

## 数据库操作

Java 的数据库操作有三重境界：

- 1、直接数据库操作，数据库连接、SQL 语句等。（自己包包子）
- 2、对应数据库中的表建一个类，以类的方式隔离数据库操作。（造一台包子机，不会包包子也能操作机器生产包子）
- 3、对于任意指定的表自动生成相应的类的 Java 程序。包括基本的数据库查询、显示与增、删、改等常规操作。（造一台能制造包子机、饺子机、馅饼机的机器）

本作业理论上应该有四~五个 Java 源程序：

- 1)对数据库的直接操作；建立连接、SQL 语句等。
- 2)对指定表建一个类，以类的方式实现对数据库操作的隔离。
- 3)对于任意指定表可自动产生相应的类的程序[程序 2]]。
- 4)1 到 2 个对于不同的表，由 3)产生的自动类实例。

### 1. 基础工作

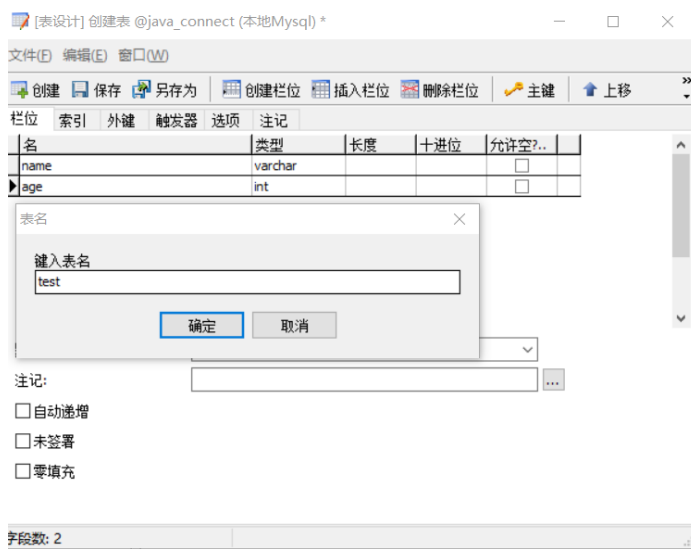
我们先连接 mysql，创建好相应的库和表。

建立库名为 java\_connect,, 库里的表名为 test。

```
mysql> create database java_connect;  
Query OK, 1 row affected (0.02 sec)
```

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| bjpowernode |  
| information_schema |  
| java_connect |  
| loginproject |  
| mhl |  
| mysql |  
| performance_schema |  
| sys |  
| travel |  
+-----+  
9 rows in set (0.00 sec)  
  
mysql>
```

可以发现 java\_connect 库已经创建好了, 为了便于观察就直接在 Navicat 中创建表格了。



这样我们就可以连接这个库了。

### 1. 直接连接方法

按部就班注意 mysql 版本以及语法 url = jdbc:mysql://127.0.0.1:3306 后要加上时区。

```

10  */
11  public class test1 {
12  public static void main(String[] args) {
13  try {
14  Class.forName("com.mysql.cj.jdbc.Driver");//注册驱动
15  } catch (ClassNotFoundException e) {
16  e.printStackTrace();
17  }
18
19  //获取连接
20  String url = "jdbc:mysql://127.0.0.1:3306/java_connect?serverTimezone=UTC";
21  String root = "root";
22  String password = "123456";
23
24  }
25  }

```

test1 x

D:\JAVA\JDK1.8\bin\java.exe ...

bill

20

tom

18

Process finished with exit code 0

### 2. 对应数据库中的表建一个类，以类的方式隔离数据库操作

将里面对表数据的操作给单独写一个类，这样便于用户的使用。

```

    ResultSet getResultSet(Statement statement, String sql) throws SQLException {
        ResultSet resultSet = statement.getResultSet();
        return resultSet;
    }

    void printfMessage(ResultSet rs) throws SQLException {
        while(rs.next()){
            System.out.println(rs.getString( columnLabel: "name"));
            System.out.println(rs.getString( columnLabel: "age"));
        }
    }
}

```

3. 对于任意指定的表自动生成相应的类的 Java 程序。包括基本的数据库查询、显示与增、删、改等常规操作。

对于任意表，这个表格必须与第二个程序里面的表有一定的相似性，这样才能直接根据表格生成相应的程序。

```

String name = "java_connect"; //这里输入你要连接的数据库名字
//补充一下test3中的几个操作方法，到时候调用test3类就可以实现对表的处理
//连接初试化操作在静态代码块中完成
{
    try {
        Class.forName("com.mysql.cj.jdbc.Driver");//注册驱动
    } catch (ClassNotFoundException e) {
        e.printStackTrace();
    }

    //获取连接
    String url = "jdbc:mysql://127.0.0.1:3306/" + name + "?serverTimezone=UTC";
    String root = "root";
    String password = "123456";
}

```

方法就简单实现了打印表格数据以及查询表格的操作，其基本方法与方法二类似。核心工作还是在方法二中，方法三只需换个表格即可。

小结：JDBC 连接数据库需要注意 mysql 版本，不同版本注册驱动有所不同，并且在新版本的 mysql 获取连接时，输入 url 地址时，除去输入对应的表格外，还要加上时区信息，缺少时区信息时无法完成数据库的连接。

```
String url = "jdbc:mysql://127.0.0.1:3306/" + name + "?serverTimezone=UTC";
```