垃圾分类回收系统

数据库设计说明书

目录

[第一部分：系统设计内容 2](#_Toc532915815)

[第一章 绪论 2](#_Toc532915816)

[1课题简介 2](#_Toc532915817)

[2设计目的 3](#_Toc532915818)

[3设计内容 3](#_Toc532915819)

[第二章 需求分析 4](#_Toc532915820)

[1功能需求 4](#_Toc532915821)

[**1.1 社区用户** 4](#_Toc532915822)

[**1.2 回收站用户** 4](#_Toc532915823)

[**1.3 管理员用户** 4](#_Toc532915824)

[2 性能需求 5](#_Toc532915826)

[第三章 概念结构设计 6](#_Toc532915827)

[1局部E-R图 6](#_Toc532915828)

[2全局E-R图 9](#_Toc532915829)

[第四章 逻辑结构设计 10](#_Toc532915830)

[1关系模式 10](#_Toc532915831)

[2数据库结构的详细设计 11](#_Toc532915832)

[第五章 物理结构设计 15](#_Toc532915833)

[1建立索引 15](#_Toc532915834)

[2存储结构 15](#_Toc532915835)

[3数据库的建立 16](#_Toc532915836)

# 第一部分：系统设计内容

## 第一章 绪论

### 1项目简介

随着垃圾分类政策的不断实施，人们开始重视垃圾分类回收，但是垃圾种类繁多，材质、颜色和外形各异，而且在不同地域和不同场景下的类别划分差异也很大。比如有按干/湿划分类别的，有按可燃/不可燃划分类别的等。原来的人工回收处理的方式，甚至是一般的垃圾分类回收软件都已经不能够满足现代垃圾回收的需要。因为这些传统的垃圾处理方式存在太多的缺陷，例如：

（1）、缺少垃圾分类，处理效率低

（2）、查询垃圾信息不方便

（3）、社区垃圾处理成本大

随着科学技术的不断提高,计算机科学日渐成熟,其强大的功能已为人们深刻认识,它进入人类社会的各个领域并发挥着越来越重要的作用。作为计算机应用的一部分,使用计算机对垃圾分类回收进行管理,有着手工管理所无法比拟的优点.例如:检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大地提高垃圾分类回收的效率,也是企业的科学化、正规化管理,与世界接轨的重要条件。

### 2设计内容

运用基于E-R 模型的数据库设计方法和关系规范化理论做指导完成从系统的分析到设计直至系统的最终实现。

首先做好需求分析，并完成数据流图和数据字典。

其次做概念结构设计，利用实体联系的方法将用户需求抽象为信息结构，得到E-R 图。

再次就是逻辑结构设计，将E-R 图转换为计算机系统所支持的逻辑模型。

然后就是物理结构设计，将逻辑数据模型转换为计算机系统所支持的数据库物理结构。

最后就是数据库的实施与运行，把数据载入及运行数据库。

## 第二章 需求分析

### 1功能需求

该垃圾分类回收系统共有三个角色：普通用户、垃圾回收员、管理员各个角色实现的功能如下：

**1.1 普通用户**

1. 拍照查询垃圾类别。
2. 请求垃圾回收。
3. 修改个人信息。
4. 修改密码。

**1.2 垃圾回收员**

1. 进行垃圾回收处理。
2. 查看本人负责的社区。
3. 修改个人信息
4. 修改密码。

**1.3** **管理员（Administrator）用户**

1. 用户管理：查询、添加、修改、删除用户信息。查询用户人数。
2. 垃圾信息管理：查询、添加、修改、删除垃圾信息。
3. 订单管理：查询、添加、修改、删除订单信息。查询订单数。
4. 垃圾回收员管理：添加、修改、删除垃圾回收员信息。查询垃圾回收员人数。

### 2性能需求

（1）支持并发访问数量在1000以上；

（2）各类操作系统响应时间控制在5s以内；

（4）要求具有良好的安全性和完整性控制。

## 第三章 概念结构设计

根据系统需求分析，可以得出垃圾分类回收系统数据库的概念模型（信息模型），以下图是用E-R图表示的垃圾分类回收系统的概念模型。最后，对设计出的数据模型进行规范化处理，使数据模型满足第三范式。

### 1局部E-R图

将改系统定名为“垃圾分类回收系统”。 各实体的分E-R图，如下图所示：

垃圾

**图1.1 垃圾实体图**

社区

**图1.2 社区实体图**

回收站

**图1.3 回收站实体图**

回收员

**图1.4 回收员实体图**

居民

**图1.5 居民实体图**

用户

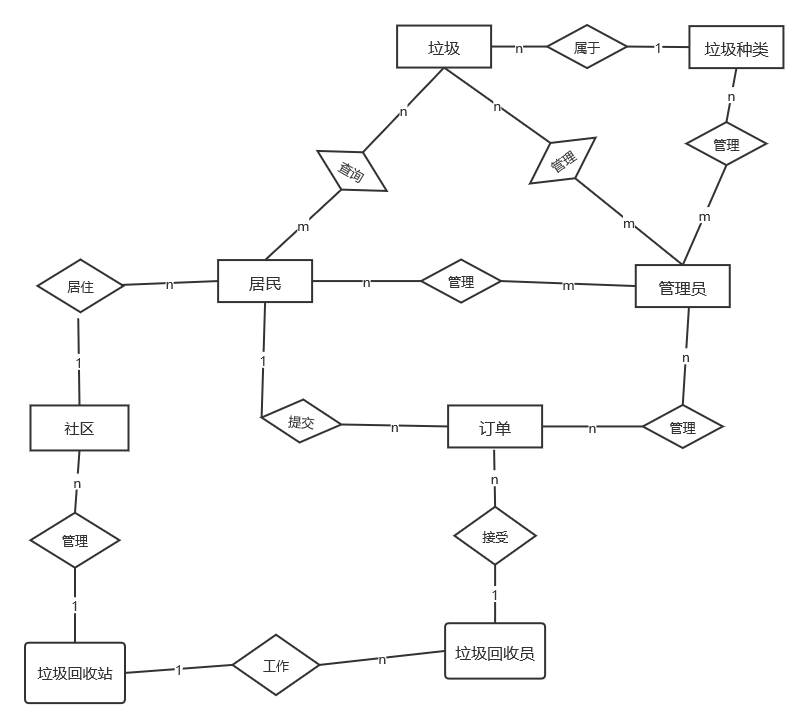
**图1.6用户实体图**

订单

**图1.7 订单实体图**

### 2全局E-R图

全局E-R图，如图E所示：

****

**图E 各实体间的联系**

## 第四章 逻辑结构设计

### 1关系模式

将垃圾分类回收系统的E-R图转换为关系数据库的数据模型，其关系模式为：

（1）社区（sno,name,manager,location,telenumber）  
（社区基本信息记录：社区号、社区名、负责人、地址、联系方式）

主码：sno

外码：无

（2）居民（jno,name,sno,sex,birthday,home,age,telenumber）  
 （居民基本信息记录：用户号、姓名、社区号、性别、生日，联系方式，年龄，家庭地址）

主码：jno

外码：sno参照社区（sno）

1. 垃圾回收站（hsno,name,manager,region,telenumber）

（垃圾回收站基本信息记录：回收站号、回收站名、负责人、管理范围，联系方式）

主码：hsno

外码：无

（4）垃圾回收员（hno,name,hsno,sex,birthday,salary,telenumber）  
 （垃圾回收员基本信息记录：工号、姓名、单位、性别、生日、薪水、联系方式）

主码：hno

外码：hsno参照垃圾回收站（hsno）

1. 垃圾（lno,name,ttype,ntype）

（垃圾基本信息记录：垃圾编号、垃圾名称、传统分类、新分类）

主码：lno

外码：无

1. user（name，password，tyoe）  
    （用户基本信息记录：用户名，密码，用户类别）

主码：name

1. 订单（dno,jno,hno,stime,ftime,price,location）  
    （订单基本信息记录：订单号、用户号、工号、下单时间、完成时间、价格、地址）

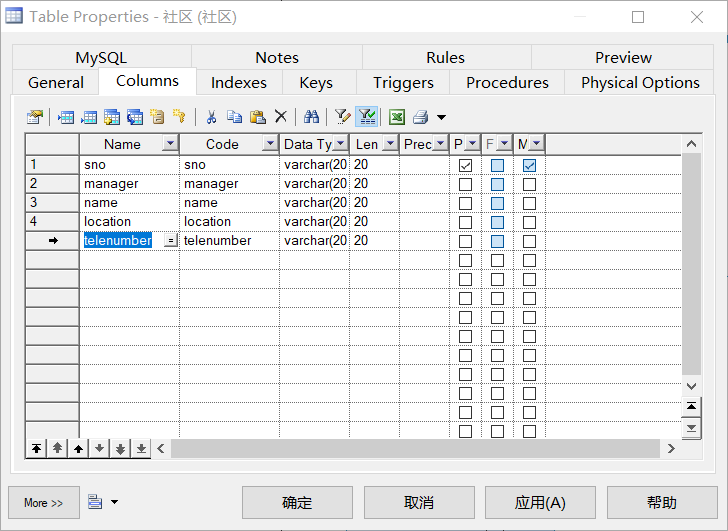
主码：dno

外码：jno参照居民（jno）；hno参照垃圾回收站（hno）

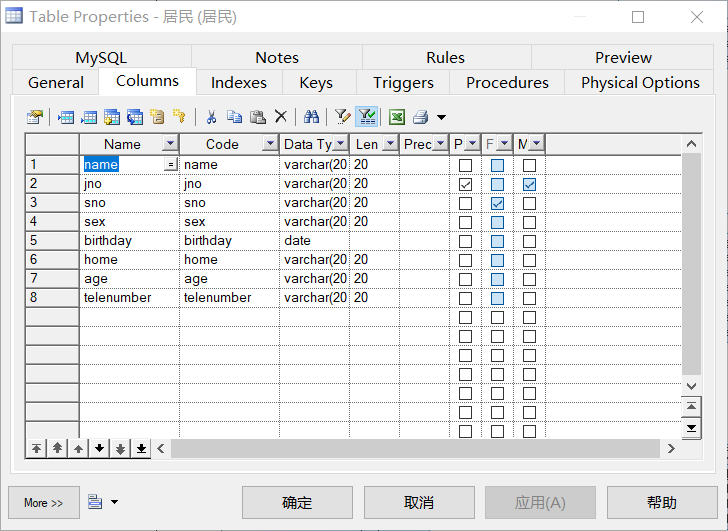
### 2数据库结构的详细设计

经过检查，此数据库的详细逻辑设计满足第3范式的要求。垃圾分类回收系统中各表的属性计情况如下：

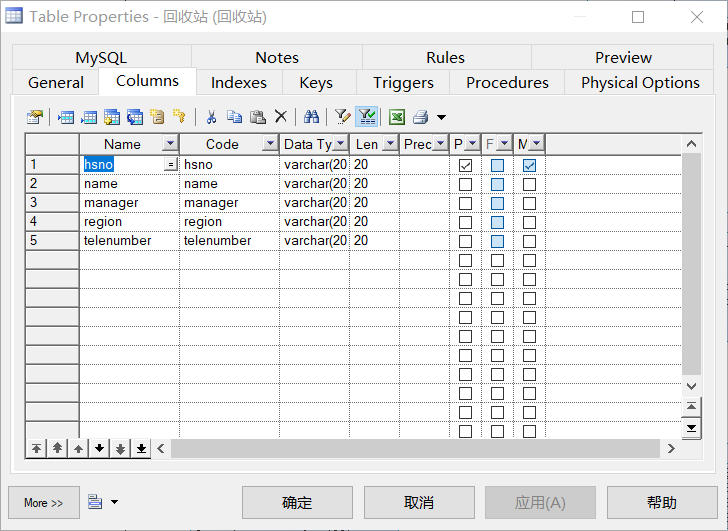
**表2-1**社区表



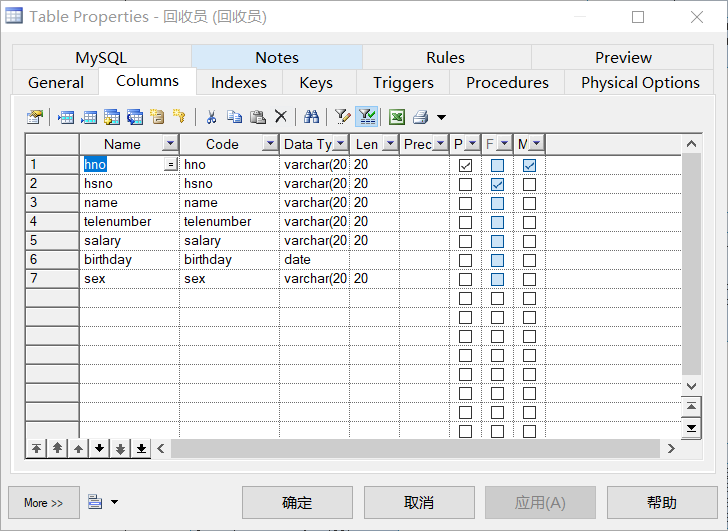
**表2-2**居民表



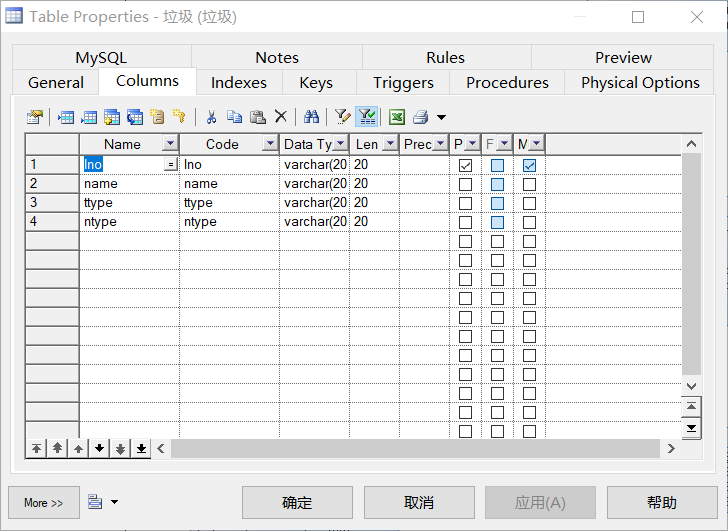
**表2-3**垃圾回收站表



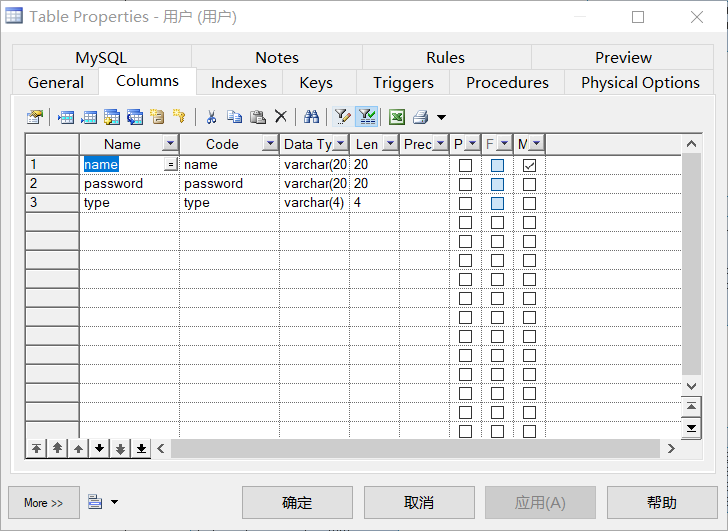
**表2-4**垃圾回收员表



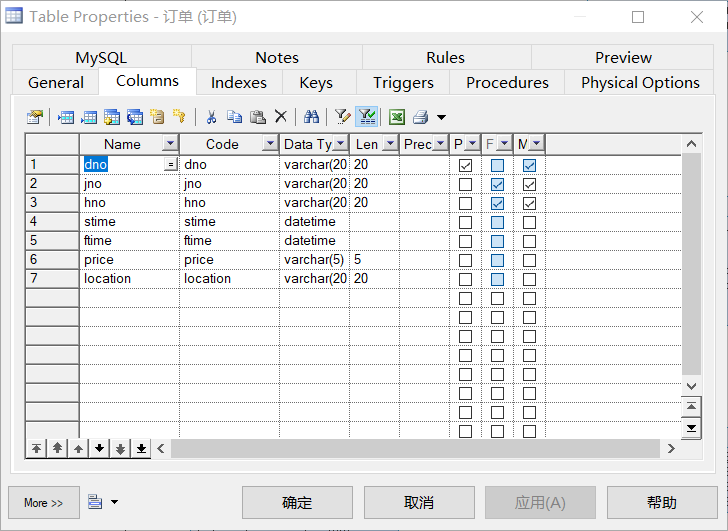
**表2-5**垃圾表



**表2-6**用户表



**表2-7**订单表



## 第五章 物理结构设计

数据库物理设计阶段的任务是根据具体计算机系统(DBMS和硬件等)的特点，为给定的数据库系统确定合理的存储结构和存取方法。所谓的“合理”主要有两个含义：一个是要使设计出的物理数据库占用较少的存储空间，另一个对数据库的操作具有尽可能高的速度。主要体现在后者。

### 1存储结构

确定数据库的存储结构主要指确定数据的存放位置和存储结构，包括确定关系、索引、日志、备份等的存储安排及存储结构，以及确定系统存储参数的配置。

将日志文件和数据库对象(表、索引等)分别放在不同的磁盘，可以改进系统的性能，提高系统的安全性。所以，系统应将日志文件和数据文件存放在不同磁盘上。

### 2数据库的建立

具体SQL代码如下：

**建立**数据库：

create database laji;

**表2-1建立**社区表（department）

CREATE TABLE `department` (

`sno` varchar(20) NOT NULL,

`name` varchar(20) DEFAULT NULL,

`location` varchar(20) DEFAULT NULL,

`manager` varchar(20) DEFAULT NULL,

`telenumber` varchar(20) DEFAULT NULL,

PRIMARY KEY (`sno`),

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-2建立**居民表（people）

CREATE TABLE `people` (

`jno` varchar(20) NOT NULL,

`name` varchar(20) DEFAULT NULL,

`sno` varchar(20) DEFAULT NULL,

`sex` varchar(20) DEFAULT NULL,

`home` varchar(20) DEFAULT NULL,

`age` varchar(20) DEFAULT NULL,

`telenumber` varchar(20) DEFAULT NULL,

`birthday` Data DEFAULT NULL,

PRIMARY KEY (`jno`),

KEY `sno` (`sno`),

CONSTRAINT `profession\_ibfk\_1` FOREIGN KEY (`sno`) REFERENCES `department` (`sno`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-3建立**垃圾回收站表（zhan）

CREATE TABLE `zhan` (

`hsno` varchar(20) NOT NULL,

`name` varchar(20) NOT NULL DEFAULT '',

`manager` varchar(20) DEFAULT NULL,

`region` varchar(20) DEFAULT NULL,

PRIMARY KEY (`hsno`),

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-4建立**垃圾回收员表（worker）

CREATE TABLE `worker` (

`hno` varchar(20) NOT NULL,

`hsno` varchar(20) DEFAULT NULL,

`name` varchar(20) DEFAULT NULL,

`telenumber` varchar(20) DEFAULT NULL,

`sex ` varchar(20) DEFAULT NULL,

` brithday ` Data DEFAULT NULL,

`salary` varchar(20) DEFAULT NULL,

PRIMARY KEY (`hno`),

KEY `hsno` (`hsno`),

CONSTRAINT `teacher\_ibfk\_1` FOREIGN KEY (`hsno`) REFERENCES `department` (`hsno`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-5建立**垃圾表（rubbish）

CREATE TABLE `rubbish` (

`lno` varchar(20) NOT NULL,

`name` varchar(20) DEFAULT NULL,

`ttype` varchar(20) DEFAULT NULL,

`ntype` varchar(20) DEFAULT NULL,

PRIMARY KEY (`lno`),

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-6建立**用户表（user）

CREATE TABLE `user` (

`name` varchar(20) NOT NULL,

`password` varchar(20) DEFAULT NULL,

`type` varchar(20) DEFAULT NULL,

PRIMARY KEY (`name`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

**表2-7建立**订单表（order）

CREATE TABLE `order` (

`dno` varchar(20) NOT NULL,

`jno` varchar(20) NOT NULL,

`hno` varchar(20) NOT NULL,

`stime datetime DEFAULT NULL,

`ftime` datetime DEFAULT NULL,

`credit` float(2,1) DEFAULT NULL,

`price` varchar(20) DEFAULT NULL,

`location` varchar(20) DEFAULT NULL,

PRIMARY KEY (`dno`),

KEY `jno` (`jno`),

KEY `hno` (`hno`),

CONSTRAINT `course\_ibfk\_2` FOREIGN KEY (`hno`) REFERENCES `worker` (`hno`),

CONSTRAINT `course\_ibfk\_1` FOREIGN KEY (`jno`) REFERENCES `people` (`jno`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;