

# 作业 2

SA20225085 朱志儒

1、已知有关系模式  $R\{A, B, C, D, E\}$ ， $R$  上的一个函数依赖集如下：

$$F = \{A \rightarrow BC, B \rightarrow CE, A \rightarrow B, AB \rightarrow C, AC \rightarrow DE, E \rightarrow A\}$$

(1) 求出  $F$  的最小函数依赖集

解：由  $A \rightarrow BC$  可知  $A \rightarrow B, A \rightarrow C$ ，由  $B \rightarrow CE$  可知  $B \rightarrow C, B \rightarrow E$ ，由  $AC \rightarrow DE$  可知  $AC \rightarrow D, AC \rightarrow E$ ，则  $A \rightarrow C$  冗余， $AB \rightarrow C$  中的  $B$  是冗余的， $AC \rightarrow E$  中的  $C$  是冗余的， $AC \rightarrow D$  中的  $C$  是冗余的，消除冗余的函数依赖后  $F = \{A \rightarrow B, B \rightarrow C, B \rightarrow E, A \rightarrow D, E \rightarrow A\}$ 。

(2) 求  $R$  的候选码

解：由  $F = \{A \rightarrow B, B \rightarrow C, B \rightarrow E, A \rightarrow D, E \rightarrow A\}$  可知  $A \rightarrow \{A, B, C, D, E\}, E \rightarrow \{A, B, C, D, E\}, B \rightarrow \{A, B, C, D, E\}$ ，所以  $A, B, E$  是  $R$  的候选码。

(3)  $R$  属于第几范式？为什么？

解： $R$  属于 BCNF，因为对于  $F$  中的任何函数依赖，其左部必定含有候选键。

(4) 请将  $R$  无损连接并且保持函数依赖地分解到 3NF

解：有最小函数依赖集  $F = \{A \rightarrow B, B \rightarrow C, B \rightarrow E, A \rightarrow D, E \rightarrow A\}$  可得， $R_1(A, B, D), R_2(B, C, E), R_3(E, A)$ ，由  $R$  的主码为  $A$ ，于是

$$p = q \cup \{R(X)\} = \{R_1(A, B, D), R_2(B, C, E), R_3(E, A), R(A)\}$$

因为  $\{A\}$  是  $R_1$  的子集，所以从  $p$  中去掉  $R(A)$ ，故最终结果为

$$p = \{R_1(A, B, D), R_2(B, C, E), R_3(E, A)\}$$

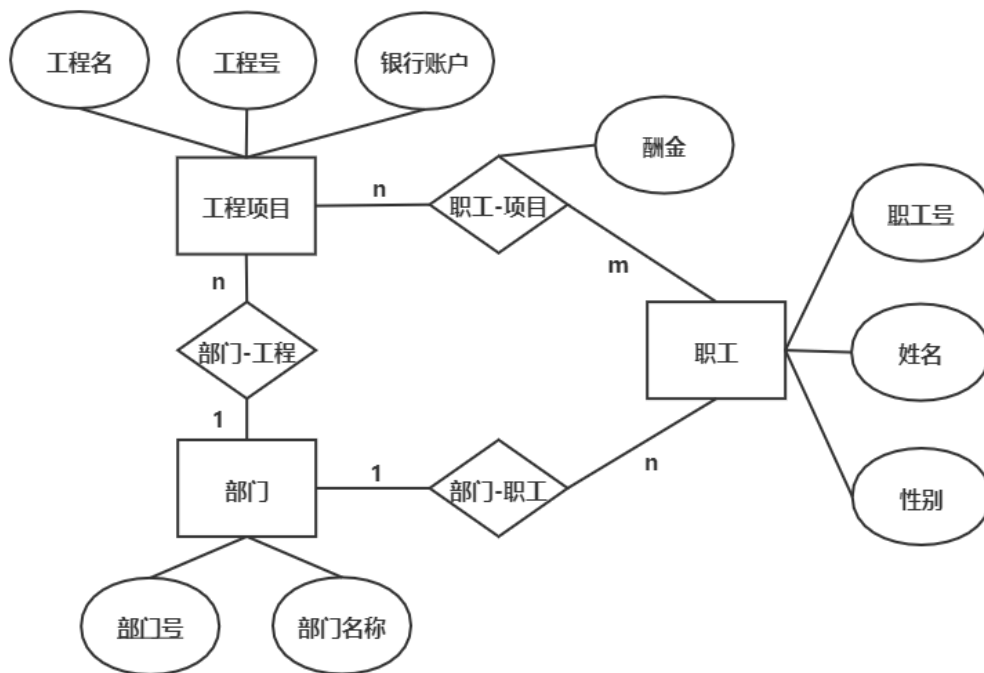
2、假设某公司要开发一个信息管理系统。根据调研，获得该公司的业务规则如下(所有实体的码可以按日常生活中的一般规则处理)：

- (1) 公司下设几个部门，如技术部、财务部、市场部等；
- (2) 每个部门承担多个工程项目，每个工程项目属于一个部门。每个部门为其承担的每个项目分别开设独立的银行账户。
- (3) 每个部门有多名职工，每一名职工只能属于一个部门。
- (4) 一个职工可能参与多个工程项目，且每个工程项目有多名职工参与施工。根据职工在工程项目中完成的情况发放酬金。

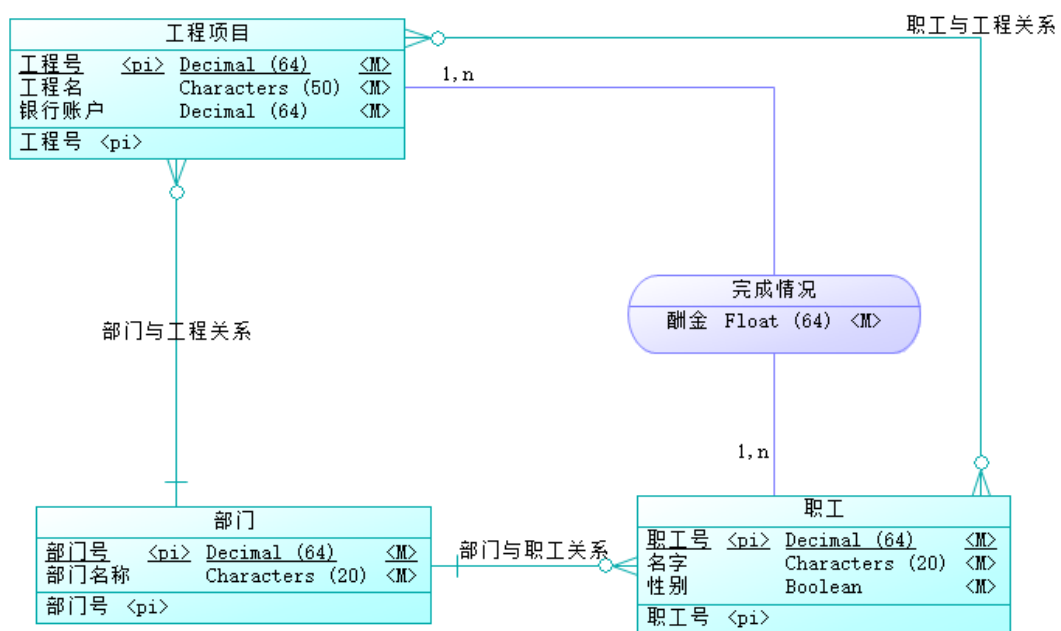
(5) 工程项目有工程号、工程名两个属性；部门有部门号、部门名称两个属性；职工有职工号、姓名、性别属性。

根据以上描述，请画出相应的 ER 模型（使用传统的 ER 建模符号），并将它转换为关系数据库模式。注：关系模式和属性名称均使用中文名称。

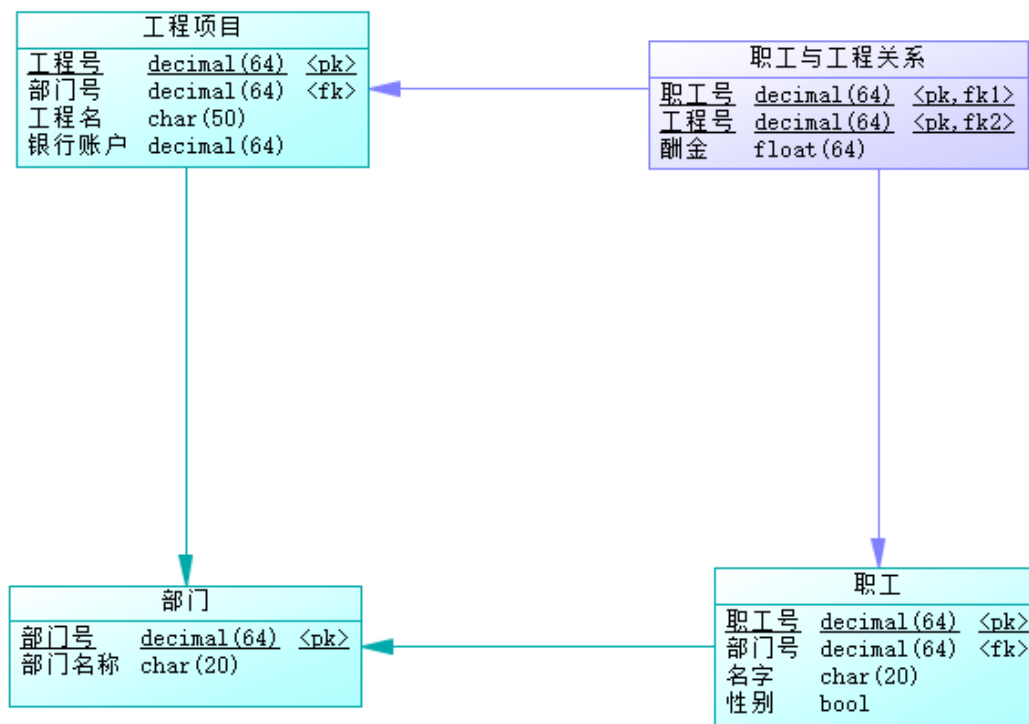
ER 模型：



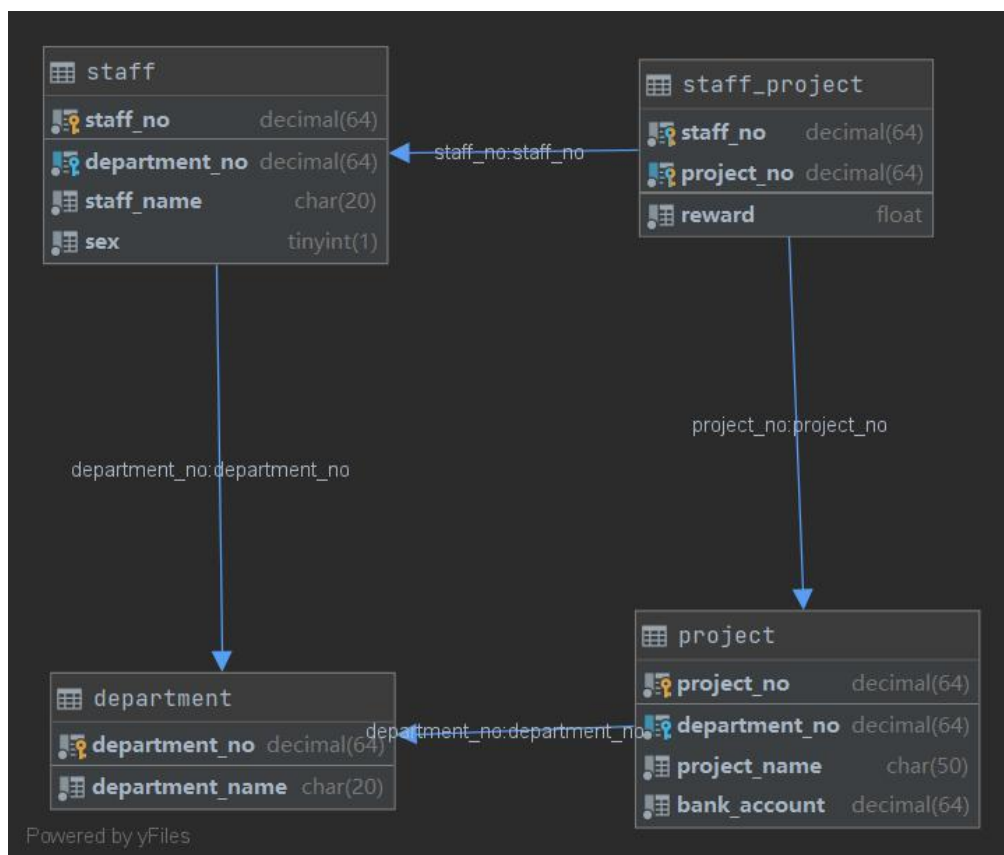
概念设计模型：



物理设计模型：



MySQL 关系数据库模式:



SQL 源码如下:

```
/*=====*/
```

```
/* DBMS name:      MySQL 5.0                                */
```

```
/* Created on:     2021/4/15 16:55:14                        */
```

```
/*=====*/
```

```
drop table if exists department;
```

```
drop table if exists project;
```

```
drop table if exists staff;
```

```
drop table if exists staff_project;
```

```
/*=====*/
```

```
/* Table: department                                         */
```

```
/*=====*/
```

```
create table department
```

```
(  
    department_no      decimal(64) not null,  
    department_name     char(20) not null,  
    primary key (department_no)  
);
```

```
/*=====*/
```

```
/* Table: project                                           */
```

```
/*=====*/
```

```
create table project
```

```
(  
    project_no         decimal(64) not null,
```

```
    department_no      decimal(64) not null,  
    project_name       char(50) not null,  
    bank_account       decimal(64) not null,  
    primary key (project_no)  
);
```

```
/*=====*/
```

```
/* Table: staff */
```

```
/*=====*/
```

```
create table staff
```

```
(  
    staff_no          decimal(64) not null,  
    department_no     decimal(64) not null,  
    staff_name        char(20) not null,  
    sex               bool not null,  
    primary key (staff_no)  
);
```

```
/*=====*/
```

```
/* Table: staff_project */
```

```
/*=====*/
```

```
create table staff_project
```

```
(  
    staff_no          decimal(64) not null,  
    project_no        decimal(64) not null,  
    reward            float not null,  
    primary key (staff_no, project_no)  
);
```

```
alter table project add constraint FK_department_project foreign key (department_no)
```

references department (department\_no) on delete restrict on update restrict;

alter table staff add constraint FK\_department\_staff foreign key (department\_no)

references department (department\_no) on delete restrict on update restrict;

alter table staff\_project add constraint FK\_completion foreign key (staff\_no)

references staff (staff\_no) on delete restrict on update restrict;

alter table staff\_project add constraint FK\_completion2 foreign key (project\_no)

references project (project\_no) on delete restrict on update restrict;