



**博客：** <https://www.cnblogs.com/HOsystem/p/14116443.html>

## 2、具体内容

对于现代的开发几乎所有的项目都是围绕着数据库展开的，很少见到没有数据库而单独存在的项目。所以任何一门编程语言如果要想发展，那么必须对数据库的开发有所支持，同样，Java从最初的时代开始就一直支持着数据库的开发标准——JDBC(Java Database Connectivity、Java数据库连接),JDBC本质上来讲并不属于一个技术，它属于一种服务。而所有服务的特征：必须按照指定的套路来进行操作。

在Java里面专门为JDBC提供有一个模块(java.sql),在JDBC里面核心的组成就是DeriverManag类以及若干接口(Connection、Statement、PreparedStatement、ResultSet)。

对于JDBC的程序数据库访问也分为如下四种形式：

·JDBC-ODBC桥连接：利用微软的ODBC技术进行数据库的连接，而后再利用JDBC技术访问ODBC技术进行数据库的开发，这种技术为Java默认支持的技术，不需要做任何额外配置即可实现；

|- 处理流程：程序 -> JDBC -> ODBC -> 数据库，操作性能很差，不会有人用的

·JDBC连接：直接利用JDBC进行数据库的连接处理；

|- 处理流程：程序 -> JDBC -> 数据库，这种连接一般直连接本地数据库服务；

·**JDBC网络连接**：通过特定的网络协议连接指定的数据库服务；

|- 处理流程：程序 -> JDBC -> 网络数据库(IP地址、端口号)；

·JDBC协议连接：自己通过编写指定的协议操作实现数据库的访问；

---



## 2、具体内容

本次将基于Oracle数据库来实现JDBC的编程实现，那么在这样的情况下请一定要保证已经成功的启动了Oracle数据库的两个服务：数据监听服务(Listener)、实例服务(SID)；

OracleOraDb11g_home1TNSListener	已启动	手动	本地系统
OracleServiceMLDN	已启动	手动	本地系统

由于本次将直接利用JDBC进行Oracle数据库的连接，所以必须保证系统之中已经配置了Oracle的驱动程序：

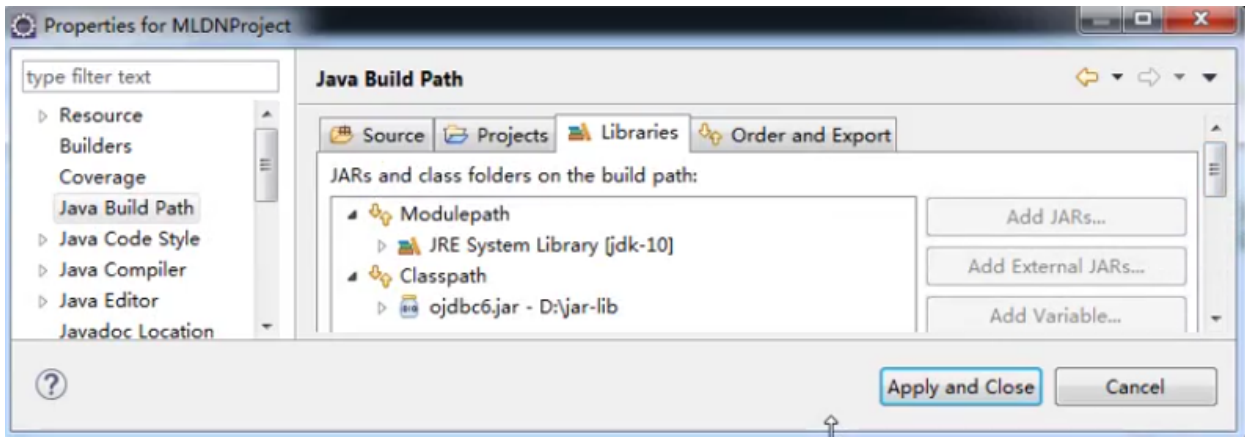
·Oracle驱动程序路径：

D:\app\mldn\product\11.2.0\dbhome\_1\jdbc\lib\ojdbc6.jar;

·**情况一**：如果下砸你使用记事本进行程序编写，则需要将其手工配置到CLASSPATH之中；



·**情况二**：基于eclipse开发，所以需要在JavaBuilderPath上配置此驱动程序。



如果现在要连接Oracle数据库，则必须采用如下的步骤进行处理：

- 通过反射机制加载数据库驱动程序：oracle.jdbc.driver.OracleDriver;

- 数据库的连接需要有一个网络的连接地址，改地址结构如下：

- |- 地址结构：jdbc:oracle:thin:@主机名称:端口号:SID;

- |- MLDN数据库：jdbc:oracle:thin:@localhost:1521:mldn;

- 数据库的用户名：scott;

- 数据库的密码：tiger;

对于数据库的连接提供有java.sql.DriverManager的程序类，利用此类中的方法可以获取一个Connection接口对象；

- 获取Connection接口对象：public static Connection getConnection(String url,String user,String password) throws SQLException;

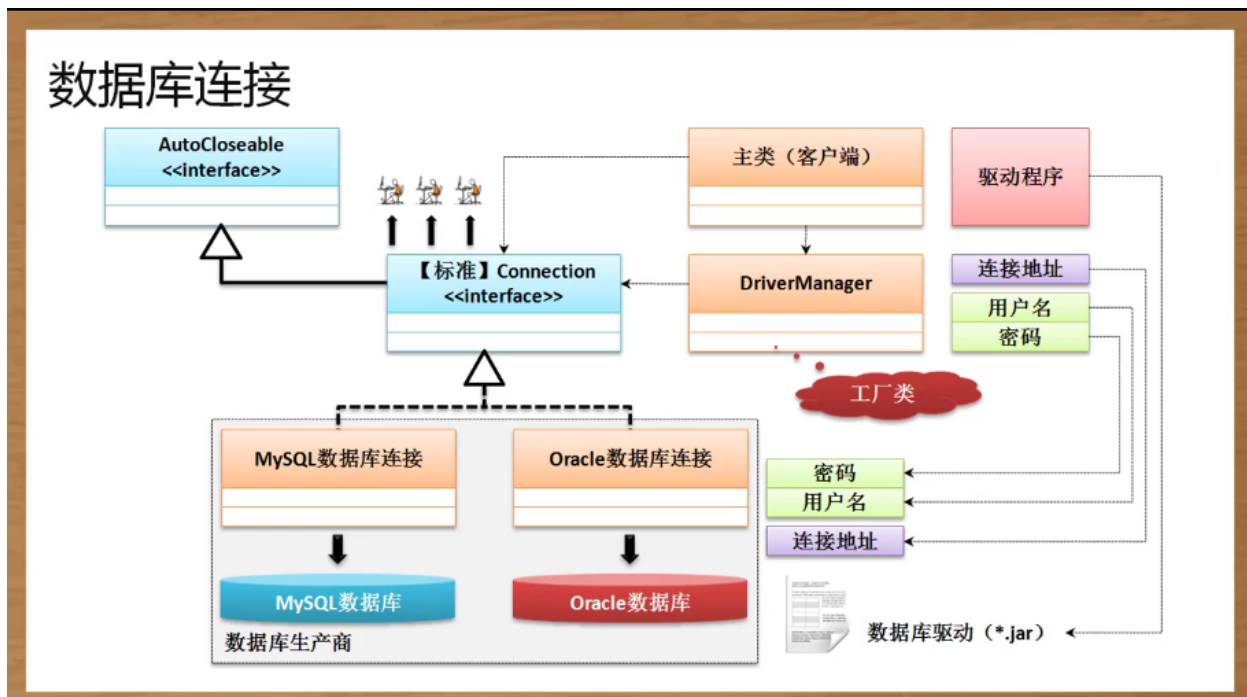
### 范例：实现具体的数据库连接操作

```
package cn.mldn.demo;
import java.sql.Connection;
import java.sql.DriverManager;
public class JDBCdemo {
    private static final String DATABASE_DRIVER = "oracle.jdbc.driver.OracleDriver" ;
    private static final String DATABASE_URL = "jdbc:oracle:thin:@localhost:1521:mldn" ;
    private static final String DATABASE_USER = "scott" ;
    private static final String DATABASE_PASSWORD = "tiger" ;
    public static void main(String[] args) throws Exception {
        Connection conn = null ; // 每一个Connection接口对象描述的就是一个用户连接
        Class.forName(DATABASE_DRIVER) ; // 向容器之中加载数据库驱动程序
        conn = DriverManager.getConnection(DATABASE_URL, DATABASE_USER,
DATABASE_PASSWORD) ;
        System.out.println(conn);
        conn.close(); // 数据库的连接资源有限一定要关闭
    }
}
```

如果此时在输出Connection接口对象的时候并没有产生任何的异常，那么就表示当前Oracle数据库已经可以正常的连接了。

但是这个时候肯定都在思考一个问题：为什么要连接Oracle数据库的时候一定要单独配置Oracle数据库的驱动程序呢？

首先JDBC是一个操作的标准，既然是标准，那么就证明里面是不牵扯到具体实现类，由数据库厂商来实现，下面将以上的连接过程通过类图的关系来进行描述：



通过结构可以发现，整个的JDBC设计实现的就是一个工厂类的处理机制。

DriverManager是一个工厂，不同的数据库生产商利用JDBC提供的标注(接口)实现各自的数据库处理操作。