首先，在连接数据库之前必须保证SQL Server是采用SQL Server身份验证方式而不是windows身份验证方式。如果在安装时选用了后者，则重新设置如下:

在默认情况下，SQL Server Express是采用集成的Windows安全验证且禁用了sa登录名。为了工作组环境下不使用不方便的Windows集成安全验证，我们要启用SQL Server Express的混合安全验证，也就是说由SQL Server来验证用户而不是由Windows来验证用户。

1、打开SQL Server Management Studio Expre

打开SQL Server，登录界面中身份验证首先选择

Windows 身份验证

，点击连接进入



SQL登录界面

当我们启动SQL Server Management Studio Express时，首先它要连接到我们的SQL Server 2008 Express，我们在安装SQL Server 2008 Express时，默认的实例为SQLExpress，服务器名称的组成为：机器名\实例名，因此，本例的服务名称为MATTJASON\SQLEXPRESS（注：安装SQL Server 2008 Express的机器名为MATTJASON）。第一次使用SQL Server Management Studio Express，由于我们必须采用Windows身份验证，这是默认安装时决定的。

2、设置身份验证

右击

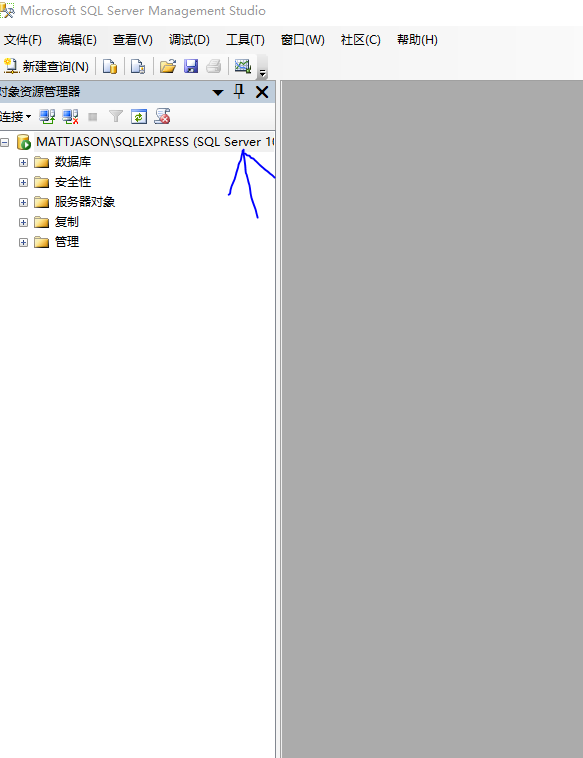
对象资源管理器下的第一行的

MATTJASON\SQLEXPRESS

（视具体机器名而变），

选择属性

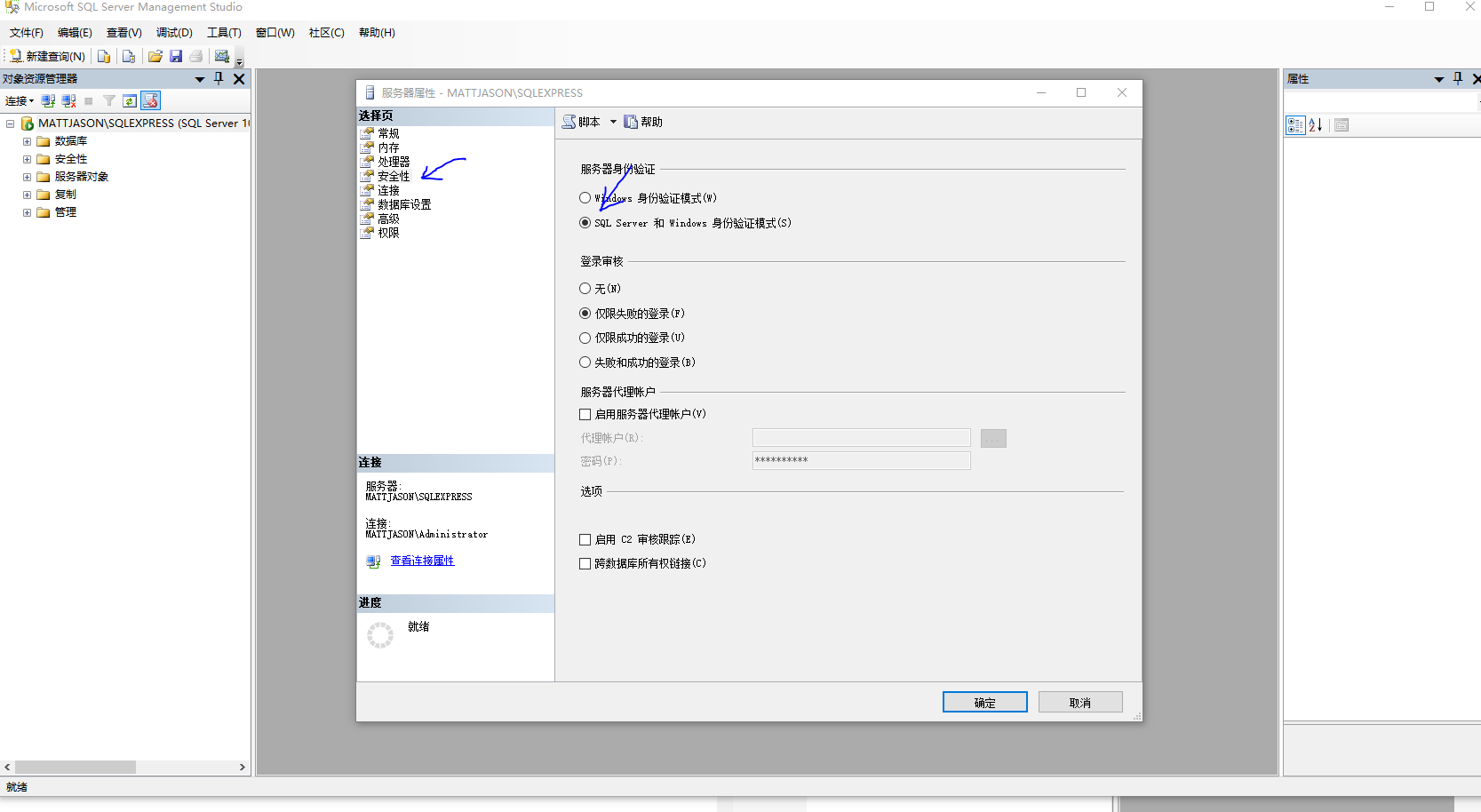
，如下图所示。



属性

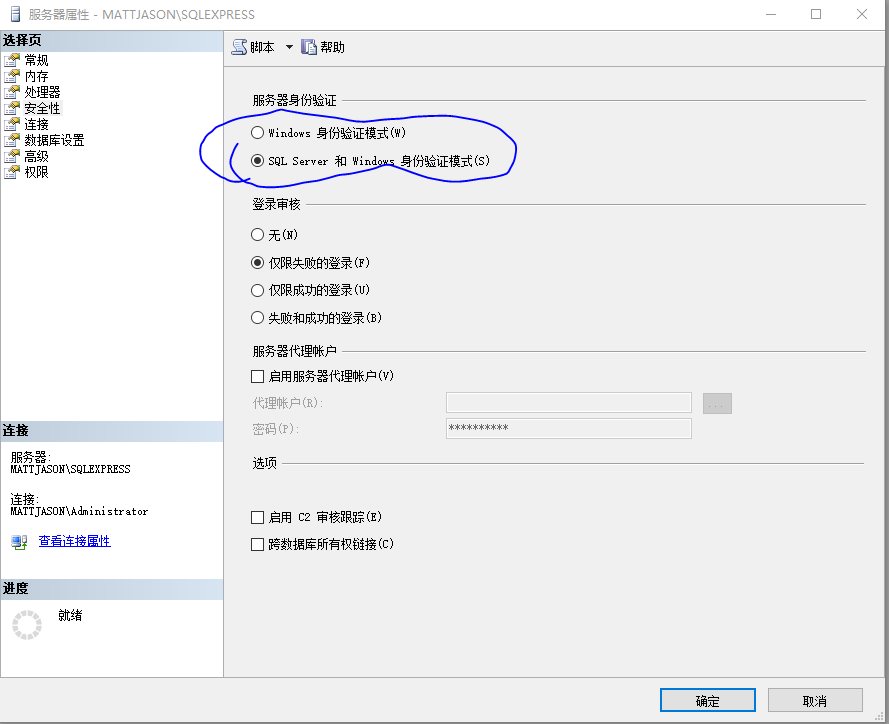
打开后选择

安全性



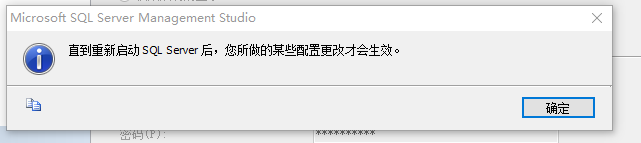
安全性

下面这张图清晰一点



大图

选择好后确认并关闭。



重启

要求重启，

退出程序

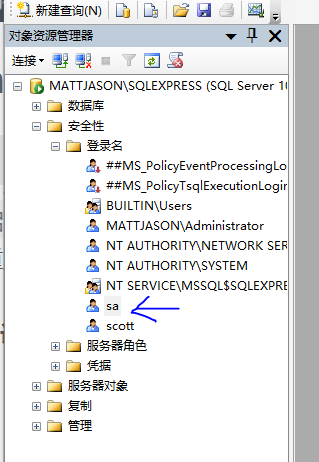
，再次进入。

这一步很重要！

3、 设置sa的密码并启用sa登录名

点击安全性，再点击登录名，看到sa，

右击选择属性



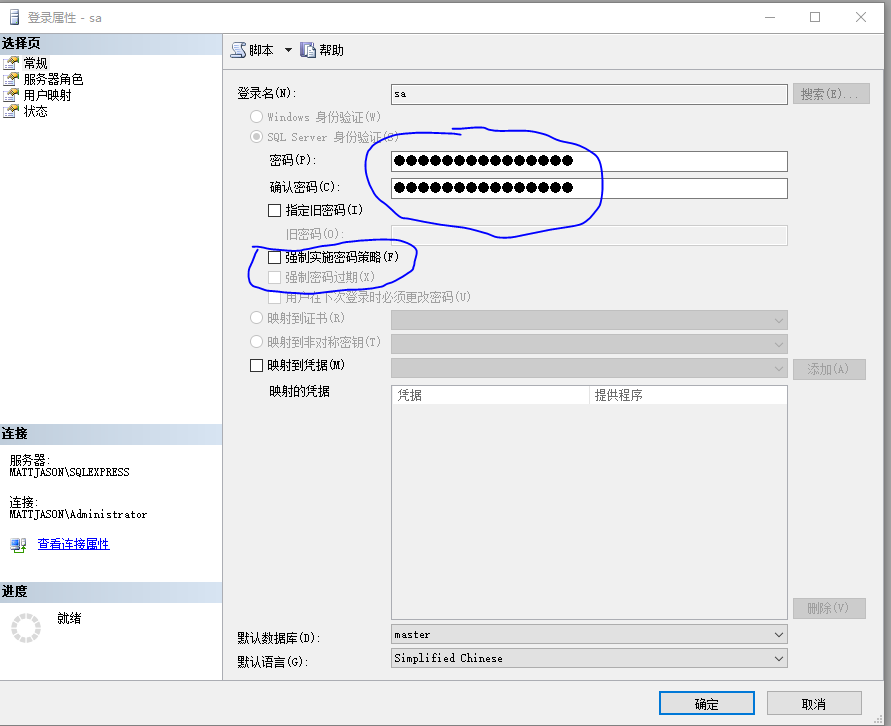
打开后如下图所示。

由于我们不知道sa的密码，所以我们须设置一个！更改密码，可以设置一个简单好记的，我设置的是qwerty。

并且取消

强制实施密码策略

的选项！



a属性

不要关闭这个窗口，接着选择左侧的“

状态

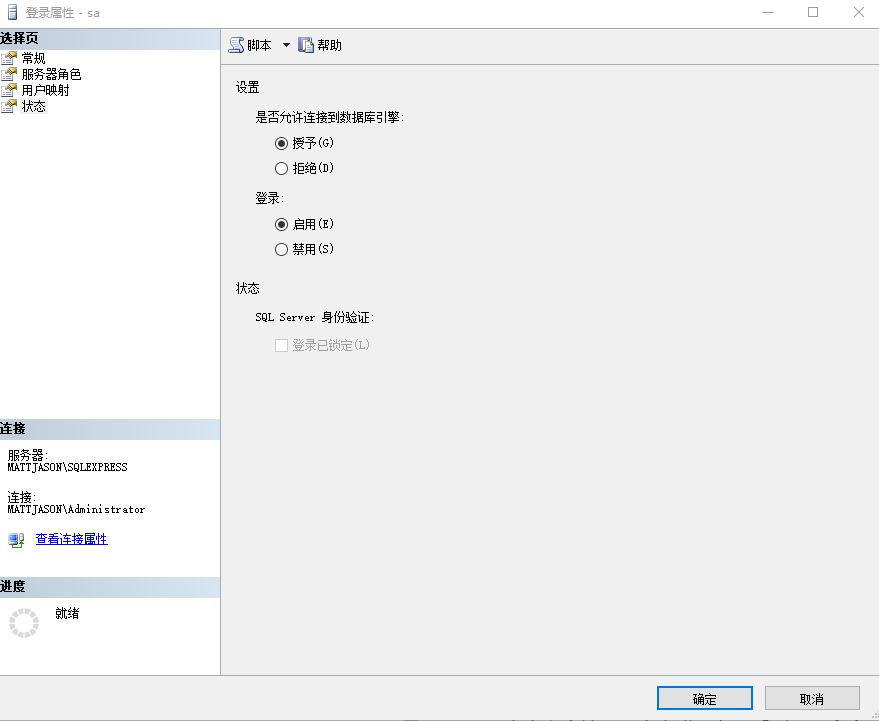
”，默认情况下，sa登录名是禁用的，因此，我们必须启用sa登录名，

选择右侧的登录下的“启用”

并且选择上方的

授予

！如图所示。



登录属性

最后，我们点击窗口下面的“确定”按钮。

好了，到此为止，SQL Server 2008 Express服务器已经可以让sa登录了，不过，

右击MATTJASON\SQLEXPRESS，选择重新启动！很重要！

4、验证sa登录

退出SQL，重新打开，

选择SQLServer身份验证方式

，输入刚才设置的sa密码，连接！



重启

如果可以成功进入，那么就配置成功了，如果进不去，说明刚才有步骤没有设置，需要重新设置一下，不要着急，我当时也是配置好久。。。。

如下配置：

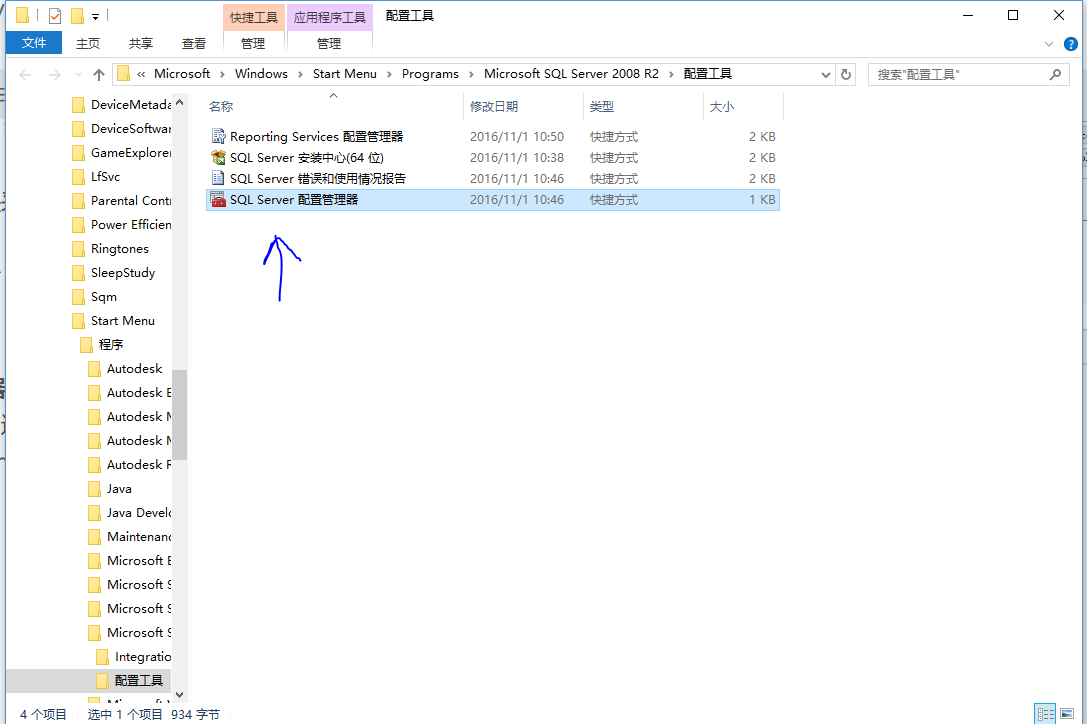
因为SQL Server 2008装好后，默认协议是没有开启的，所以要打开SQL Server配置管理器中开启：

5、SQL Server配置管理器(可选)

安装好SQL Server 2008后，运行 开始 → 所有程序 → Microsoft SQL Server 2018 → 配置工具 →

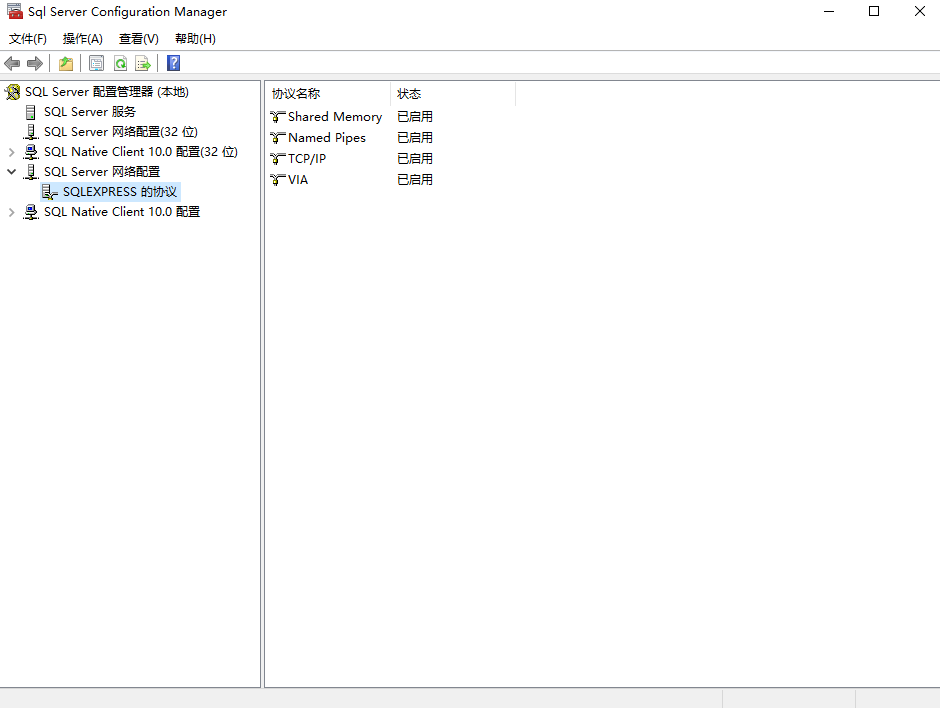
SQL Server配置管理器

，如下图所示：



SQL配置管理器

打开的窗口如下图所示。在左边栏找到 SQL Server网络配置选项，点开它的小箭头，会看到“【你的数据库名】的协议” （图中是SQLEXPRESS的协议），选中它，看右边栏。



Named Pipe

右击启用，同样的，右击

TCP/IP

点击启用！

双击

TCP/IP

（右键→属性），在弹出的窗口中选择 “IP地址” 选项卡.

IP1和IP10

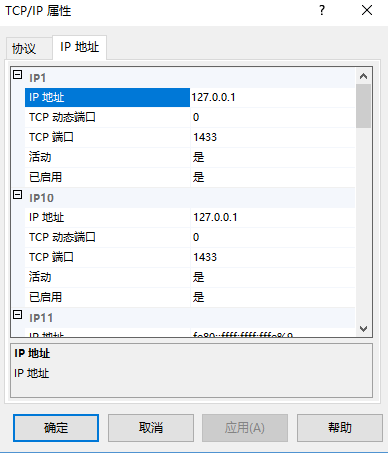
的【IP地址】设为

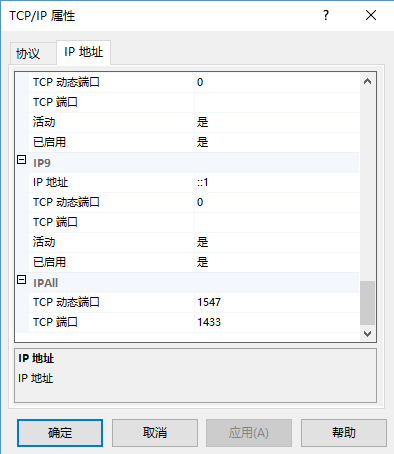
127.0.0.1

，并将所有【IP+数字】的【已启用】设为是。接着，拖动下拉条到最下方，将IP1、IP10、 IPAll 中的【TCP端口】设成

1433

，这是SQL的传输端口，非常重要！





重新启动计算机！

6、打开telnet并测试1433端口(可选)

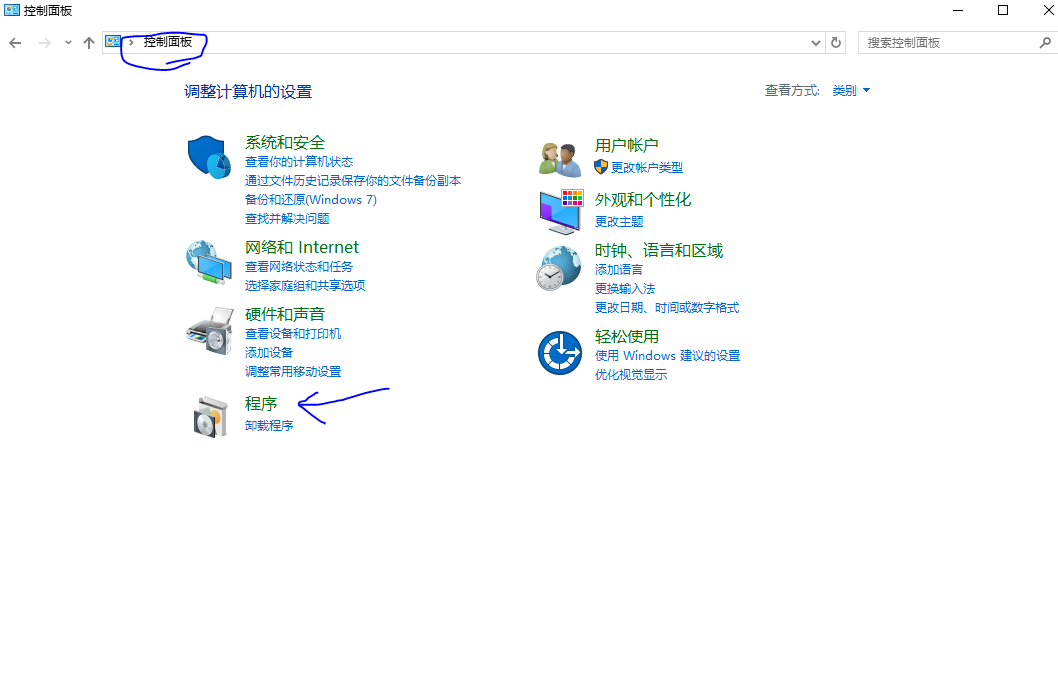
接下来使用telnet命令测试1433端口是否打开。首先要保证telnet服务开启。

打开telnet服务：

依次点击“开始”→“控制面板”→“程序”（W10下直接在文件资源管理器上的搜索框输入

控制面板

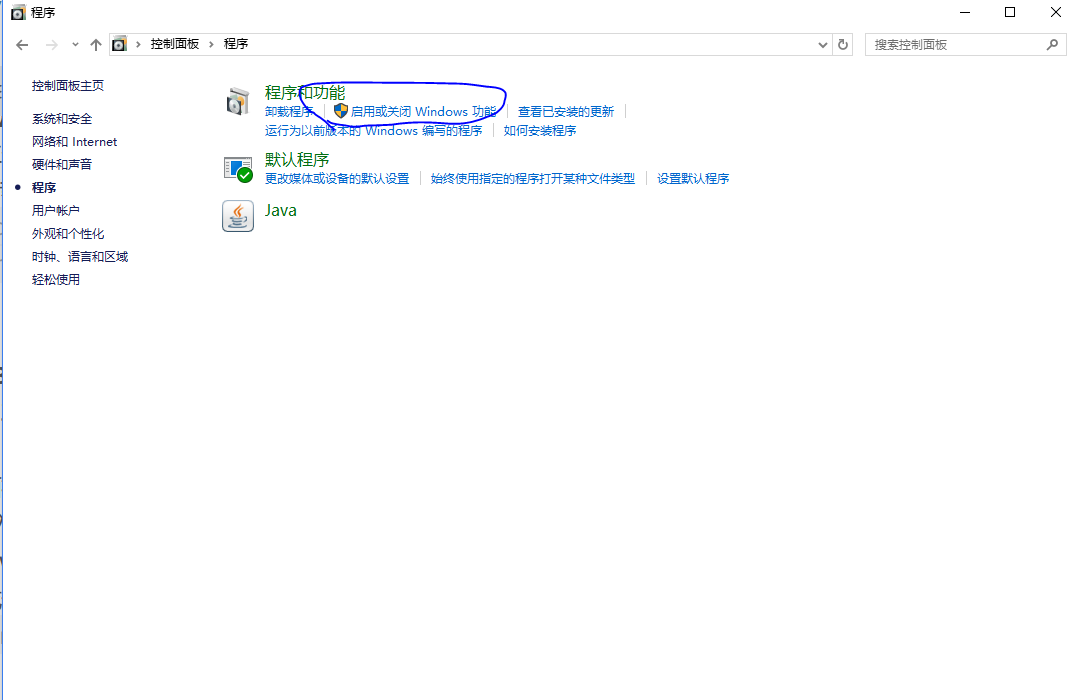
进入），“在程序和功能”找到并点击“打开或关闭Windows 功能”进入Windows 功能设置对话框。找到并勾选“Telnet客户端，最后“确定”稍等片刻即可完成安装。



程序

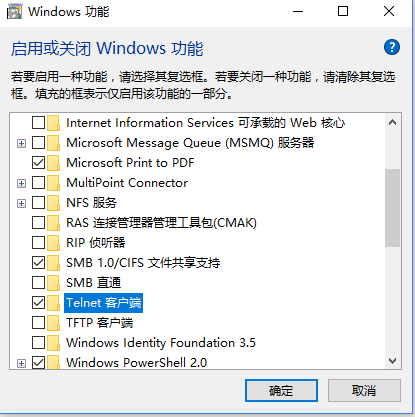
找到

启用或关闭Windows功能



功能

找到Telnet，点选，确定。



安装完成后，Telnet服务默认情况下是禁用的。还需执行“开始”→“运行”（或者按住win键和R键），输入

ervices.msc

打开服务管理器。找到并双击

Telnet服务项

，设置其启动方式为

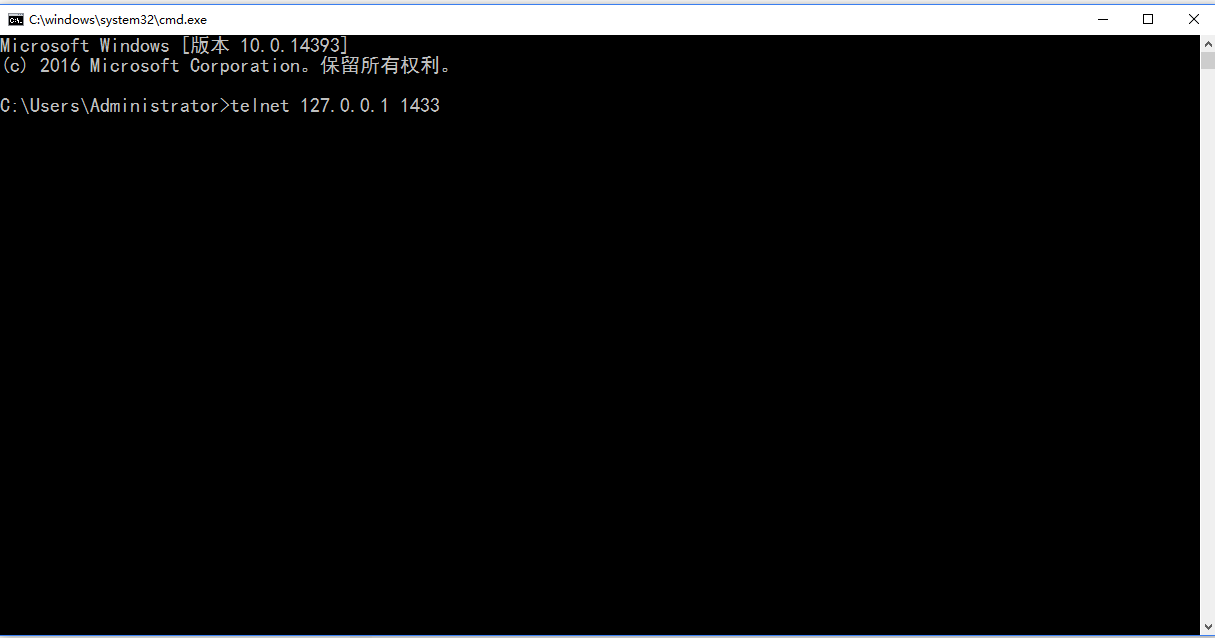
“手动”

(更安全，只在需要的时候才启用)，最后“启动”该服务“确定”退出即可。

完成上一步后。开始菜单 → 运行cmd → 输入：

telnet 127.0.0.1 1433

，（注意telnet与127之间有空格，1与1433之间有空格）



测试是否成功，如果成功则说明配置成功，如果连接失败则说明上面的配置有误，同志仍需努力！

7、下载Microsoft JDBC Driver 4.0 for SQL Server 官方下载地址：http://www.microsoft.com/zh-cn/download/details.aspx?id=11774

下载sqljdbc\_4.0.2206.100\_chs.tar.gz（2.2M），解压文件，得到sqljdbc41.jar和sqljdbc42.jar。如果你使用的是jre1.7版本，则忽略sqljdbc.jar（因为它用不了，而且如果和sqljdbc41.jar一起用会出错），只留下sqljdbc41.jar。

以下设置均针对jre1.7版本（1.7以下应该也适用）：

在D盘新建一个文件夹，命名为sqljdbc4，将sqljdbc41.jar复制一个进去。

8、配置环境变量

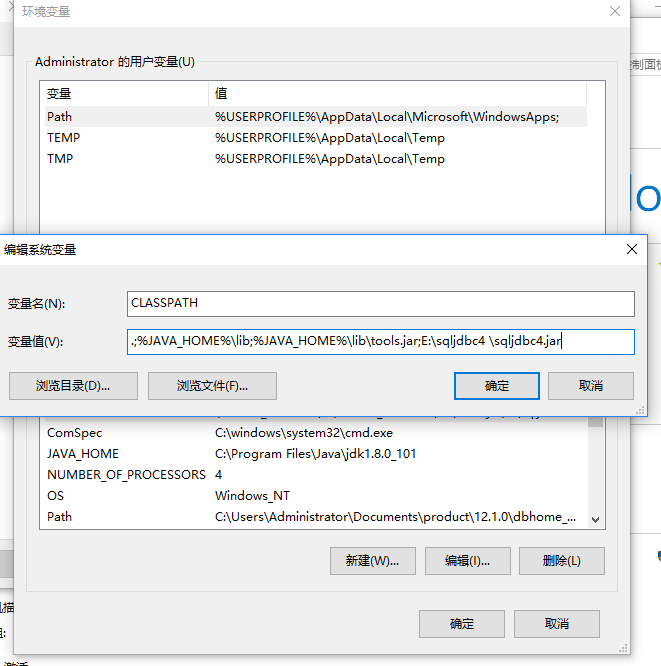
右击 我的电脑 → 属性 → 高级系统设置（高级） → 环境变量，在系统变量中双击CLASSPATH变量（或选中CLASSPATH后 → 编辑），在最后面追加 “

D:\sqljdbc4 \sqljdbc41.jar

” （

注意最前面有个

）若不存在CLASSPATH，就新建CLASSPATH变量，并且将其值设为“D:\sqljdbc4 \sqljdbc41.jar”。



环境变量

连续点击 确定 以退出环境变量配置。

有几个地方需要注意：

（1）我们需要将sqljdbc41.jar类库文件拷贝到E:\Program Files\Java\jdk1.7.0\jre\lib\ext目录下。（看你安装在哪个盘，如果是C盘，则最前面的E改为C，下同）

（2）我们需要将sqljdbc4.jar类库文件拷贝到E:\Program Files\Java\jre7\lib\ext目录下

最好是，只要是jre文件夹，都复制一个sqljdbc41.jar到jre7\lib\ext里去！！

（3）如果是使用Tomcat做服务器（我使用的是Tomcat7)，那么我们需要将sqljdbc4.jar类库文件拷贝到C:\apache-tomcat-7.0.11\lib目录下。

（4）如果是使用Tomcat做服务器，那么我们需要将sqljdbc4.jar类库文件拷贝到E:\apache-tomcat-7.0.11\webapps\gaofei\WEB-INF\lib目录下（gaofei目录是我的应用，这个路径相信你会看明白）

注意，只有sqljdbc4.jar ！！如果把sqljdbc.jar和sqljdbc4.jar都拷在一起的话，这样就算你全都做对了，也会持续出现有“此驱动程序不支持JRE1.7，请使用支持JDBC4.0的sqljdbc4.jar的类库”的问题。因为jdk默认地选择了sqljdbc.jar（前面我已经提到，只留下sqljdbc4.jar）。

9、使用Eclipse测试连接SQL Server 2008数据库

1、打开SQL Server 2008，在其中新建数据库 Test，然后退出SQL Server 2008。

2、运行Eclipse，新建一个Java Project 名为 Test。

3、右单击src，依次选择

Build Path → Configure Build Path

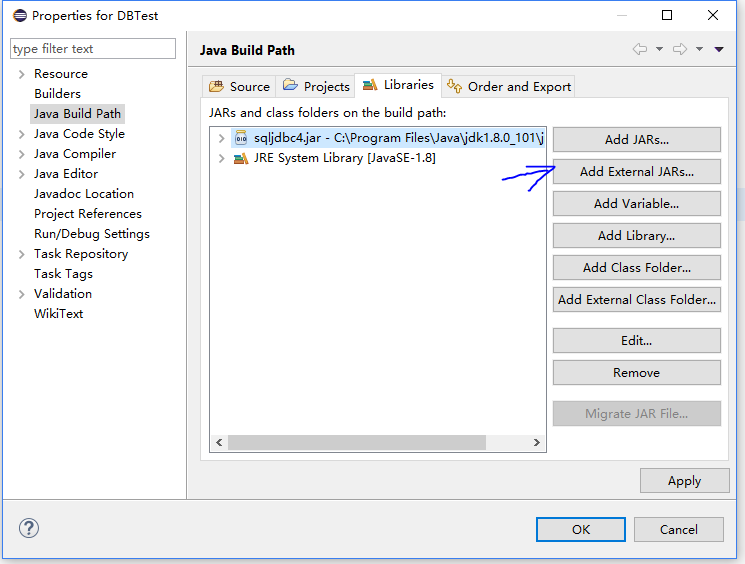
，在打开的窗口的右边选择 Libraries标签，然后单击

Add External JAR

，找到刚才下载的

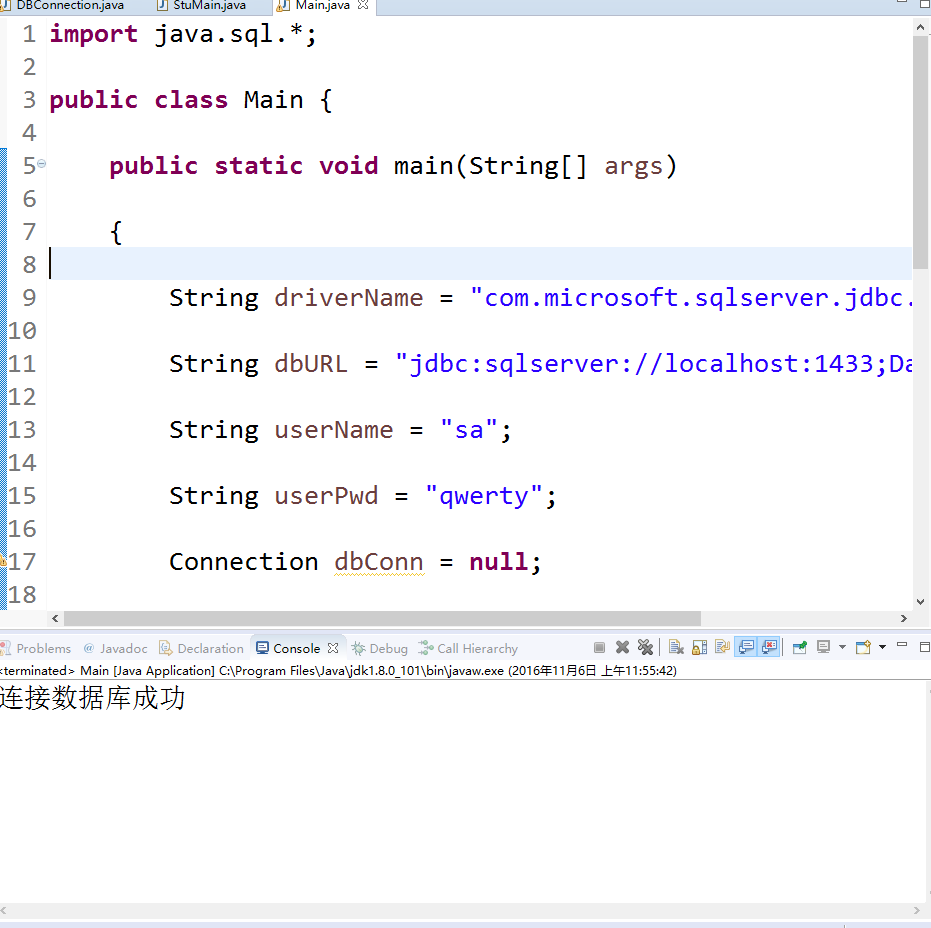
qljdbc4.jar

文件并打开，然后单击 OK 完成构建路径的配置。



接下来进行测试！最后一步啦！ 在Test中新建包pkg，在pkg中新建一个类Main，在其中输入代码如下：

温馨提示：如果要对数据库中的某个表进行操作，需要像这样子做：String sql = “SELECT\* FROM [数据库名].[dbo].[表名] where xxx “; 例如String sql = “SELECT\* FROM [metro].[dbo].[4] wherexxx” 。注意，中括号是必要的，不能去掉。



成功

最后运行，如果出现连接成功就说明大功告成了，可以开始你的数据库开发之旅了！！