个人健康管理系统

目 录

[个人健康管理系统 1](#_Toc674)

[第一章 简介 2](#_Toc6937)

[1.1Java简述 2](#_Toc2272)

[1.2 软件简述 3](#_Toc29627)

[第二章 需求分析（图） 3](#_Toc18825)

[2.1 设计草图 3](#_Toc11309)

[2.2 功能需求 4](#_Toc8834)

[第三章 系统设计 7](#_Toc17778)

[3.1设计目的 7](#_Toc26821)

[3.2类与接口 7](#_Toc16529)

[第四章 系统实现 8](#_Toc337)

[4.1 实现技术 8](#_Toc32347)

[4.2 工具类JDBC驱动 9](#_Toc16977)

[4.3 JDBC简介 9](#_Toc6755)

[第五章 测试 9](#_Toc26301)

[总结 11](#_Toc688)

第一章 简介

1.1Java简述

20世纪90年代，硬件领域出现了单片式计算机系统，这种价格低廉的系统一出现就立即引起了自动控制领域人员的注意， 因为使用它可以大幅度提升消费类电子产品(如电视机顶盒、面包烤箱、移动电话等)的智能化程度。Sun公司为了抢占市场先机，在1991年成立了一一个称为Green的项目小组，帕特里克、詹姆斯高斯林、麦克舍林丹和其他几个工程师一起组成的工作小组在加利福尼亚州]洛帕克市沙丘路的一个小工作室里面研究开发新技术，专攻计算机在家电产品上的嵌入式应用。

由于C+ +所具有的优势,该项目组的研究人员首先考虑采用C+ +来编写程序。但对于硬件资源极其匮乏的单片式系统来说，C+ +程序过于复杂和庞大。另外由于消费电子产品所采用的嵌入式处理器芯片的种类繁杂，如何让编写的程序跨平台运行也是个难题。为了解决困难， 他们首先着眼于语言的开发，假设了- 种结构简单、符合嵌入式应用需要的硬件平台体系结构并为其制定了相应的规范，其中就定义了这种硬件平台的二进制机器码指令系统(即后来成为字节码的指令系统) ,以待语言开发成功后,能有半导体芯片生产商开发和生产这种硬件平台。对于新语言的设计，Sun公司研发人员并没有开发一 种全新的语言，而是根据嵌入式软件的要求，对C++进行了改造，去除了留在C++的一些不太实用及影响安全的成分，并结合嵌入式系统的实时性要求，开发了一 种称为Oak的面向对象语言。

Java语言是一种高级的面向对象的程序设计语言。高级语言是与低级语言相对，低级语言指的是汇编语言，高级语言指的是类似java、c、c++等常用语言。Java语言编写的程序，可以不受平台限制，也就是跨平台，从PC到手机都有Java开发的程序和游戏，Java程序可以在任何计算机、操作系统和支持Java的硬件设备上运行。Java是于1995年由Sun公司推出的一种极富创造力的面向对象的程序设计语言，它是由有Java之父的Sun研究院院士詹姆斯.戈士林博士亲手设计而成的，并完成Java技术的原始编译器和虚拟机

1.2 软件简述

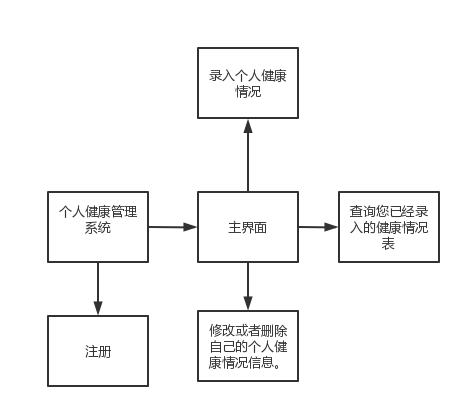
系统简述：

当通过数据库存储的账号密码进行登陆系统后，系统会直接进入信息管理的界面，里面可以插入自己的个人健康情况进行进行录入，然后可以查询到自己的健康情况。

你也可以选择自己的个人健康情况进行更改，或者针对自己的个人健康情况进行针对处理方案。

第二章 需求分析（图）

2.1 设计草图



2.2 功能需求

①登录：通过数据库账号密码进行登录。

②注册：当没有可以使用的账号时候可以先进行注册然后进行登录个人健康信息管理系统。

else if (one==2){

System.*out*.println("-注册界面");

System.*out*.println("请输入您像录入的车牌号(省分+大写字母+5位数字或者字母例如:辽A666666)");

String carId = scanner.next();

System.*out*.println("请输入您的联系电话)");

String houstPhone = scanner.next();

car.insertCarUser(new CarUser(carId,houstPhone));

System.*out*.println("注册成功返回登陆界面");

continue;

}

else

{

System.*err*.println("");

break;

}

③主界面

可以登录后进入主界面进行个人健康管理系统的管理操作，可以进行录入信息，也可以修改已经录入的信息，或者删除已经录入的信息。

if (stopOne==1){

System.*out*.println("-取车");

//渲染

List list = car.selectAllOrders();

if (list!=null){

for (int m = 0; m <list.size() ; m++) {

System.*out*.println(list.get(m));

}

System.*out*.println("请输入车牌号");

CarOrder carOrder = car.queryOrder(carId);

System.*out*.println(carOrder);

for (int m = 0; m >-1 ; m++) {

System.*out*.println("您应付"+carOrder.getPrice()+"元");

System.*out*.println("请输入您支付的金额");

String num = scanner.next();

if (Integer.valueOf(num)>=carOrder.getPrice()){

int e = Integer.valueOf(num)-carOrder.getPrice();

System.*out*.println("支付成功，找零"+e+"元");

break;

}

else{

System.*err*.println("支付有误重新支付");

continue;

}

}

}

else {

System.*out*.println("没有停车信息");

break;

}

else if (two==3){

System.*out*.println("请输入您想修改的手机号");

String phone = scanner.next();

car.updateNewUser(new CarUser(carId,phone));

System.*out*.println("修改成功");

continue;

}

else {

break;

}

1. 系统设计

3.1设计目的

可以作为自己的健康记录本，可以根据针对自己的体检情况实施一些自己需要修改的措施，方便一些老人或者学生使用此系统。

3.2类与接口

类和类：继承关系且只能单继承，可以多层继承

类和接口：实现关系，可以单实现，也可以多实现

接口和接口：继承关系，可以是单继承，也可以多继承

成员区别

　　抽象类：

　　　　成员变量：可以是变量，可以是常量

　　　　有构造方法

　　　　成员方法：可以是抽象的，可以是非抽象的

　　接口：

　　　　成员变量：只能是静态的常量

　　　　没有构造方法

　　　　成员方法：只能是抽象的

关系区别

　　类和类：继承关系，只能单继承，可以多层继承。

　　类和接口：实现关系，可以单实现，也可以多实现。

　　接口和接口：继承关系，可以单继承，也可以多继承。、

设计区别：

　　抽象类：定义的是该继承体系中的共性内容

　　　　被继承的，体现的是is a的关系

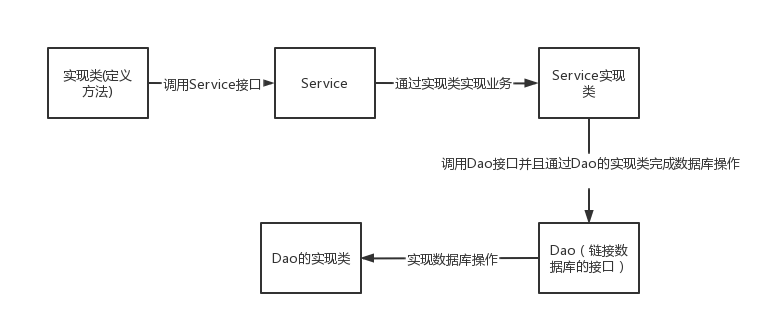
　　接口：定义的是事物扩展的功能

　　　　被实现的，体现的是like a的关系

1. 系统实现

4.1 实现技术

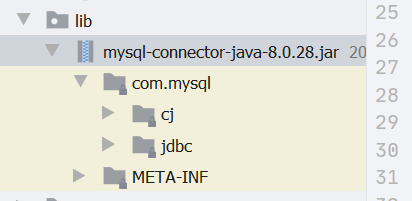
通过三层架构mvc模型的形式进行实现，接口和实现类进行实现。



所有的业务通过最开始的实现类实现，实现类内部有很多方法，每个方法内部都会通过Service实现类创建Service对象，也就是业务层，通过Service调用已经写好了的业务方法，通过业务方法的实现类，创建Dao层的实现类创建Dao的对象并且通过实现类完成对数据库的增删改查的操作。最终的返回值会在实现类中出现。

4.2 工具类JDBC驱动

想让java与数据库中间有关系，那么就需要一个驱动去解决这个问题，通过创建一个工具类，



4.3 JDBC简介

JDBC的全称是Java数据库连接(Java Database connect)，它是一套用于执行SQL语句的Java API。应用程序可通过这套API连接到关系数据库，并使用SQL语句来完成对数据库中数据的查询、更新和删除等操作。应用程序使用JDBC访问数据库的方式如下图所示。

第五章 测试

①完成登陆操作测试



测试结果成功

②注册测试



测试结果成功

③添加新的健康信息



测试结果成功

③修改健康信息



测试成功

总结

在java课程设计中，要认清楚到底这在我们的生活中的作用是什么，我个人认为这在我们的今后生活中，这酒店的设计是一种在以后的工作中能起到很有用处的课程设计，这也许是老师这用做的为我们着想的一部分吧。在这次课程设计中，我们也能可以看出，学计算机的在以后找工作中的出处，让我们知道计算机技术的应用的广泛应用。在java课程的学习中，也使我明白了更多的以前没有没有见过的知识，学到了更多的知识，更完整的弄清楚计算机科学与技术这门学科。在学java这门学科时，我开始还以为它跟c差不多的，但随得学习的深入就弄明白了这两者的关系有得很大的差别，在写程序上就能看出来这其中的差别的不同处。