**天津职业技术师范大学**

**课 程 设 计 任 务 书**

学院 **班级** 学生

课程设计课题：

***人事信息管理系统***

1. 课程设计工作日自 年 月 日至 年 月 日
2. 同组学生：
3. 课程设计任务要求（包括课题来源、类型、目的和意义、基本要求、完成时间、主要参考资料等）：

工具软件介绍：

Nacivat Premium

Mysql 8.0.27

研究目的：

本人事管理系统的研究目的是设计和实现一个高效、可靠的人事信息管理系统，以提高人力资源管理的效率和准确性。通过该系统，可以方便地管理员工的各种信息，包括基本信息、教学信息和科研信息，实现信息的录入、修改、查询和统计等功能，同时满足不同用户角色的权限管理需求。

需求分析：

1. 用户角色需求：
   * 系统用户分为管理员和普通用户两种角色。
   * 管理员可以添加原始的职工信息，对普通管理员和普通用户进行增加、删除、编辑和查询等操作。
   * 普通用户可以查看自己的科研信息，并录入科研信息，待管理员审阅。
2. 员工信息管理需求：
   * 管理员可以添加、修改和删除员工的基本信息，包括编号、姓名、性别、学历、所属部门、毕业院校、健康情况、职称、职务、奖惩等。
   * 管理员可以按照一定条件进行员工信息的查询和统计，以便生成相应的报表或输出结果。
3. 教学信息管理需求：
   * 管理员可以录入教师的教学信息，包括教师编号、姓名、课程编号、课程名称、课程时数、学分和课程性质等。
   * 教师信息和教学信息之间是一对多的关系，一个教师可以对应多个教学信息。
4. 科研信息管理需求：
   * 管理员可以录入教师的科研信息，包括教师编号、研究方向、课题研究情况和专利、论文及专著发表情况等。
   * 教师信息和科研信息之间是一对多的关系，一个教师可以对应多个科研信息。
5. 数据查询和统计需求：
   * 系统需要提供灵活的查询功能，管理员可以根据不同的条件进行员工信息、教学信息和科研信息的查询。
   * 系统需要支持对查询结果进行统计，并能够生成报表或输出结果。
6. 视图和权限管理需求：
   * 系统需要提供视图功能，以便用户可以方便地查阅信息。
   * 系统需要支持不同用户角色的权限管理，确保用户只能访问其权限范围内的信息和功能。
7. 安全和数据完整性需求：
   * 系统需要具备安全性措施，包括用户认证和访问控制，以保护敏感信息的安全。
   * 系统需要通过使用触发器等机制，确保数据的完整性，例如在删除员工信息时进行相关的回滚操作

关系模式：

员工信息表（Employee）

* 员工编号（EmployeeID）（主键）
* 姓名（Name）
* 性别（Gender）
* 学历（Education）
* 所属部门（Department）
* 毕业院校（GraduateSchool）
* 健康情况（HealthStatus）
* 职称（JobTitle）
* 职务（Position）
* 奖惩（RewardsPunishments）

教师信息表（Teacher）

* 教师编号（TeacherID）（主键）
* 姓名（Name）

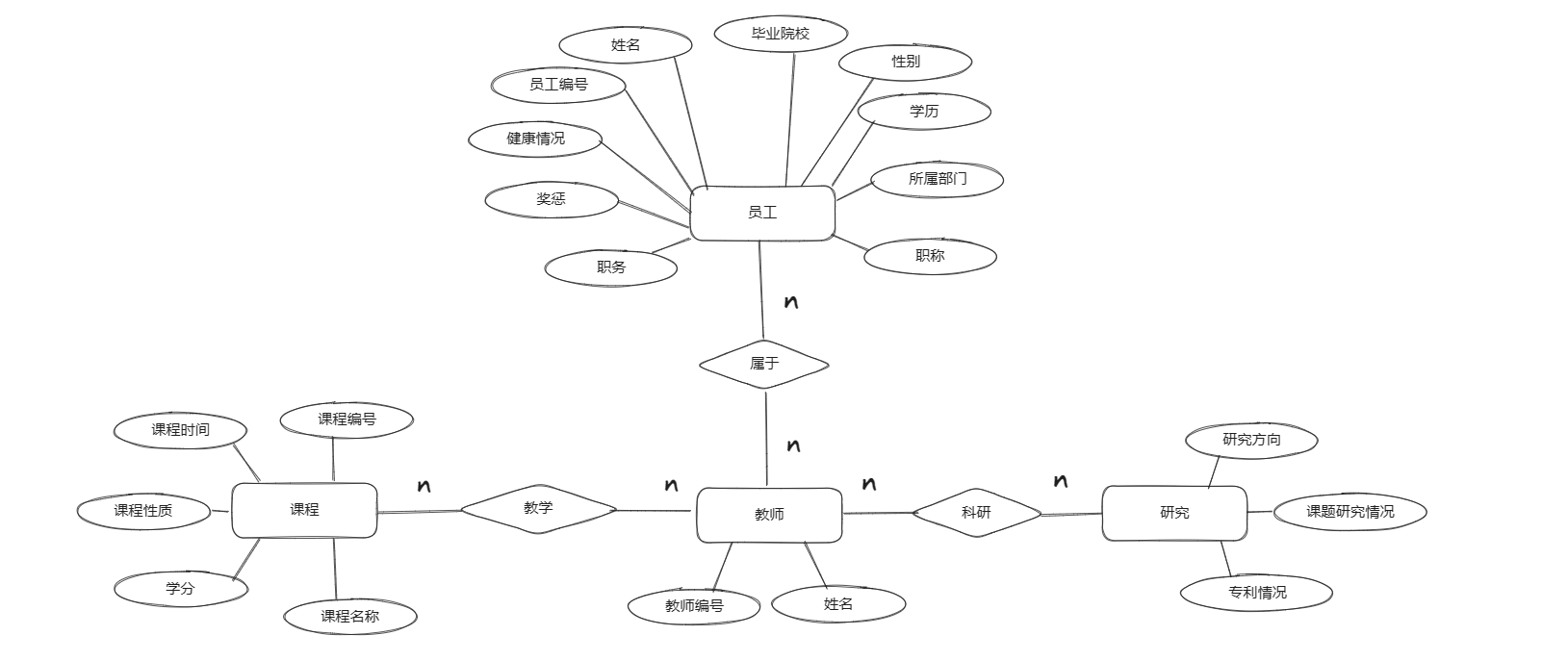
教学信息表（Teaching）

* 教师编号（TeacherID）（外键，引用教师信息表的教师编号）
* 课程编号（CourseID）
* 课程名称（CourseName）
* 课程时数（CourseHours）
* 学分（Credit）
* 课程性质（CourseNature）

科研信息表（Research）

* 教师编号（TeacherID）（外键，引用教师信息表的教师编号）
* 研究方向（ResearchDirection）
* 课题研究情况（ResearchStatus）
* 专利、论文及专著发表情况（PublicationStatus）

ER图：



Mysql代码：

CREATE DATABASE IF NOT EXISTS personnel ;

USE personnel;

-- 创建管理员表

CREATE TABLE Admin (

AdminID INT PRIMARY KEY,

Name VARCHAR(50)

);

-- 创建普通用户表

CREATE TABLE User (

UserID INT PRIMARY KEY,

Name VARCHAR(50)

);

-- 创建教师信息表

CREATE TABLE Teacher (

TeacherID INT PRIMARY KEY,

Name VARCHAR(50)

);

-- 创建员工信息表

CREATE TABLE Employee (

EmployeeID INT PRIMARY KEY,

Name VARCHAR(50),

Gender VARCHAR(10),

Education VARCHAR(50),

Department VARCHAR(50),

GraduateSchool VARCHAR(100),

HealthStatus VARCHAR(50),

JobTitle VARCHAR(50),

Position VARCHAR(50),

RewardsPunishments VARCHAR(100)

);

-- 创建教学信息表

CREATE TABLE Teaching (

TeacherID INT,

CourseID INT,

CourseName VARCHAR(50),

CourseHours INT,

Credit INT,

CourseNature VARCHAR(50),

FOREIGN KEY (TeacherID) REFERENCES Teacher(TeacherID)

);

-- 创建科研信息表

CREATE TABLE Research (

TeacherID INT,

ResearchDirection VARCHAR(100),

ResearchStatus VARCHAR(100),

PublicationStatus VARCHAR(100),

FOREIGN KEY (TeacherID) REFERENCES Teacher(TeacherID)

);

-- 创建审批表

CREATE TABLE Approval (

TeacherID INT,

ResearchID INT,

Status VARCHAR(50),

FOREIGN KEY (TeacherID) REFERENCES Teacher(TeacherID)

);

-- 创建视图以便查阅信息

CREATE VIEW EmployeeView AS

SELECT Employee.EmployeeID, Employee.Name, Employee.Gender, Employee.Education, Employee.Department, Employee.GraduateSchool, Employee.HealthStatus, Employee.JobTitle, Employee.Position, Employee.RewardsPunishments, Teaching.CourseID, Teaching.CourseName, Teaching.CourseHours, Teaching.Credit, Teaching.CourseNature, Research.ResearchDirection, Research.ResearchStatus, Research.PublicationStatus

FROM Employee

LEFT JOIN Teaching ON Employee.EmployeeID = Teaching.TeacherID

LEFT JOIN Research ON Employee.EmployeeID = Research.TeacherID;

-- 创建触发器以处理不当操作时的回滚事务等

CREATE TRIGGER RollbackTrigger

BEFORE DELETE ON Employee

FOR EACH ROW

BEGIN

-- 进行相关回滚操作

END;

**指导教师签字： 系主任签字：**