**《Java ProgramingⅡ》实验报告**

**班级：软件工程2班**

**学号：162210702206**

**姓名：车玉华**

目录

[第1章 实验一 java 语言基本技能 2](#_Toc518121133)

[1.1 实验目的 2](#_Toc518121134)

[1.2 实验内容和过程 2](#_Toc518121135)

[1.3 整体思路 2](#_Toc518121136)

[第2章 实验二 JDBC数据库编程 3](#_Toc518121137)

[2.1 实验目的 3](#_Toc518121138)

[2.2 实验内容和过程 3](#_Toc518121139)

[2.3 整体思路 3](#_Toc518121140)

[2.4 源代码： 3](#_Toc518121141)

[第3章 实验三 Java web application 4](#_Toc518121142)

[3.1 实验目的 4](#_Toc518121143)

[3.2 实验内容和过程 4](#_Toc518121144)

[3.3 整体思路 5](#_Toc518121145)

[3.4 源代码 5](#_Toc518121146)

# 实验一 java 语言基本技能

## 实验目的

掌握java 语言的一些基本技能。

## 实验内容和过程

* static域和方法的理解和使用
* this理解和使用
* String不可变类的理解和使用
* StringBuilder可变类的理解和使用

## 整体思路

1. this关键字指的是本身对象，属于对象的变量。
2. String类的底层实现是一个定长的数组，这就是真正的不可变，当String字符串相加的时候，其实是调用数组的复制，将2个数组拷贝到新的数组。
3. StringBuilder的底层实现也是一个数组，但是这个数组具有扩容机制，当检测到数组要超过上限的时候，会按当前某个倍数扩容。同时，StringBuilder是线程不安全的，StringBuffer是线程安全的两者的区别是，StringBuffer多了一个synchronized关键字

# 实验二 JDBC数据库编程

## 实验目的

学会使用JDBC存取典型关系数据库，比如MySql数据，对数据进行适当的处理。

## 实验内容和过程

* 驱动程序装载
* 数据库连接
* 数据库数据的存取
* 数据库数据的处理

## 整体思路

1. 首先下载MySQL的JDBC驱动包,然后导入项目中
2. 使用Class.forName(“com.mysql.jdbc.Driver”)注册驱动
3. 通过DriverManager获取connection
4. 使用connection.createStatement()得到Statement
5. 通过Statement执行CRUD
6. 以上过程完全可以封装成一个工具类,降低耦合,同时也避免了代码的冗余

## 源代码：

**import java.sql.DriverManager;**

**import java.sql.ResultSet;**

**import java.sql.SQLException;**

**import java.sql.Statement;**

**import java.sql.Connection;**

**import java.sql.DriverManager;**

**import java.sql.SQLException;**

**public class javaTest {**

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

**String URL="jdbc:mysql://127.0.0.1:3306/imooc?useUnicode=true&amp;characterEncoding=utf-8";**

**String USER="root";**

**String PASSWORD="tiger";**

**//1.加载驱动程序**

**Class.forName("com.mysql.jdbc.Driver");**

**//2.获得数据库链接**

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

**//3.通过数据库的连接操作数据库，实现增删改查（使用Statement类）**

**Statement st=conn.createStatement();**

**ResultSet rs=st.executeQuery("select \* from user");**

**//4.处理数据库的返回结果(使用ResultSet类)**

**while(rs.next()){**

**System.out.println(rs.getString("user\_name")+" "**

**+rs.getString("user\_password"));**

**}**

**//关闭资源**

**rs.close();**

**st.close();**

**conn.close();**

**}**

**}**

# 实验三 Java web application

## 实验目的

了解和认识Java web application的涉及的核心技术、组成和开发过程

## 实验内容和过程

实现一个简单的学生信息管理：学生信息（学号、姓名、性别和年龄等）输入；学生信息的查询；

* 客户端设计与实现
  + HTML
  + CSS
  + Javascript
* 服务器端设计与实现
  + Servlet
  + Java
  + JDBC

## 整体思路

1. 客户端的设计:

1. 使用HTML5设计客户端的整体框架

第一部分为提交组,包含学号,姓名,性别和年龄的输入以及一个提交按钮

第二部分为查询组,包含学号的输入以及一个提交按钮

2. 使用CSS设计HTML文档的样式

3. 使用JavaScript设计HTML控件的行为,如点击按钮后向服务器发送数据并返回信息

2. 服务端的设计:使用Servlet承载HTML页面;使用JDBC连接mysql数据库,当有用户使用提交组的按钮提交学号,姓名,性别,年龄信息时,将其插入至数据库,当有用户使用查询组的按钮查询学生信息时,查询保存在数据库的信息并显示于HTML页面上

## 源代码

//MyServlet.java

package demo.servlet;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\* Servlet implementation class MyServlet

\*/

@WebServlet("/MyServlet")

public class MyServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public MyServlet() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String str=request.getParameter("num");

response.setContentType("text/html;charset=UTF-8");

String name=getName(str);

PrintWriter out=response.getWriter();

out.write(name);

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

doGet(request, response);

}

//根据学号获取用户名

public String getName(String num) {

String str="";

//连接数据库

Connection connection;

ResultSet set;

try {

Class.forName("com.mysql.jdbc.Driver");

connection=DriverManager.getConnection("jdbc:mysql://localhost:3306/student","root","123456");

Statement statement=connection.createStatement();

String sql="select name from student\_info where num='"

+ num

+ "'";

set=statement.executeQuery(sql);

while(set.next()) {

str=set.getString("name");

}

set.close();

connection.close();

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return str;

}

}

//student\_info.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Student\_info</title>

</head>

<body>

<form action="MyServlet" method="GET">

学号<input type="text" name="num">

<input type="submit" value="submit">

</form>

<form action="MyServlet" method="POST">

学号:<input type="text" name="num">

姓名:<input type="text" name="name">

性别:<input type="text" name="sex">

年龄:<input type="text" name="age">

<input type="submit" value="submit">

</form>

</body>

</html>

实验结果:

