# 第10章

1. 填空
2. Connection
3. last()
4. 预编译
5. update
6. DriverManager
7. 判断
8. 错
9. 对
10. 对
11. 对
12. 错
13. 选择
14. A
15. A
16. C
17. C
18. B
19. 简答
20. JDBC编程的步骤如下：

（1）加载并注册数据库驱动。

（2）通过DriverManager获取数据库连接。

（3）通过Connection对象获取Statement对象。

（4）使用Statement对象执行SQL语句。

（5）操作ResultSet结果集。

（6）关闭连接，释放资源。

1. 编程

1. 该程序代码如下所示：

import java.sql.\*;

public class Example01 {

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

// 1. 注册数据库的驱动

DriverManager.registerDriver(new com.mysql.jdbc.Driver());

// 2.通过DriverManager获取数据库连接

String url = "jdbc:mysql://localhost:3306/jdbc";

String username = "root";

String password = "itcast";

Connection conn = DriverManager.getConnection(url, username, password);

// 3.通过Connection对象获取Statement对象

Statement stmt = conn.createStatement();

// 4.使用Statement执行SQL语句。

String sql = "select \* from users";

ResultSet rs = stmt.executeQuery(sql);

// 5. 操作ResultSet结果集

System.out.println("id | name | password | email |");

while (rs.next()) {

int id = rs.getInt("id"); // 通过列名获取指定字段的值

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

String psw = rs.getString("password");

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

System.out.println(id + " | " + name + " | " + psw + " | " + email);

}

// 6.回收数据库资源

rs.close();

stmt.close();

conn.close();

}

}

2. 该功能的实现步骤如下：

（1）JDBCUtils工具类的代码如下所示：

import java.sql.\*;

public class JDBCUtils {

// 加载驱动，并建立数据库连接

public static Connection getConnection() throws SQLException,

ClassNotFoundException {

DriverManager.registerDriver(new com.mysql.jdbc.Driver());

String url = "jdbc:mysql://localhost:3306/jdbc";

String username = "root";

String password = "itcast";

Connection conn = DriverManager.getConnection(url, username,

password);

return conn;

}

// 关闭数据库连接，释放资源

public static void release(Statement stmt, Connection conn) {

if (stmt != null) {

try {

stmt.close();

} catch (SQLException e) {

e.printStackTrace();

}

stmt = null;

}

if (conn != null) {

try {

conn.close();

} catch (SQLException e) {

e.printStackTrace();

}

conn = null;

}

}

public static void release(ResultSet rs, Statement stmt,

Connection conn){

if (rs != null) {

try {

rs.close();

} catch (SQLException e) {

e.printStackTrace();

}

rs = null;

}

release(stmt, conn);

}

}

（2）Example02的代码如下所示：

import java.sql.Connection;

import java.sql.PreparedStatement;

public class Example02 {

public static void main(String[] args) {

Connection conn = null;

PreparedStatement preStmt = null;

try {

// 加载并注册数据库驱动

conn=JDBCUtils.getConnection();

String sql = "INSERT INTO users(name,password)VALUES(?,?)";

preStmt = conn.prepareStatement(sql);

for (int i = 0; i < 5; i++) {

preStmt.setString(1, "name2" + i);

preStmt.setString(2, "password2" + i);

preStmt.addBatch();

}

preStmt.executeBatch();

} catch (Exception e) {

e.printStackTrace();

} finally { // 释放资源

JDBCUtils.release(null, preStmt, conn);

}

}

}