1.官网下载MySQL（https://www.mysql.com/）

2.解压后以管理员身份打开命令行（win+R输入cmd）

转到mysql的bin目录

安装mysql的服务：mysqld --install

初始化mysql，在这里，初始化会产生一个随机密码：mysqld --initialize --console

开启mysql的服务：net start mysql

登录验证mysql是否安装成功：mysql -u root -p

修改密码

配置MySQL环境变量

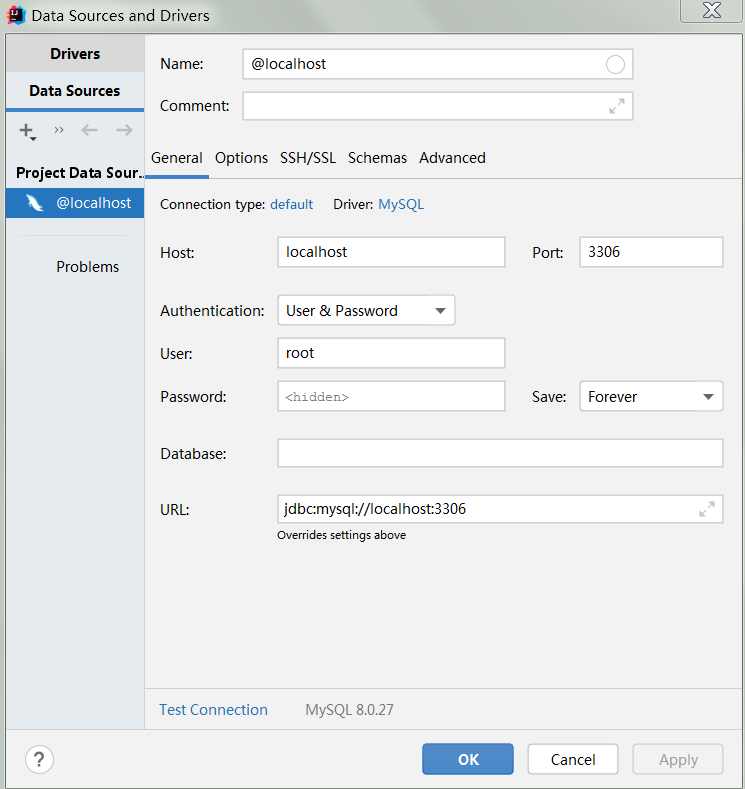
<https://www.cnblogs.com/winton-nfs/p/11524007.html>

3.配置JDBC：

导入jar包：<https://mvnrepository.com/artifact/mysql/mysql-connector-java>

使用依赖导入即可，不必下载jar包

连接数据库：



不用像web课程时一样配置driver

我的密码：123456

4.测试：

import java.sql.Connection;

//Connection接口是JDBC为数据库的连接所提供的一个接口，具有操作数据库和控制数据源的作用

import java.sql.DriverManager;

import java.sql.SQLException;

//DriverManager类管理一组 JDBC驱动程序的基本服务，也有操作数据库和控制数据源的作用

public class Blog {

public static final String url=

"jdbc:mysql://localhost:3306/rctms?useUnicode=true&characterEncoding=utf-8&useSSL=false&serverTimezone = GMT";

// MySQL的JDBC URL编写方式：jdbc:mysql://主机名称：连接端口/数据库的名称?

public static final String user = "root";

//连接数据库的用户名，一般都设置为root

public static final String password = "123456";

//连接数据库的密码

public static void main(String[] args) throws ClassNotFoundException {

Connection con = null;

Class.forName("com.mysql.cj.jdbc.Driver");

//使用Class类动态加载驱动程序

try {

con = DriverManager.getConnection(url, user, password);

System.out.println(con);

System.out.println("连接成功");

con.close();

} catch (SQLException e) {

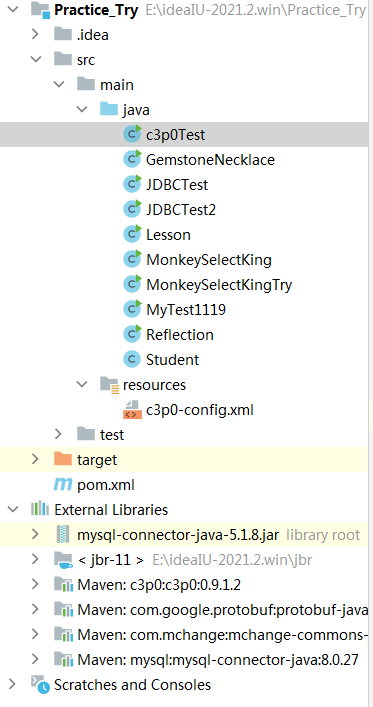
e.printStackTrace();

}

}

}

二、c3p0配置：

在resources下写c3p0-config.xml

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

<c3p0-config>

<!-- 默认配置，如果没有指定则使用这个配置 -->

<default-config>

<property name="user">root</property>

<property name="password">123456</property>

<property name="jdbcUrl">jdbc:mysql://localhost:3306/user?useUnicode=true&amp;characterEncoding=utf-8&amp;useSSL=false&amp;serverTimezone = GMT"</property>

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

<property name="checkoutTimeout">30000</property>

<property name="idleConnectionTestPeriod">30</property>

<property name="initialPoolSize">3</property>

<property name="maxIdleTime">30</property>

<property name="maxPoolSize">100</property>

<property name="minPoolSize">2</property>

<property name="maxStatements">200</property>

</default-config>

<!-- 命名的配置,可以通过方法调用实现 -->

<named-config name="test">

<property name="user">root</property>

<property name="password">123456</property>

<property name="jdbcUrl">jdbc:mysql://localhost:3306/user?useUnicode=true&amp;characterEncoding=utf-8&amp;useSSL=false&amp;serverTimezone = GMT"</property>

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

<!-- 如果池中数据连接不够时一次增长多少个 -->

<property name="acquireIncrement">5</property>

<!-- 初始化数据库连接池时连接的数量 -->

<property name="initialPoolSize">20</property>

<!-- 数据库连接池中的最大的数据库连接数 -->

<property name="maxPoolSize">25</property>

<!-- 数据库连接池中的最小的数据库连接数 -->

<property name="minPoolSize">5</property>

</named-config>

</c3p0-config>

导入的依赖：

<!-- https://mvnrepository.com/artifact/com.mchange/mchange-commons-java -->

<dependency>

<groupId>com.mchange</groupId>

<artifactId>mchange-commons-java</artifactId>

<version>0.2.15</version>

</dependency>

<!-- https://mvnrepository.com/artifact/com.mchange/c3p0 -->

<dependency>

<groupId>c3p0</groupId>

<artifactId>c3p0</artifactId>

<version>0.9.1.2</version>

</dependency>

编写测试类：

import com.mchange.v2.c3p0.ComboPooledDataSource;

import com.mysql.jdbc.Connection;

import javax.sql.DataSource;

/\*\*

\* @date 2021年11月28日 09:33

\*/

public class c3p0Test {

public static void main(String[] args) {

long begin = System.currentTimeMillis();

try {

DataSource ds = new ComboPooledDataSource("test");

Connection conn = (Connection) ds.getConnection();

conn.close();

} catch (Exception e) {

e.printStackTrace();

}

long end = System.currentTimeMillis();

System.out.println("Running time:" + (end - begin));

}

}