**数据库实训报告**

22920212204392 黄勖

## 实验要求

根据实训课程内容，假设C银行因为实际业务需要添加一个新对象为理财经理，其与数据库中其他对象存在着以下关系：一个理财经理可以服务多个客户，一个客户只能有一个理财经理。

## 目标

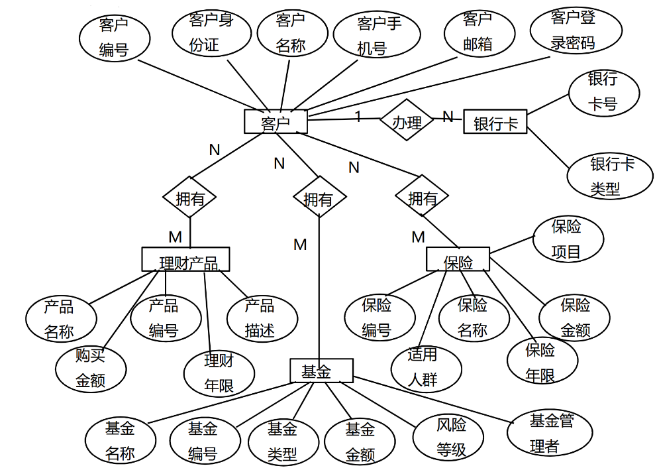
1. 根据C银行新添加的对象关系，画出修改后的关系模式和ER图。
2. 根据修改后的ER图创建新的表格。
3. 插入一些生成数据并设计查询验证新添加的对象和关系。

根据上述目标，提交相应的图表及文字说明。

## 解答

1. 修改关系模式和ER图

原有的ER图：



原有的关系模式：

客户（客户编号、客户名称、客户邮箱，客户身份证，客户手机号，客户登录密码）

银行卡（银行卡号，银行卡类型）

理财产品（产品名称，产品编号，产品描述，购买金额，理财年限）

保险（保险名称，保险编号，保险金额，适用人群，保险年限，保障项目）

基金（基金名称，基金编号，基金类型，基金金额，风险等级，基金管理者）

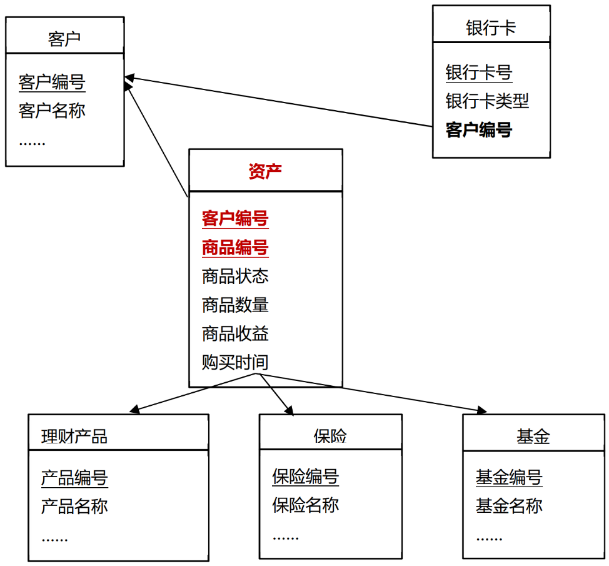
对象之间的关系：

 一个客户可以办理多张银行卡

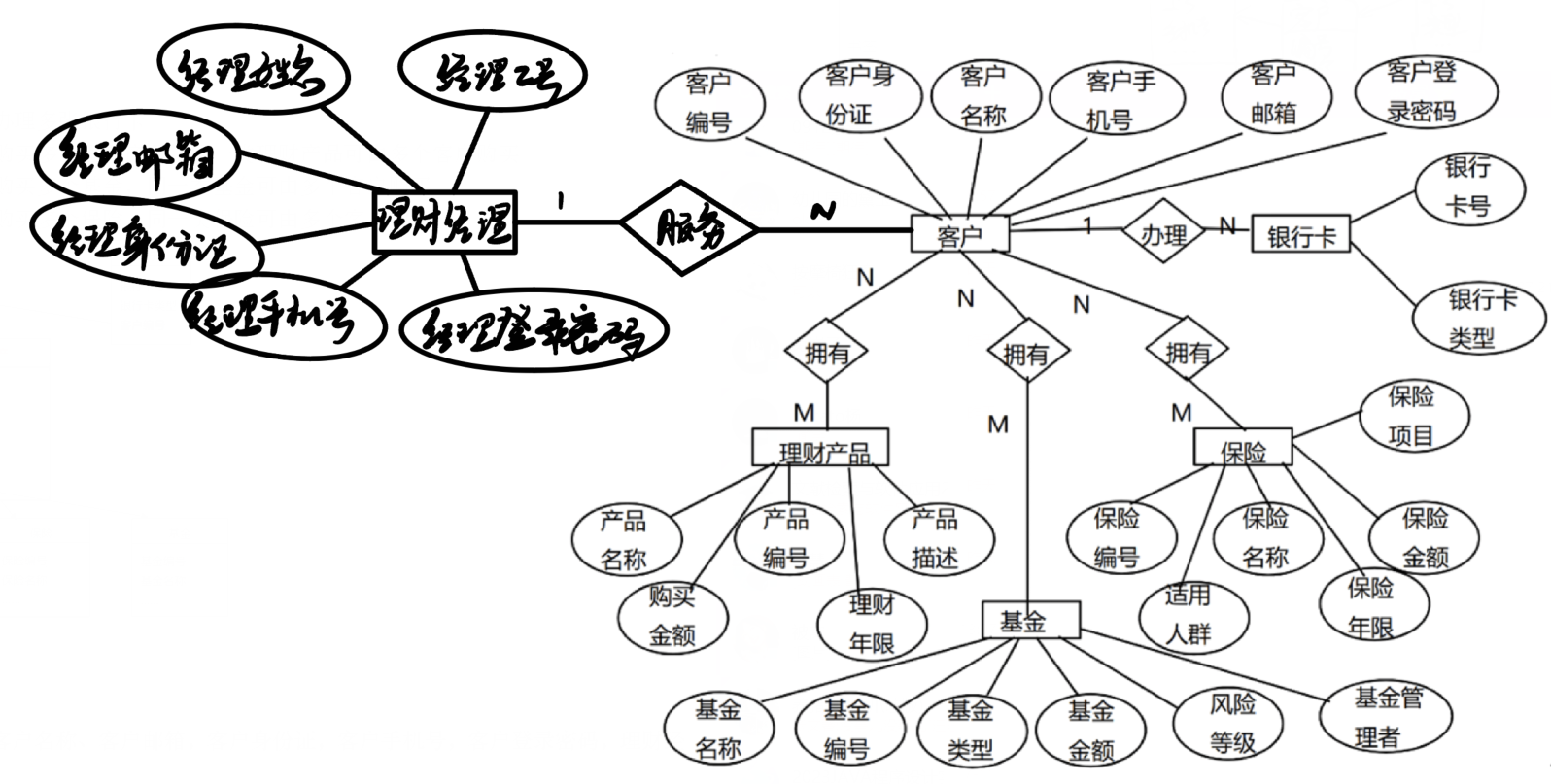
 一个客户可以购买多个理财产品，同一类理财产品可由多个客户购买

 一个客户可以购买多个基金，同一类基金可由多个客户购买

 一个客户可以购买多个保险，同一类保险可由多个客户购买



**修改后的ER图：**



**修改后的关系模式：**

客户（客户编号、客户名称、客户邮箱，客户身份证，客户手机号，客户登录密码，理财经

理）

银行卡（银行卡号，银行卡类型）

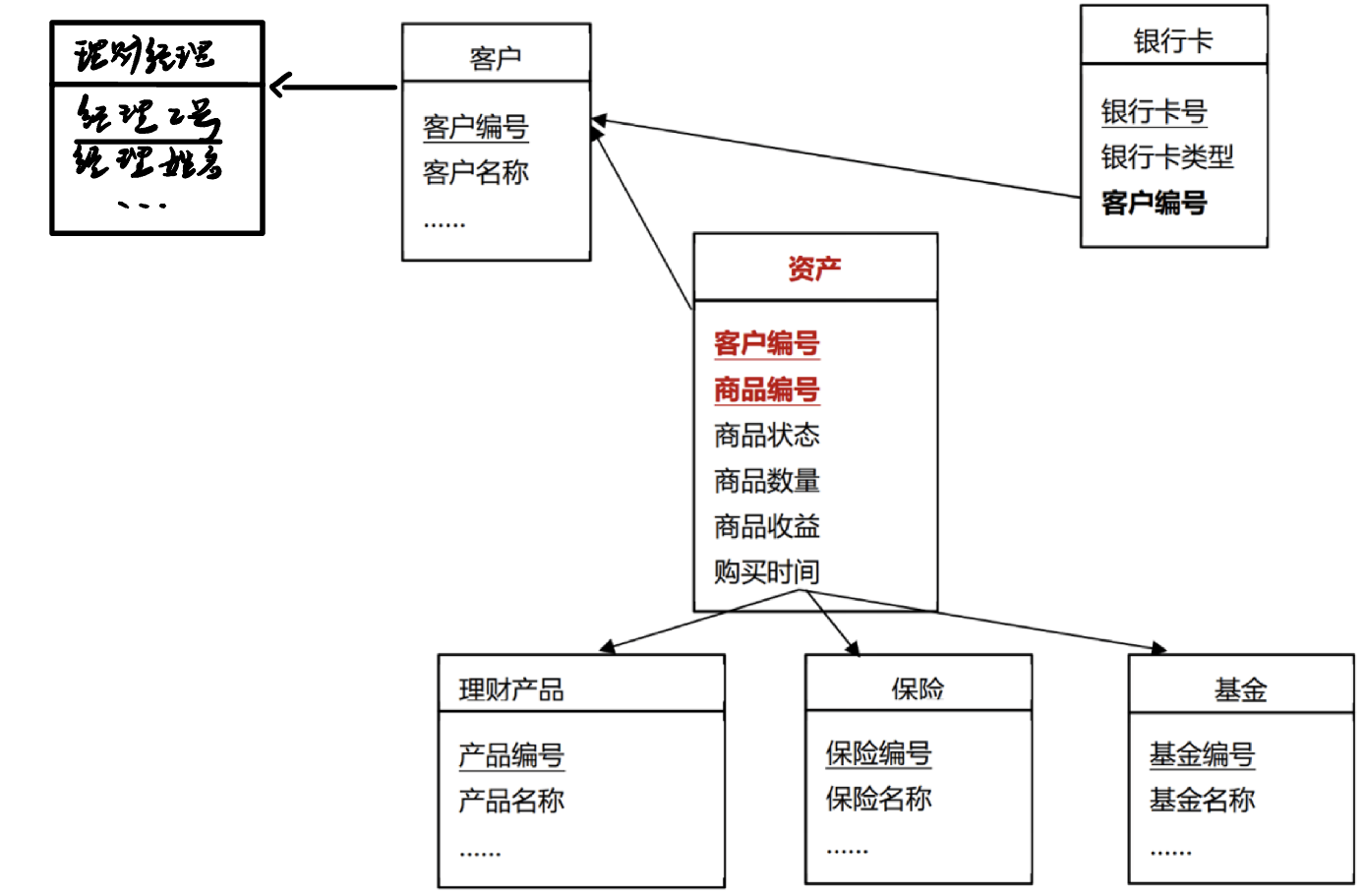
理财产品（产品名称，产品编号，产品描述，购买金额，理财年限）

保险（保险名称，保险编号，保险金额，适用人群，保险年限，保障项目）

基金（基金名称，基金编号，基金类型，基金金额，风险等级，基金管理者）

理财经理（经理工号，经理姓名，经理邮箱，经理身份证，经理手机号，经理登录密码）

ps: 一个理财经理可以服务多个客户，一个客户只能有一个理财经理。



2. 根据修改后的 ER 图创建新的表格。

理财经理表：

CREATE TABLE manager

(

c\_manager\_id INT PRIMARY KEY,

c\_name VARCHAR(100) NOT NULL,

c\_mail CHAR(30) UNIQUE,

c\_id\_card CHAR(20) UNIQUE NOT NULL,

c\_phone CHAR(20) UNIQUE NOT NULL,

c\_password CHAR(20) NOT NULL

);

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 约束 | 说明 |
| c\_manager\_id | INTEGER | PRIMARY KEY | 经理工号 |
| c\_name | VARCHAR(100) | NOT NULL | 经理名称 |
| c\_mail | CHAR(30) | UNIQUE | 经理邮箱 |
| c\_id\_card | CHAR(20) | UNIQUE NOT NULL | 经理身份证 |
| c\_phone | CHAR(20) | UNIQUE NOT NULL | 经理手机号 |
| c\_password | CHAR(20) | NOT NULL | 经理登录密码 |

客户表：

CREATE TABLE client

(

c\_id INT PRIMARY KEY,

c\_name VARCHAR(100) NOT NULL,

c\_mail CHAR(30) UNIQUE,

c\_id\_card CHAR(20) UNIQUE NOT NULL,

c\_phone CHAR(20) UNIQUE NOT NULL,

c\_password CHAR(20) NOT NULL,

c\_manager\_id CHAR(10),

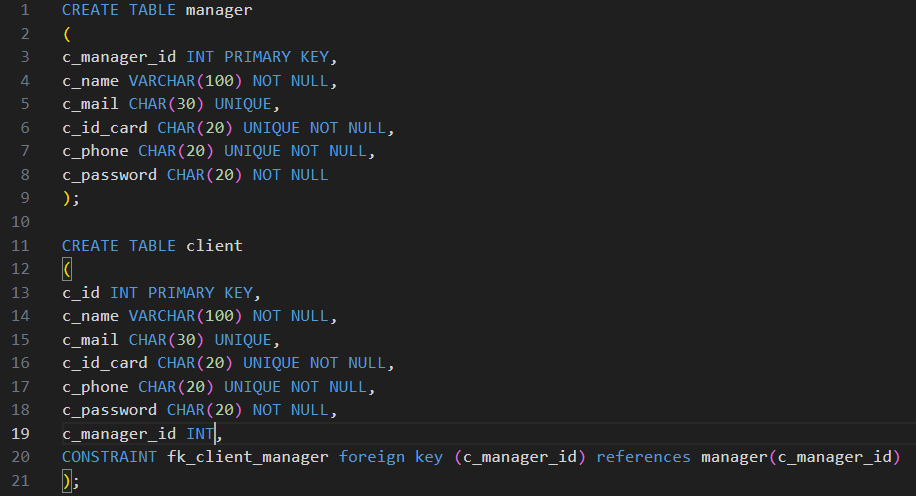
CONSTRAINT fk\_client\_manager foreign key (c\_manager\_id) references manager(c\_manager\_id)

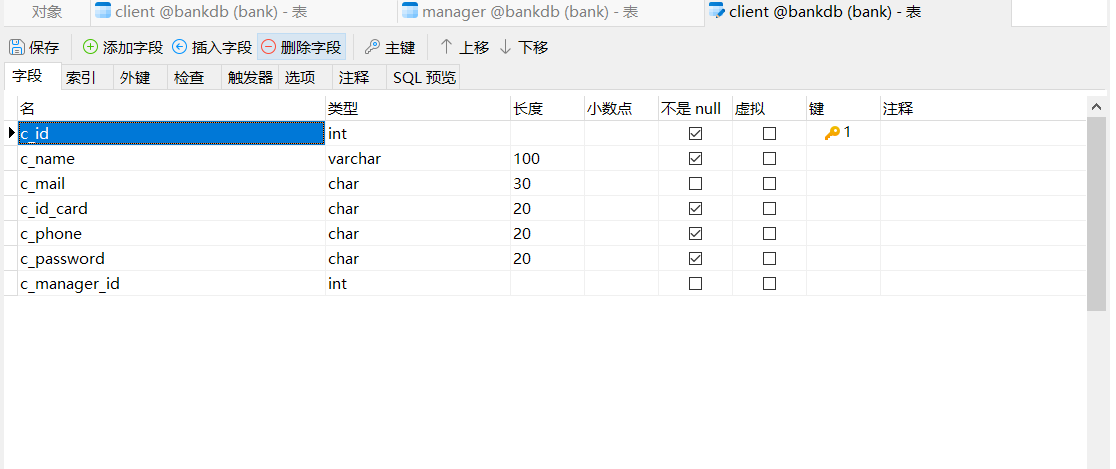
);

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 约束 | 说明 |
| c\_id | INTEGER | PRIMARY KEY | 客户编码 |
| c\_name | VARCHAR(100) | NOT NULL | 客户名称 |
| c\_mail | CHAR(30) | UNIQUE | 客户邮箱 |
| c\_id\_card | CHAR(20) | UNIQUE NOT NULL | 客户身份证 |
| c\_phone | CHAR(20) | UNIQUE NOT NULL | 客户手机号 |
| c\_password | CHAR(20) | NOT NULL | 客户登录密码 |
| c\_manager\_id | INTEGER | FOREIGN KEY | 理财经理工号 |

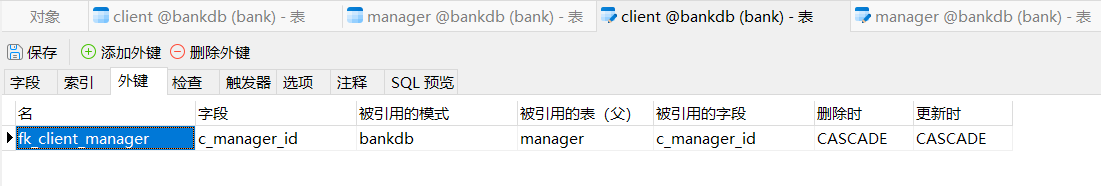
1. 插入一些生成数据并设计查询验证新添加的对象和关系。

这里使用可视化数据库管理软件Navicat展示FusionDB数据库建立的过程和结果：

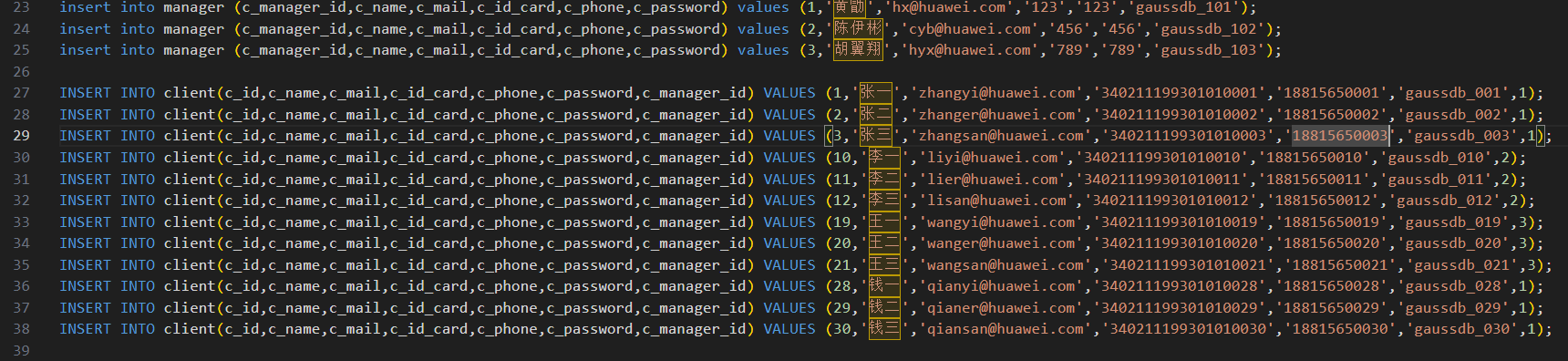




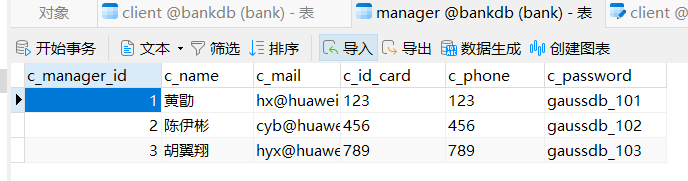




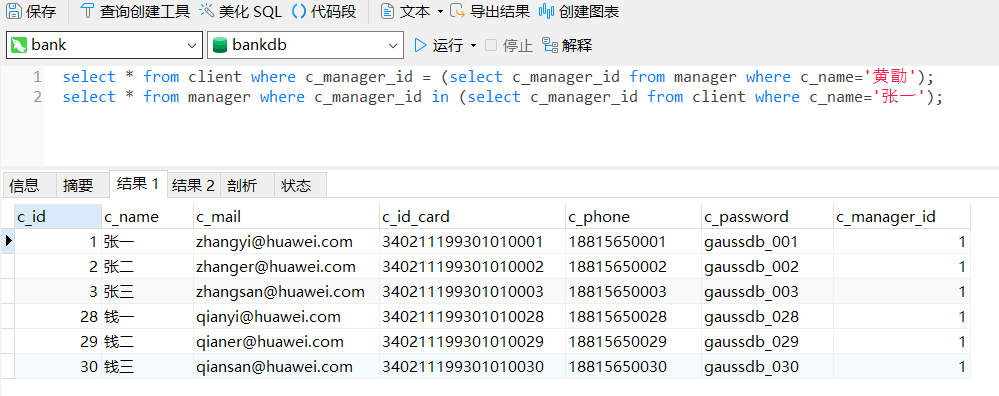
插入数据：



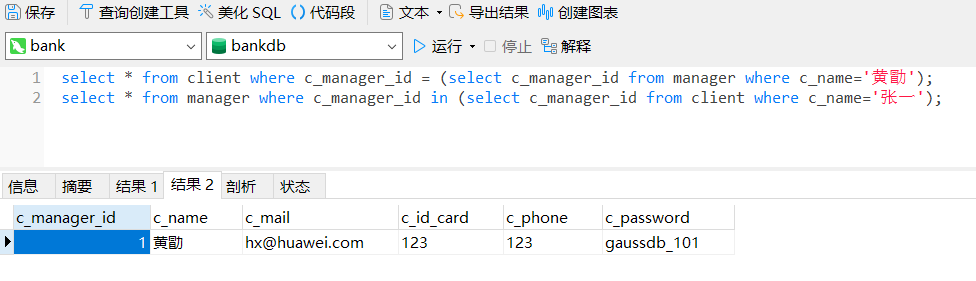




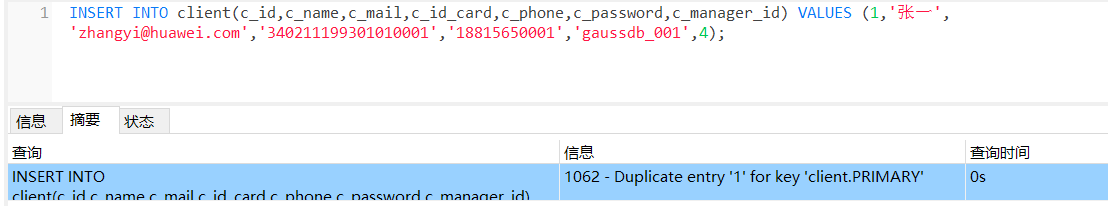
查询：



正确查找到理财经理黄勖的客户。

正确查找到张一的理财经理黄勖。

插入数据：



违反主码约束，无法插入数据。



违反外码约束（没有这个理财经理id），无法插入数据。

**至此完成本周实验内容。**