.port}"/>

<!-- Offers convenience methods and automatic mapping between MongoDB JSON documents and your domain classes. -->

<bean id="mongoTemplate" class="org.springframework.data.mongodb.core.MongoTemplate">

<constructor-arg ref="mongo"/>

<constructor-arg name="databaseName" value="test"/>

</bean>

</beans>

上述代码片段value即Mongo要访问的数据库名称。test表示在Mongo Server中项目使用的数据库为test数据库。确保该数据库存在。

## resources/spring/ApplicationContext.xml

在该文件中，也需要配置数据库名称为test

<**mongo:mongo host="${mongo.host}" port="${mongo.port}"**/>

<!-- Offers convenience methods and automatic mapping between MongoDB JSON documents and your domain classes. -->

<**bean id="mongoTemplate" class="org.springframework.data.mongodb.core.MongoTemplate"**>

<**constructor-arg ref="mongo"**/>

<**constructor-arg name="databaseName" value="test"**/>

</**bean**>

## shiro.properties

修改成如下：

#localhost  
  
shiro.loginUrl=http://localhost:8080/cas/login?service=http://localhost:8080/fiso/shiro-cas  
shiro.logoutUrl=http://localhost:8080/cas/logout?service=http://localhost:8080/fiso/shiro-cas  
shiro.cas.serverUrlPrefix=http://localhost:8080/cas  
shiro.cas.service=http://localhost:8080/fiso/shiro-cas  
  
shiro.failureUrl=/logoutsys  
shiro.successUrl=/login\_toLogin

# cas服务配置

CAS是Central Authentication Service的缩写，中央认证服务，一种独立开放指令协议。CAS 是 [Yale](https://baike.baidu.com/item/Yale/10754982) 大学发起的一个开源项目，旨在为 Web 应用系统提供一种可靠的单点登录方法，CAS 在 2004 年 12 月正式成为 JA-SIG 的一个项目。CAS单点登录服务具有如下特征：

1、开源的企业级单点登录解决方案。

2、CAS Server 为需要独立部署的 Web 应用。

3、CAS Client 支持非常多的客户端(这里指单点登录系统中的各个 Web 应用)，包括 Java, .Net, PHP, Perl, Apache, uPortal, Ruby 等。

由于cas服务需要访问数据库因此，需要对cas的某些配置文件进行配置

## deployerConfigContext.xml

该文件位于目录：

C:\apache-tomcat-9.0.2\webapps\cas\WEB-INF

对数据源配置进行修改，修改成如下：

<!-- 设置数据源 -->

<bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName" value="com.mysql.jdbc.Driver"></property>

<property name="url" value="jdbc:mysql://localhost:3306/liucheng2?useUnicode=true&amp;characterEncoding=utf8"></property>

<property name="username" value="root"></property>

<property name="password" value="root"></property>

</bean>

# 运行

运行IDEA，启动Restart Server，如果自动跳转到单点登录页面，即部署成功。

注意：如果想要部署的勤务系统可以被局域网内的其他人进行访问，比如说都连接了同一个wifi，这样每个计算机均处于相同的网段，只要把上述所有的localhost修改成自己的IP即可，比如说本次自己的IP为192.168.0.106.

这样部署出来的结果