[1引言 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466022)

[1.1编写目的 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466023)

[1.2背景 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466024)

[1.3定义 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466025)

[1.4参考资料 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466026)

[2外部设计 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466027)

[2.1标识符和状态 2](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466028)

[2.2使用它的程序 3](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466029)

[2.3约定 3](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466030)

[2.4专门指导 3](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466031)

[2.5支持软件 3](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466032)

[3结构设计 3](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466033)

[3.1概念结构设计 4](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466034)

[3.2逻辑结构设计 6](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466035)

[3.3物理结构设计 7](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466036)

[4运用设计 8](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466037)

[4.1数据字典设计 8](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466038)

[4.2安全保密设计 1](file:///D:\绘图\customer-order-master\document\数据库设计说明书.doc#_Toc521466039)1

**数据库设计说明书**

# 1引言

## 1.1编写目的

本文档用于项目开发小组成员开发前的阅读。有以下目的：

1. 了解需要使用的技术及规范；
2. 了解项目开发进度计划；

系统的阅读对象为：程序员、数据库设计人员、系统分析员、软件测试人员。

## 1.2背景

软件项目名称：客户订购系统

项目任务提出者：软件工程课程设计指导老师

开发者：万鸿达、陈永涛、寇喆、李鹏飞、李欣宇、史绪豪、凌寒、张嘉奇

用户：管理员，客户

## 1.3定义

本文件中用到的专门术语的定义和外文的首字母组词的原词组列表

如表1：

|  |  |
| --- | --- |
| **原词组** | **含义** |
| JDK | Java开发工具包 |
| IDE | 集成开发环境 |
| IDEA | 一款智能的Java开发IDE |
| Maven | 一款项目依赖管理工具 |
| Mysql | 一款开源的关系型数据库 |
| Tomcat | 一款Java web应用容器 |
| Node.js | 一个基于Chrome V8引擎的js运行环境 |
| Vue | 一款封装好的便于使用的js框架 |
| Spring | 一个开源的J2EE应用程序框架 |
| SpringMVC | 可构建web应用MVC架构模块的Spring框架 |
| Mybatis | 一款开源的基于Java的[持久层](https://baike.baidu.com/item/%E6%8C%81%E4%B9%85%E5%B1%82/3584971?fromModule=lemma_inlink)框架 |
| Mybatis plus | 简化CRUD操作的Mybatis增强工具包 |
| Spring Boot | 简化新Spring应用的初始搭建以及开发过程 |
| Spring Cloud | Spring系列的微服务，利用Spring Boot的开发便利性巧妙地简化了分布式系统基础设施的开发 |
| Redis | 一个key-value[存储系统](https://baike.baidu.com/item/%E5%AD%98%E5%82%A8%E7%B3%BB%E7%BB%9F?fromModule=lemma_inlink) |
| Nacos | 阿里开发的开源服务注册中心、配置中心 |

## 1.4参考资料

## 书本资料：

《软件工程开发与技术》第二版 西安电子科技大学出版社 江开耀主编

《软件体系结构原理、方法与实践》第二版 清华大学出版社

《软件需求工程》 毋国庆等编著 机械工业出版社

Java Api 官方文档

Spring、SpringMVC、Mybatis、MybatisPlus、SpringBoot、SpringCloud官方文档

Redis官方文档

Vue、Node.js官方文档

# 2、外部设计

用户名：root

密码：111111

权限：管理员

有效时间：开发期间

说明：系统正式发布后，可能更改数据库用户/密码，请在统一位置编写数据库连接字符串，在发行前请予以改正。

## 2.2使用它的程序

1.本系统采用 nacos2.2；mysql8.0+；jdk1.8；redis3.2.100；Node.js 18.12.1；

IDEA编辑器,集成maven3.6.1 +tomcat9;

开发环境，因而程序员或者系统分析员，或者任何想要使用此数据库操作的人员，需要安装此开发工具,并建立相同的数据库。

2.数据库中建立的表名对于表的表的各个属性，在下面会有描述。

## 2.3专门指导

对于从事此数据库的生成、此数据库的测试、维护的人员，提出如下参考意见：

数据库的输入统一采用键盘。由于现在没有设置权限，故只要不违背基本的限制，所有的操作都是可以进行的，可是对于本系统需要的数据，却是有数据长度限制的，尤其是有说明的部分，所以测试人员在具体进行数据测试时，要注意数据输入格式。

## 2.5支持软件

1. 开发环境：

nacos2.2；mysql8.0+；jdk1.8；redis3.2.100

Node.js 18.12.1；

IDEA编辑器,集成maven3.6.1 +tomcat9;

1. 选用技术及项目说明：

是一款前后端分离项目，采用分布式架构前端使用vue框架，后端使用SpringCloud微服务开发主要后端技术有

SpringCloud+SpringBoot+MybatisPlus+Redis

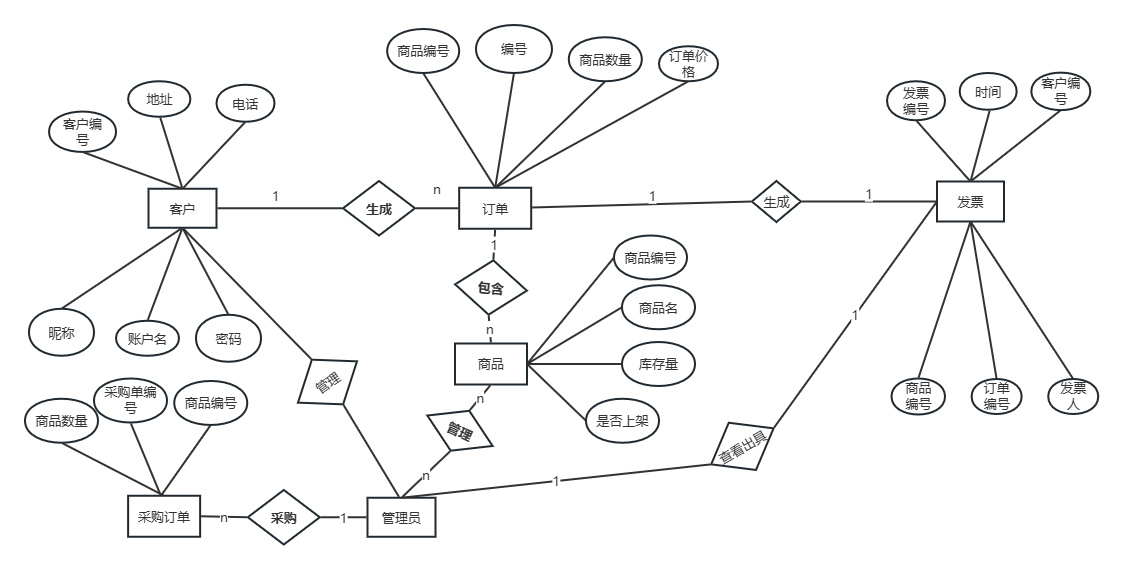
主要前端技术有 vue.js +html+css

# 3结构设计

## 3.1概念结构设计

E-R图也即实体-联系图(Entity Relationship Diagram)，提供了表示实体型、属性和联系的方法，用来描述现实世界的概念模型。

3.1.1 系统的总体E-R图：



## 3.2逻辑结构设计

3.2.1设计思想

E-R转换为关系模型一般遵循的原则:一个实体型转换为一个关系模型.实体的属性就是关系的属性,实体的码就是关系的码.

对于实体型图的联系则有以下不同的情况:

1．一个1:1联系可以转换为一个独立的关系模型,也可以与任意一端对应的关系模型式合并，如果转换为一个独立的关系模式，则与该联系相联的各实体的码以及联系本身的属性均转换为关系的属性，每个实体的码均是该关系的候选码。如果与某一个实体对应关系模式合并，则需要在该关系模式的属性中加如入一个关系模式的码和联系本身的属性。

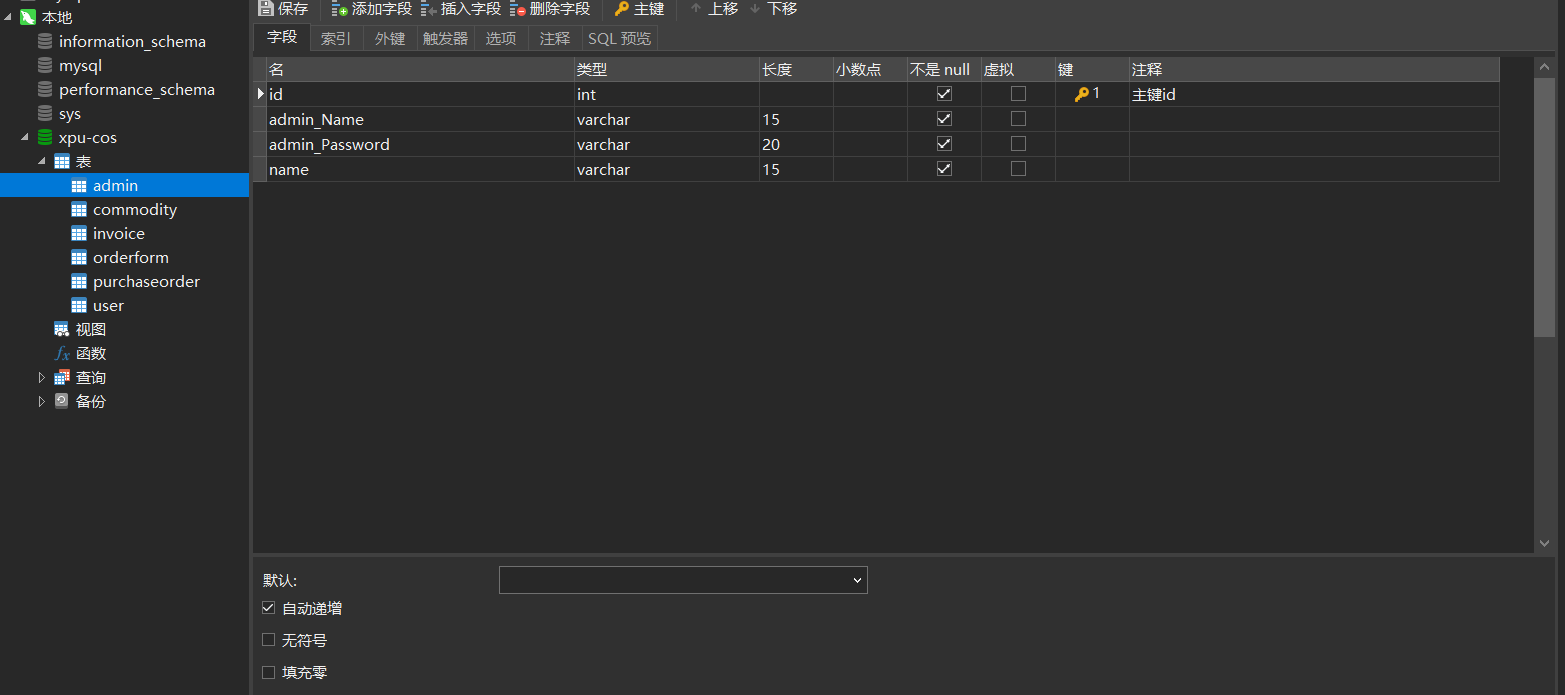
2．一个1:n联系可以转换为一个独立的关系模式，也可以与n端对应的关系模式合并。如果转换模式为一个独立的关系模式，则与该联系相连的各个实体的码以及联系本身的属性均转换为关系的属性，而关系的码为n端实体的码。

3．一个m：n联系转换为一个关系模式，与该联系相连的各实体的码以及联系本身的属性均转换关系的属性，各实体的码组成关系的码或关系码的一个部分。

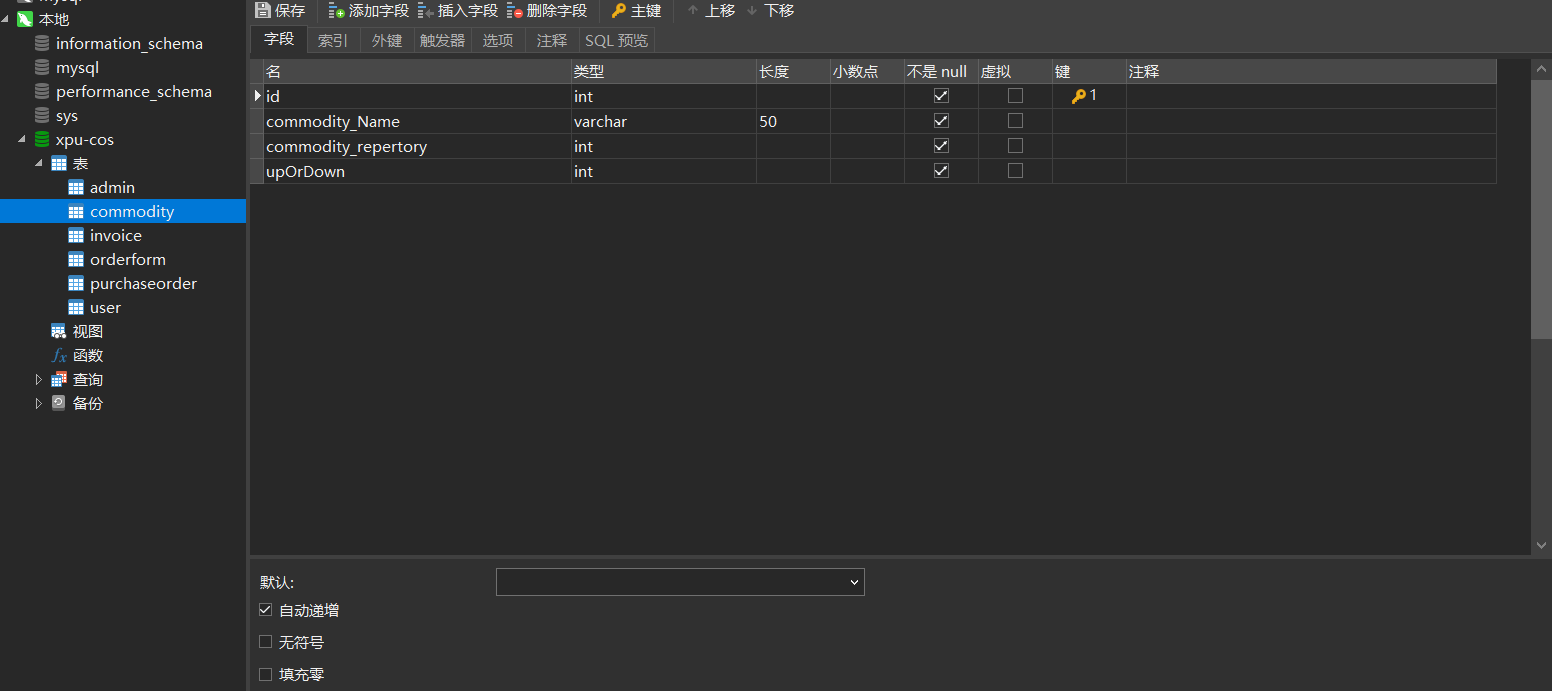
4．3个或3个以上的实体间的一个多元联系可以转换为一个关系模式。与该多元联系相连的各实体的码以及联系本身的属性均转换为关系的属性，各实体的码组成关系的码或关系码的一部分。

**3.2.2数据库表**

（1）管理员表



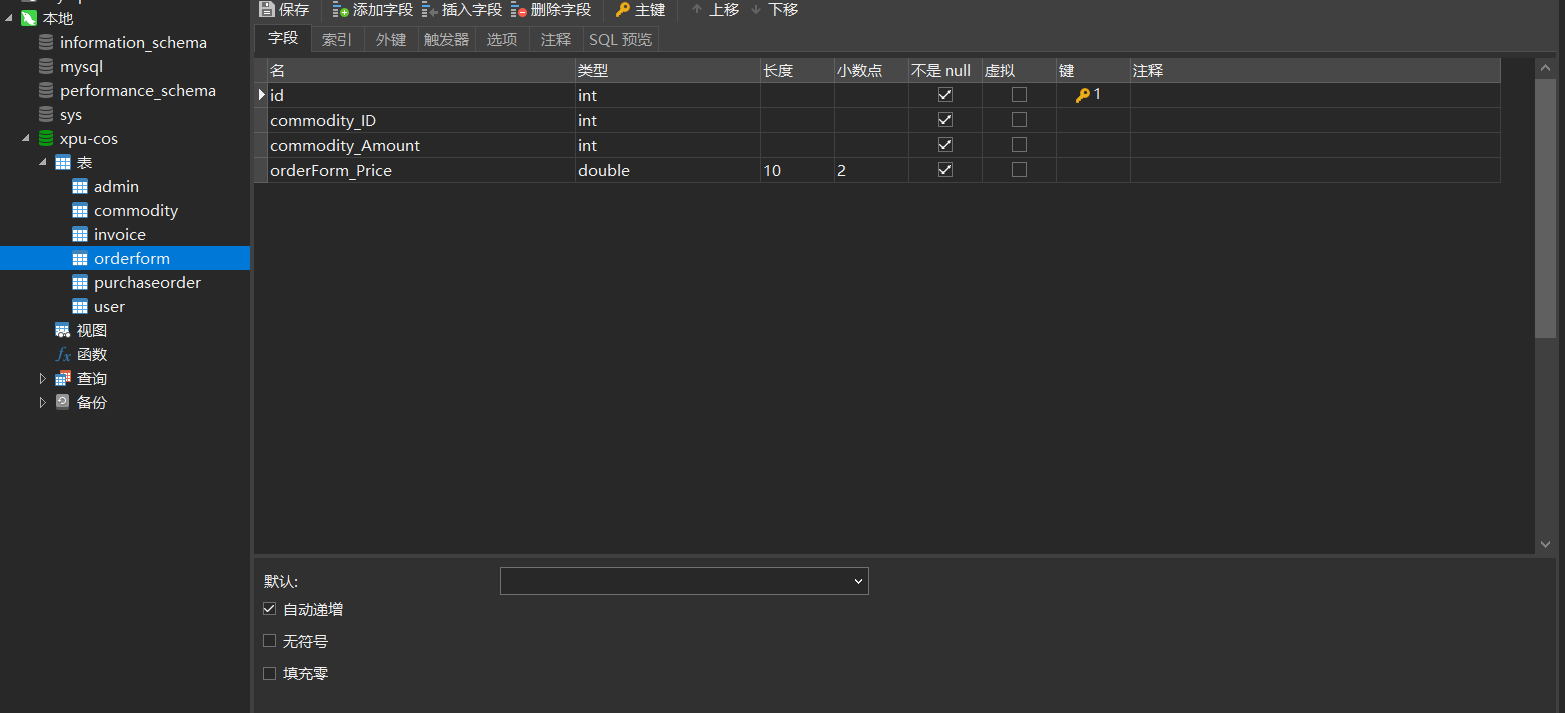
（2）商品表



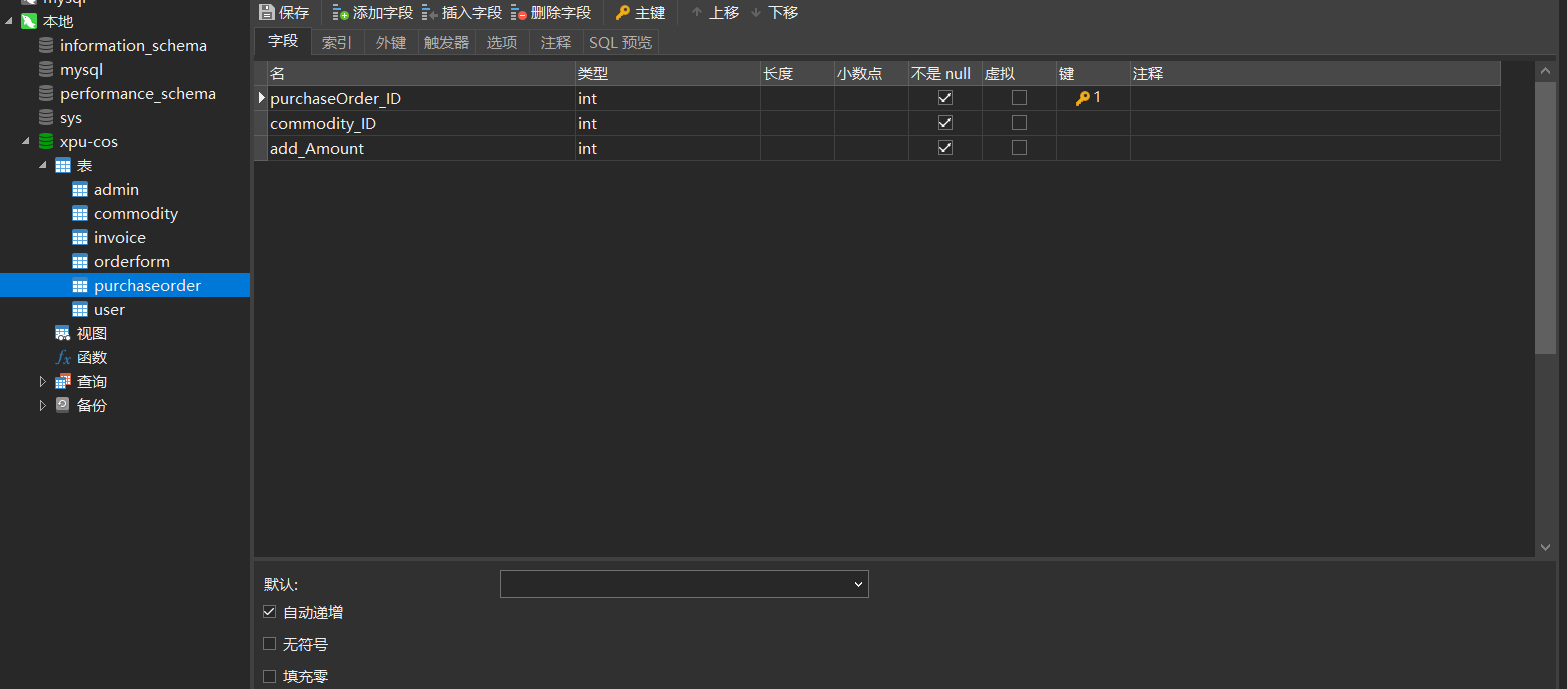
（3）发票表

## 

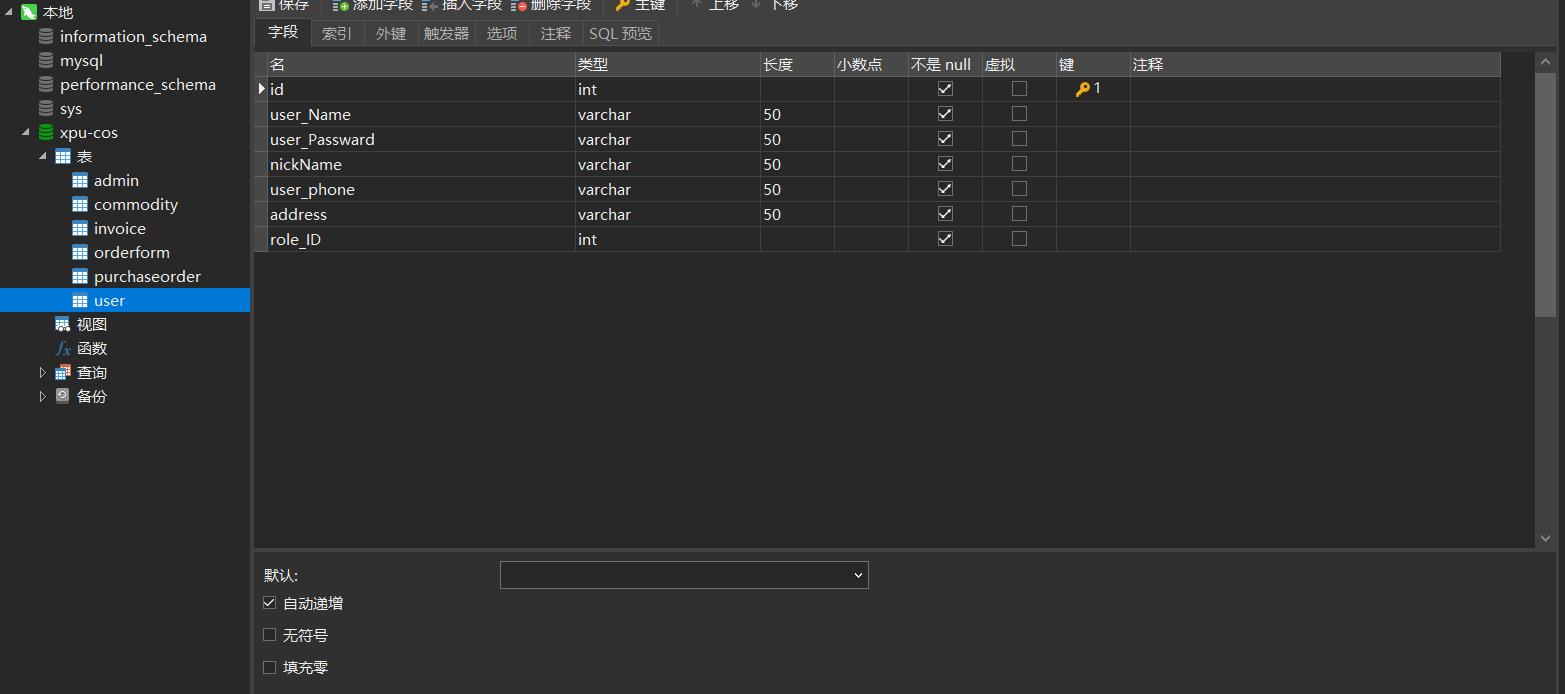
（4）订单信息表



（5）采购订单表



（6）客户表



## 3.3物理结构设计

## 3.4数据结构与程序的关系

系统的数据结构由标准数据库语言SQL生成

# 4运用设计

## 4.1数据字典设计

数据库名称：cusOrder

创建日期：2023年1月4日

字典是关于数据库中数据的描述，它的作用是在软件分析和设计过程中为有关人员提供关于数据描述信息的查询，以保证数据的一致性。

（1）数据结构：数据结构是可以进一步分解的数据包。数据结构由两个或两个以上相互关联的数据元素或者其他数据结构组成的。

根据数据分析，得到数据结构如下：

数据结构名称：用户信息

含义说明：是管理员查看用户也是后台查看用户的重要数据结构，定义了用户的详细相关信息。 组成：用户ID，用户名，用户密码，电话、地址、角色id。

②数据结构名称：商品信息

含义说明：是管理员和用户浏览商品也是后台查看商品的重要数据结构，定义了商品的详细相关信息。

组成：商品ID，商品名，库存，是否上架，商品价格 。

③数据结构名称：商品类别

含义说明：是管理员和用户浏览商品的所属类别也是后台查看商品的重要数据结构，定义了商品类别的详细相关信息。

组成：商品类别ID，类别名称，类别描述。

（2）数据流 ：数据流由一个或一组固定的数据项组成。定义数据流时，不仅说明数据流的名称、组成等，还应指明它的来源、去向和数据流量等。

数据流名称：用户信息

简述：用户登录和注册时产生的信息，用来验证用户信息

数据来源：用户登录和注册

数据流向：用户信息

数据结构：用户名，密码

（3）数据存储 ：数据存储在数据字典中只描述数据的逻辑存储结构，而不涉及它的物理组织。

①数据存储名称：用户信息表

描述：存储用户的详细信息

组成：用户ID，用户名，用户密码，电话、地址、角色id。

②数据存储名称：商品表

描述：存储商品的详细信息

组成：商品ID，商品名，库存，是否上架，商品价格 。

相关处理：查询和展示

（4）处理过程：处理逻辑的定义仅对数据流程图中最底层逻辑加以说明。

① 处理过程名称：查找

输入数据：商品关键字

输出数据：相关商品信息

处理逻辑：输入商品关键字，输出相关所有商品的信息

处理过程名称：登录

输入数据：用户名和密码

输出数据：true或false

处理逻辑：用户输入自己的用户名和密码，在后台进行验证是否正确，正确返回true，不正确返回false。

③ 处理过程名称：注册

输入数据：用户名、密码和性别

输出数据：true或false

处理逻辑：用户输入自己的用户名、密码并且选择性别，在后台写入数据库中，写入成功返回true，不成功返回false。

## 4.2安全保密设计

数据库由专门数据库管理用员对数据库操作，需要注意以下几项安全问题：

访问安全 、网络安全 、传输安全 、备份安全 、数据安全

设置以下安全规则：

1. 每天进行数据备份是保障系统安全的重要手段，保证备份安全

2. 系统设置用户的标识以鉴定是否是合法用户，保证数据安全。

3. 系统对不同的数据设置不同的访问级别，限制访问用户可查询的处理数据类别和内容，保证网络安全。

4. 系统对不同用户设置不同的权限，区分不同的用户，如区分客户和管理员，保证访问安全。

5. 传输皆采用md5 算法传输数据，不采用明文传输，保证传输安全