**实验1：数据库设计与应用开发**

姓名：王丁子睿 学号：1183710211 班级：1803104

1. **数据库设计**
   1. **需求分析**

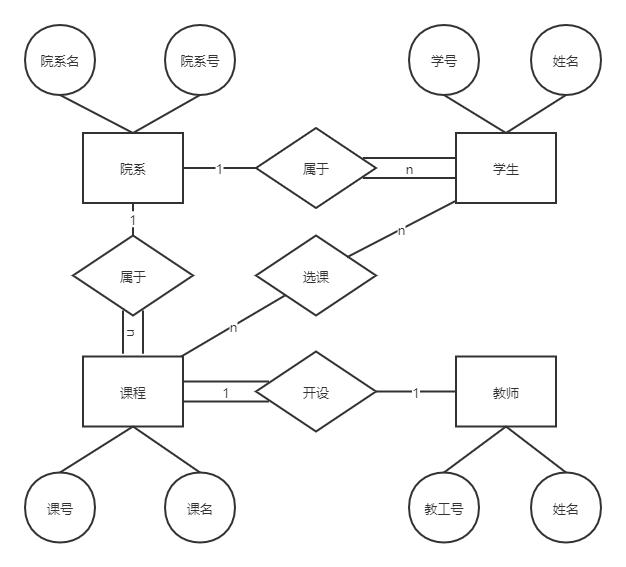
本次实验面向选课系统进行设计，分别有院系、教师、学生、课程四个对象，每个对象都有一个ID，作为唯一标识（主关键字）；一个名称，便于用户的使用。除此之外，每个学生、课程都有所属院系，用来限制其（被）选课的范围。

上述中ID为INT类型，名称为TEXT类型。

数据给出输入后，将会将其转换为对应的格式，并在不符合对应格式的要求时报错。

* 1. **概念数据库设计**

设计的ER图如下：



* 1. **逻辑数据库设计**

对于学生属于院系，将院系号并入学生的关系中。

对于课程属于院系，将院系号并入课程的关系中。

对于教师开设课程，将教工号并入课程的关系中。

对于学生选择课程，新建关系记录对应的学号和课号。

综上所述，得到以下五个关系：

|  |  |
| --- | --- |
| 关系名 | 关系实例 |
| 院系(COLLEGE) |  |
| 教师(TEACHER) |  |
| 学生(STUDENT) |  |
| 课程(COURSE) |  |
| 选课单(SCHEDULE) |  |

* 1. **物理数据库设计**

除课单外，其他关系的主键均为ID。

学生关系中，所属院系为对应院系关系的外键。

课程关系中，所属院系为对应院系关系的外键。

课单关系中，选课学生为对应学生关系的外键，被选课程为对应课程关系的外键。

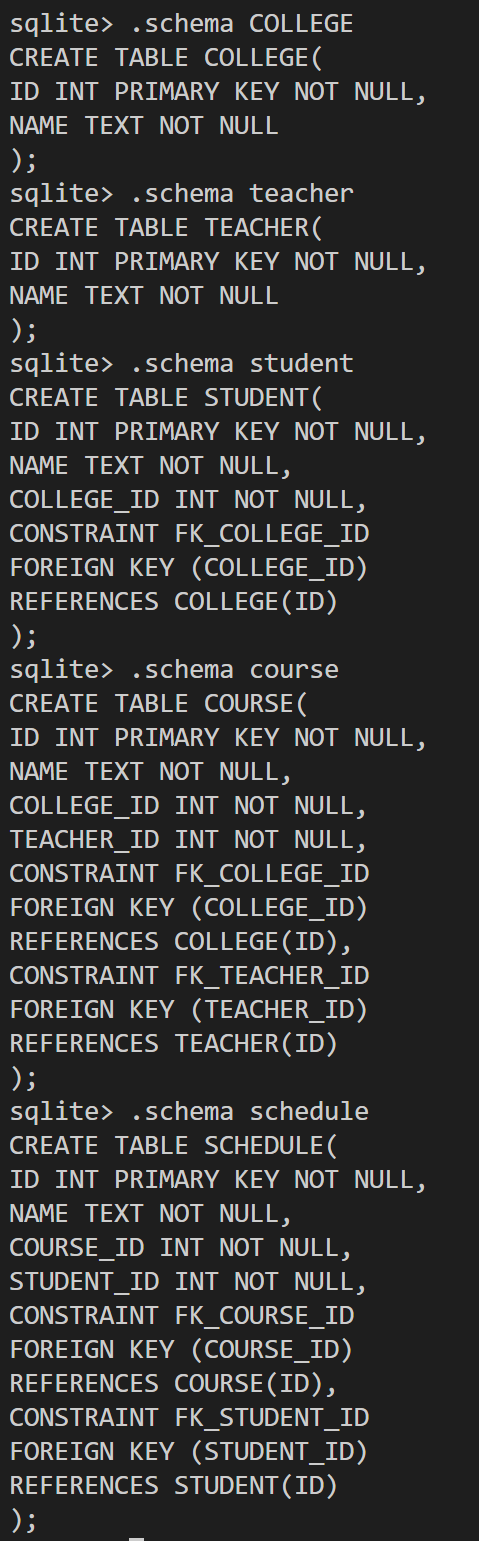
* 1. **数据库建立**

所用DBMS为SQLite。

所用数据库可见data/data.db。

* + 1. 概念模式

各关系的逻辑结构和特征如下所示：



* + 1. 外模式

提供三种外模式。

1. 管理员模式：可以向COLLEGE中添加新的院系、教师和学生。
2. 教师模式：可以向COURSE中添加新的课程，由于添加课程需要院系的信息（新课程的所属院系需已经存在），故为此创建视图，定义为”CREATE VIEW college\_info AS SELECT id, name FROM college”。
3. 学生模式：可以从COURSE中选择课程，由于选课需要课程的信息，故为此创建视图，定义为”CREATE VIEW course\_info AS SELECT id, name, college\_id FROM course”。
4. **数据库应用开发**

数据库开发采用Python的sqlite3包。

前端开发采用Python的PyQt包。

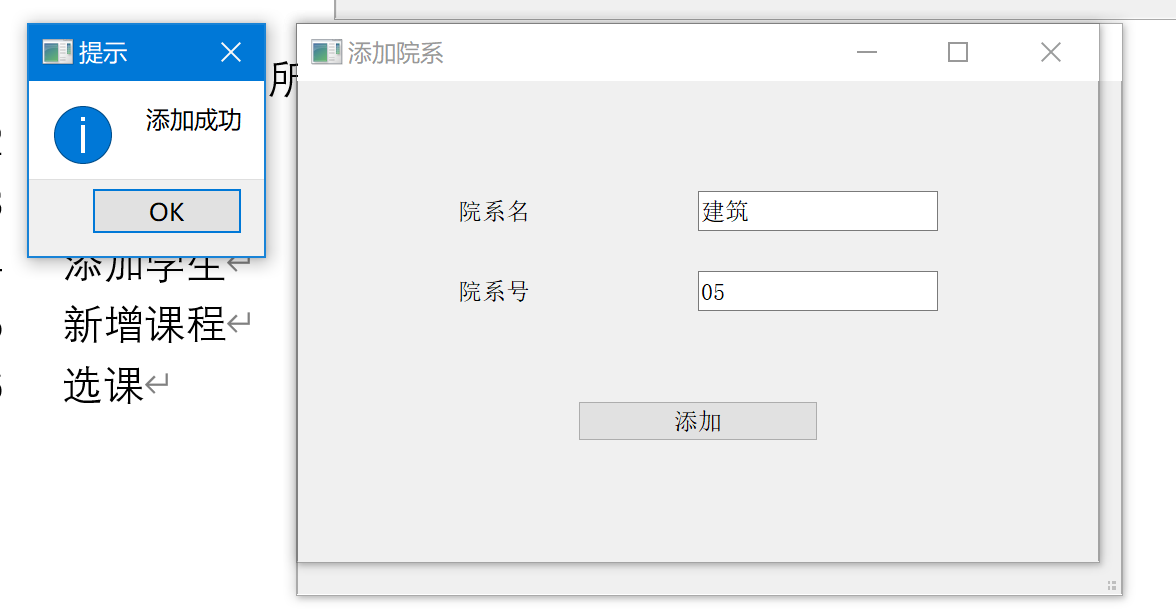
具体代码见src文件夹。

1. **实验结果**
   1. **主界面**

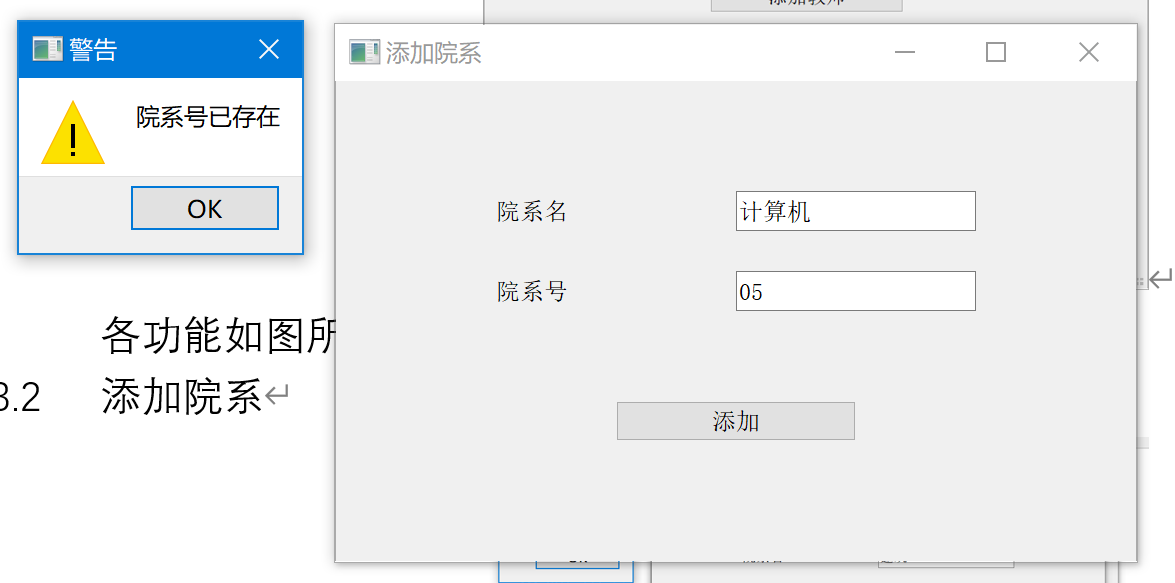
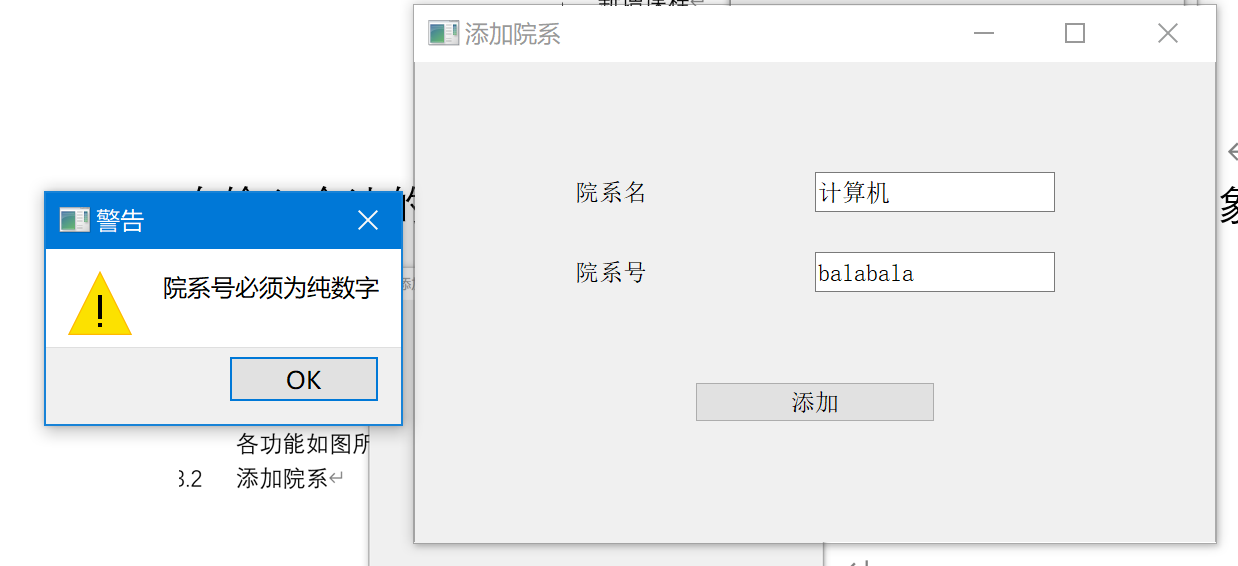


各功能如图所示。

* 1. **添加院系**



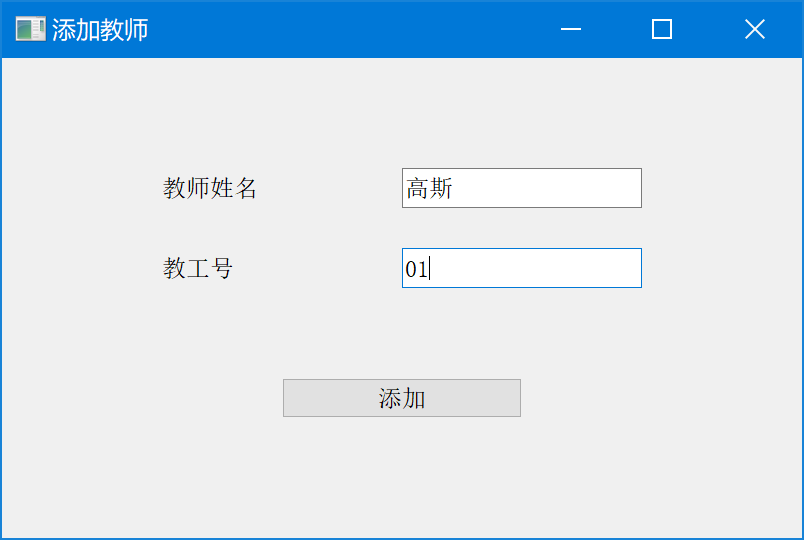
在输入合法的院系名和院系号后，会向数据库添加该条对象，并提示添加成功。

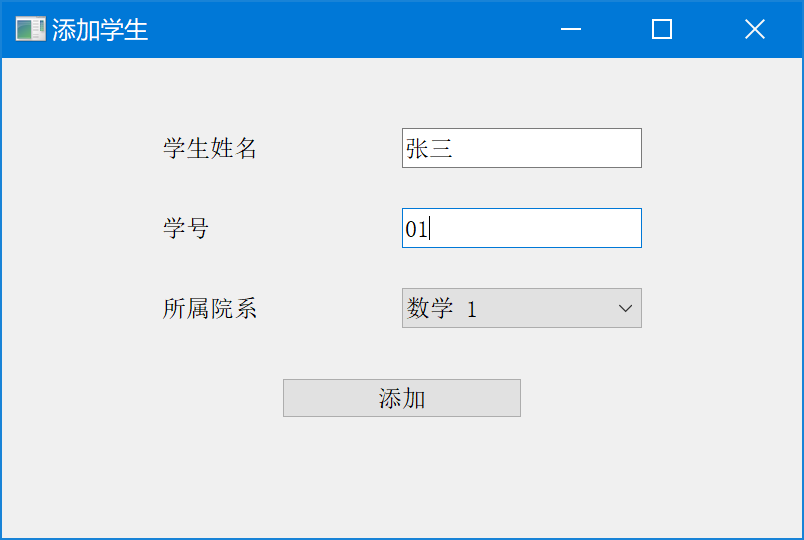
若输入不合法，则会提示相应的错误信息。

以下各操作与之类似，不再赘述。

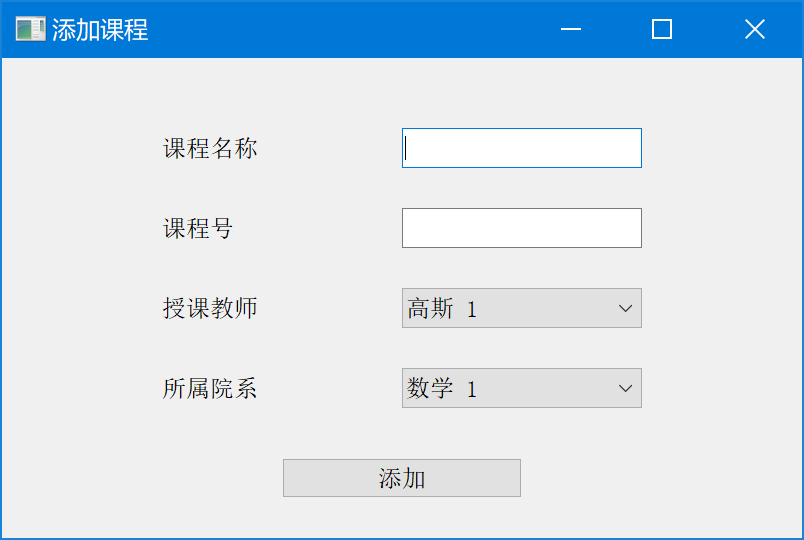
* 1. **添加教师**



* 1. **添加学生**



* 1. **新增课程**



* 1. **选课**

