

课程名称: JavaEE 技术	年级: XXX	实验日期:
姓名: XXX	学号: XXXXX	班级: XXX
实验名称: JDBC 操作数据库	实验序号: 实验一	成员人数: 1

## 一、实验目的及要求

### 实验目的:

- 1、学习 JDBC 操作数据库的编程方法;
- 2、了解 Eclipse 导入数据库驱动包的方法;

### 实验原理:

- 1、jdbc 是 sun 公司定义的一套访问数据库的规范(接口 + 类),主要是放在 java.sql 包下面。
- 2、导入相应 JDBC 的数据库的实现(数据库驱动)
- 3、Connection 接口代表数据库的链接,客户端所有的交互都是通过 Connection 对象完成的。

## 二、实验环境(仪器与材料)

Windows 7/10, MySQL/MariaDB, Eclipse, Tomcat

## 三、实验内容及完成情况

### 1、常用数据库的 URL 写法

Oracle: jdbc:oracle:thin:@localhost:1521:sid

SqlServer: jdbc:sqlserver://localhost:1433;DatabaseName=xxx

MySql: jdbc:mysql://localhost:3306/orcl

### 2、在 SQL Server 2008 中创建用户数据库

create database UsersManager

on

( name='UsersManager\_data',  
filename='d:\UsersManager\_data.mdf',  
size=10mb

)

log on

( name='UsersManager\_log',  
filename='d:\UsersManager\_log.ldf',  
size=10mb

)

go

use UsersManager

create table TUserLogin

(

id int primary key,

--标签编号, 主码

username nvarchar(20) not null,

--用户姓名 可变字符串类型, 长度 20

[password] nvarchar(20),

--密码 可变字符串类型, 长度 20

grade int default 1,

--用户级别, 默认为 1

email nvarchar(50)

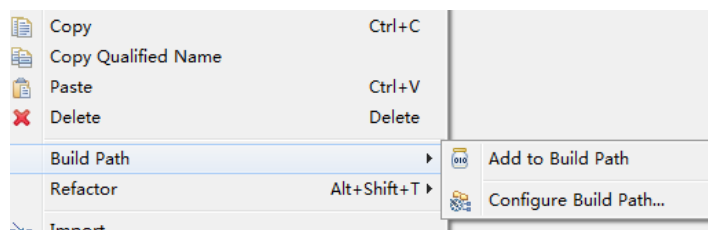
```

)
use UsersManager
insert into TUserLogin values(1,'wl1','123456',5,'wl1@126.com')
insert into TUserLogin values(2,'wl2','123456',1,'wl2@126.com')
insert into TUserLogin values(3,'wl3','123456',1,'wl3@126.com')
insert into TUserLogin values(4,'wl4','123456',1,'wl4@126.com')
insert into TUserLogin values(5,'wl5','123456',1,'wl5@126.com')
insert into TUserLogin values(6,'wl6','123456',1,'wl6@126.com')
insert into TUserLogin values(7,'wl7','123456',1,'wl7@126.com')
insert into TUserLogin values(8,'wl8','123456',1,'wl8@126.com')
insert into TUserLogin values(9,'wl9','123456',1,'wl9@126.com')
insert into TUserLogin values(10,'wl10','123456',1,'wl10@126.com')
insert into TUserLogin values(11,'admin','123456',5,'admin@126.com')

```

## 2、具体操作步骤:

- (1) 新建 java 工程，新建包，新建类，类名首字母大写。
- (2) 在工程下新建 lib 文件夹，引入 sqljdbc4. jar 驱动
- (3) 装载驱动完毕后，右键点击驱动包，在弹出的菜单中选择“Build Path” → “Add to Build Path”



- (4) 实验一：打印数据库中的全体用户名和电子邮件

```

package com.wl.jdbc;
import java.sql.*;
public class JdbcTest {
    public static void main(String[] args) throws SQLException {
        //使用JDBC操作Sql Server数据库
        Connection ct=null;
        PreparedStatement ps=null;
        ResultSet rs=null;
        try {
            //1 加载驱动
            Class.forName("com.mysql.jdbc.Driver");
            //2 得到链接
            ct = DriverManager.getConnection("jdbc:mysql://localhost:3306/
            usermanager?characterEncoding=utf-8","root","root");
            //创建sql对象
            ps=ct.prepareStatement("select * from TUserLogin");
            //执行sql语句
            rs=ps.executeQuery();
            while(rs.next()){
                System.out.println(rs.getString("username")+"|"+
                rs.getString("email"));
            }
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } finally{
            //关闭资源
            if(rs!=null){

```

```
<terminated> JdbcTest (1) [Java Ap
w11|w11@126.com
w12|w12@126.com
w13|w13@126.com|
w15|w15@126.com
w16|w16@126.com
w17|w17@126.com
w18|w18@126.com
w19|w19@126.com
```

编号: 12 姓名: mary 密码: 123456 级别: 3 电子邮件: mary@sohu.com

第 3 页 共 4 页

```
//关闭资源
if(rs!=null){
    rs.close();
}
if (ps!=null){
    try {
        ps.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
if(ct!=null){
    try {
        ct.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}
```

执行后，在 Sql Server 中执行 `select * from TUserLogin`，可以查看结果，也可以像实验一一样，定义一个结果集对象，使用 `while` 循环进行遍历输出。

运行结果：

id	username	password	grade	email
1	wl1	123456	1	wl1@126.com
2	wl2	123456	1	wl2@126.com
3	wl3	123456	1	wl3@126.com
5	wl5	123456	1	wl5@126.com
6	wl6	123456	1	wl6@126.com
7	wl7	123456	1	wl7@126.com
8	wl8	123456	1	wl8@126.com
9	wl9	123456	1	wl9@126.com
12	mary	123456	3	mary@sohu.com
NULL	NULL	NULL	NULL	NULL

#### 四、 出现的问题及解决方案

- 1、没有引入 `sqljdbc4.jar` 驱动，编程时没有智能提示，也无法运行。
- 2、输出遍历时，必须使用 `ResultSet` 对象，使用 `while(rs.next())` 进行遍历。
- 3、注入参数可以使用：`ps.setString(1, "12")` 的方式。这种方式比起字符串拼接的方式更加安全。
- 4、请把程序放置在包中，如果没有建立包，则系统会提供一个默认包，不利于管理。

#### 五、教师评语