首先我们需要往所用的软件中添加相对应的sqljdbc.jar。

大体流程

1、加载JDBC驱动程序：

　　在连接数据库之前，首先要加载想要连接的数据库的驱动到JVM（Java虚拟机），这通过java.lang.Class类的静态方法forName(String  className)实现。

2、提供JDBC连接的URL。

3、创建数据库的连接：

　　•要连接数据库，需要向java.sql.DriverManager请求并获得Connection对象， 该对象就代表一个数据库的连接。

　　•使用DriverManager的getConnectin(String url , String username , String password )方法传入指定的欲连接的数据库的路径、数据库的用户名和密码来获得。

4、创建一个Statement：

　　•要执行SQL语句，必须获得java.sql.Statement实例，Statement实例分为以下3种类型：     
      （1）执行静态SQL语句。通常通过Statement实例实现。   Statement stmt = con.createStatement() ;   
      （2）执行动态SQL语句。通常通过PreparedStatement实例实现。 PreparedStatement pstmt = con.prepareStatement(sql) ;     
      （3）执行数据库存储过程。通常通过CallableStatement实例实现。   CallableStatement cstmt = con.prepareCall("{CALL demoSp(? , ?)}") ;

5、执行SQL语句：

　　Statement接口提供了三种执行SQL语句的方法：executeQuery 、executeUpdate和execute     
    （1）ResultSet executeQuery(String sqlString)：执行查询数据库的SQL语句，返回一个结果集（ResultSet）对象。     
    （2）int executeUpdate(String sqlString)：用于执行INSERT、UPDATE或DELETE语句以及SQL DDL语句，如：CREATE TABLE和DROP TABLE等     
    （3）execute(sqlString):用于执行返回多个结果集、多个更新计数或二者组合的语句。     
     ResultSet rs = stmt.executeQuery("SELECT \* FROM ...") ;     
     int rows = stmt.executeUpdate("INSERT INTO ...") ;     
     boolean flag = stmt.execute(String sql) ;

6、处理结果：

　　两种情况：     
     （1）执行更新返回的是本次操作影响到的记录数。     
     （2）执行查询返回的结果是一个ResultSet对象。     
   　　 • ResultSet包含符合SQL语句中条件的所有行，并且它通过一套get方法提供了对这些行中数据的访问。     
   　　 • 使用结果集（ResultSet）对象的访问方法获取数据：

//（列是从左到右编号的，并且从列1开始）

while(rs.next()){

String name = rs.getString("name") ;

　 int id= rs.getInt("id") ;

}

7、关闭JDBC对象      
     操作完成以后要把所有使用的JDBC对象全都关闭，以释放JDBC资源，关闭顺序和声明顺序相反：     
     1、关闭记录集     
     2、关闭声明     
     3、关闭连接对象

[IMG_256](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

if(rs != null){ // 关闭记录集

try{

rs.close() ;

}catch(SQLException e){

e.printStackTrace() ;

}

}

if(stmt != null){ // 关闭声明

try{

stmt.close() ;

}catch(SQLException e){

e.printStackTrace() ;

}

}

if(conn != null){ // 关闭连接对象

try{

conn.close() ;

}catch(SQLException e){

e.printStackTrace() ;

}

}

[IMG_257](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

一、编写DBUtil.java

[IMG_258](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

package Database;import java.sql.\*;

public class DBUtil {

//这里可以设置数据库名称

private final static String URL = "jdbc:sqlserver://localhost:1433;DatabaseName=People";

private static final String USER="sa";

private static final String PASSWORD="123456";

private static Connection conn=null;

//静态代码块（将加载驱动、连接数据库放入静态块中）

static{

try {

//1.加载驱动程序

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

//2.获得数据库的连接

conn=(Connection)DriverManager.getConnection(URL,USER,PASSWORD);

} catch (ClassNotFoundException e) {

e.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

//对外提供一个方法来获取数据库连接

public static Connection getConnection(){

return conn;

}

//测试用例

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

//3.通过数据库的连接操作数据库，实现增删改查

Statement stmt = conn.createStatement();

//ResultSet executeQuery(String sqlString)：执行查询数据库的SQL语句 ，返回一个结果集（ResultSet）对象。

ResultSet rs = stmt.executeQuery("select id,name,age from BasicInfo");

while(rs.next()){//如果对象中有数据，就会循环打印出来

System.out.println(rs.getInt("id")+","+rs.getString("name")+","+rs.getInt("age"));

}

}

}

[IMG_259](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

二、编写一个示例的模型类Person

[IMG_260](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

package Database;

public class Person {

private int id;

private String name;

private int age;

public int getId(){return this.id;}

public void setId(int id){this.id = id;}

public String getName(){return this.name;}

public void setName(String name){this.name = name;}

public int getAge(){return this.age;}

public void setAge(int age){this.age = age;}

}

[IMG_261](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

三、编写PersonDao.java

[IMG_262](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

package Database;import java.sql.\*;import java.util.ArrayList;import java.util.List;

public class PersonDao {

//------------------------------------------add--------------------------------------------------------

public void addPerson(Person person)throws SQLException{

//首先拿到数据库的连接

Connection conn=DBUtil.getConnection();

String sql="" +

"insert into BasicInfo"+

"(id,name,age) "+

"values(?,?,?)";//参数用?表示，相当于占位符;

//预编译sql语句

PreparedStatement psmt = conn.prepareStatement(sql);

//先对应SQL语句，给SQL语句传递参数

psmt.setInt(1, person.getId());

psmt.setString(2, person.getName());

psmt.setInt(3, person.getAge());

//执行SQL语句 psmt.execute();

/\*\*

\* prepareStatement这个方法会将SQL语句加载到驱动程序conn集成程序中，但是并不直接执行

\* 而是当它调用execute()方法的时候才真正执行；

\*

\* 上面SQL中的参数用?表示，相当于占位符，然后在对参数进行赋值。

\* 当真正执行时，这些参数会加载在SQL语句中，把SQL语句拼接完整才去执行。

\* 这样就会减少对数据库的操作

\*/

}

//------------------------------------------add--------------------------------------------------------

//------------------------------------------update--------------------------------------------------------

public void updatePerson(Person person)throws SQLException{

//首先拿到数据库的连接

Connection conn=DBUtil.getConnection();

String sql="" +

"update BasicInfo set name = ?,age = ? where id = ?";//参数用?表示，相当于占位符

//预编译sql语句

PreparedStatement psmt = conn.prepareStatement(sql);

//先对应SQL语句，给SQL语句传递参数

psmt.setString(1, person.getName());

psmt.setInt(2, person.getAge());

psmt.setInt(3, person.getId());

//执行SQL语句 psmt.execute();

}

//------------------------------------------update--------------------------------------------------------

//------------------------------------------delete--------------------------------------------------------

public void deletePerson(int id) throws SQLException{

Connection conn=DBUtil.getConnection();

String sql="" +

"delete from BasicInfo where id = ?";

PreparedStatement psmt = conn.prepareStatement(sql);

psmt.setInt(1,id);

//执行SQL语句 psmt.execute();

}

//------------------------------------------delete--------------------------------------------------------

//------------------------------------------SearchOne--------------------------------------------------------

public Person SearchOne(int id) throws SQLException{

Person p = null;

Connection conn=DBUtil.getConnection();

String sql="" +

"select \* from BasicInfo where id = ?";

PreparedStatement psmt = conn.prepareStatement(sql);

psmt.setInt(1,id);

//执行SQL语句

ResultSet rs = psmt.executeQuery();

while(rs.next()){

p = new Person();

p.setId(rs.getInt("id"));

p.setName(rs.getString("name"));

p.setAge(rs.getInt("age"));

}

return p;

}

//------------------------------------------SearchOne--------------------------------------------------------

//------------------------------------------Search--------------------------------------------------------

public List<Person> Search() throws SQLException{

Connection conn = DBUtil.getConnection();

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("select id,name,age from BasicInfo");

List<Person> people = new ArrayList<Person>();

Person p = null;

while(rs.next()){//如果对象中有数据，就会循环打印出来

p = new Person();

p.setId(rs.getInt("id"));

p.setName(rs.getString("name"));

p.setAge(rs.getInt("age"));

people.add(p);

}

return people;

}

//------------------------------------------Search--------------------------------------------------------

}

[IMG_263](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

四、编写Test

[IMG_264](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

package Database;import java.sql.\*;import java.util.ArrayList;import java.util.List;

public class Test {

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

// TODO Auto-generated method stub

PersonDao p = new PersonDao();

//add

Person person0 = new Person();

person0.setId(1);

person0.setName("小明");

person0.setAge(20);

p.addPerson(person0);

//update

Person person1 = new Person();

person1.setId(1);

person1.setName("陈伟霆");

person1.setAge(35);

p.updatePerson(person1);

//delete

int id = 1;

System.out.println(p.SearchOne(id).getName());

//search

List<Person> people = new ArrayList<Person>();

people = p.Search();

for(Person person : people){

String str = person.getId()+","+person.getName()+","+person.getAge();

System.out.println(str);

}

}

}

[IMG_265](https://www.cnblogs.com/zhangxiaoshuang/p/javascript:void(0);" \o "复制代码)

参考博客：http://www.cnblogs.com/Qian123/p/5339164.html