《数据库系统原理》课程设计

系统设计报告

题目名称：图书馆空间预约系统

学号及姓名： 14061002 王雅洁

14061176 黄秋宇

2016 年 10 月 17 日

组内同学承担任务说明

|  |  |  |
| --- | --- | --- |
|  | 王雅洁 | 黄秋宇 |
| 系统设计阶段 | 需求描述  需求分析  E-R图初稿 | 数据流图  数据元素表  数据库逻辑模式设计 |
| 系统实现阶段 |  |  |
| 系统报告撰写 |  |  |

目录

[一、 需求分析 3](#_Toc465362793)

[(一) 需求描述 3](#_Toc465362794)

[(二) 数据流图 4](#_Toc465362795)

[(三) 数据元素表 6](#_Toc465362796)

[二、 数据库概念模式设计 9](#_Toc465362797)

[(一) 系统初步 E-R图 9](#_Toc465362798)

[(二) 系统基本E-R图 13](#_Toc465362799)

[三、 数据库逻辑模式设计 14](#_Toc465362800)

[(一) 数据库关系模式 14](#_Toc465362801)

[(二) 关系模式范式等级的判定与规范化 16](#_Toc465362802)

[(三) 数据库设计优化 19](#_Toc465362803)

[四、 最终版修改说明 19](#_Toc465362804)

## 需求分析

### 需求描述

#### 实际需求

图书馆空间管理当前管理混乱，主要存在以下两个问题

1. 一些人在图书馆占完座之后，人却消失了好几个小时，导致想自习的同学没有空位置，而现有的位置已被别人占了但处于空闲状态，导致没有被真正的利用，造成资源的浪费。
2. 社团及班级活动经常需要用到讨论室，但由于对资源分配不透明，导致某些时间段同时有多个社团需要讨论室，而图书馆的讨论室资源有限，从而出现竞争现象，并且由于讨论室资源的分配不均，在另一些时间段没有社团或班级用时，讨论室处于空闲状态，从而出现资源浪费的问题。

我们设计的图书馆空间预约系统正是针对这两个问题，明确系统定位，从而很好的解决了问题。

#### 系统概述

图书馆空间预约系统是经过严格设计、逻辑清晰、功能明确的针对在校学生、教职工、团体组织的数据库系统。数据库系统规范了预约、离席等有关空间管理的流程，通过操纵学生、团体、单人座位、讨论室等实体，达到了图书馆空间利用效率的最大化，同时对数据库支持增删改查等各种操作。

#### 主要功能

* 面向学生的功能
  + 注册

未在图书馆空间预约系统中完成注册的学生可以通过选择自助注册来完成验证。

* + 修改个人信息

学生可以通过个人信息页修改自己的联系电话等信息。

* + 查询座位

学生登录后，可以通过图形化界面直接查看各个座位情况，座位情况包括：已被预约、空闲。

* + 预约座位

学生登录后，根据自己的需求选择一个状态为“空闲”的座位，在正确填写到位时间后即可完成预约。

* + 取消预约

对于当前时间还未超过到位时间的座位预约订单，学生可以选择取消预约，避免导致违约。

* 面向团体的功能
  + 注册

未在图书馆空间预约系统中完成注册的团体可以通过选择自助注册来完成验证。

* + 修改个人信息

团体负责人可以利用团体账号登录，通过团体信息页修改负责人的联系电话等信息，即团体的联系电话等。

* + 查询讨论室

团体负责人登录后，首先选择需要查看的讨论室，确认选择可以查看该讨论室的当前状态以及对应空闲时间段。

* + 预约讨论室

团体负责人登录后，可以根据团队的会议需求和讨论室已有的时间表，合理安排会议时间，正确填写到位时间并填写预约时长后即可完成预约。

* + 取消预约

对于当前时间还未超过到位时间的讨论室预约订单，团体负责人可以选择取消预约，避免导致违约。

* 面向管理员的功能
  + 注册

未在图书馆空间预约系统中完成注册的管理员可以通过选择自助注册来完成验证。

* + 管理统计表

管理员登录后可以查看统计表，管理所有记录，其中统计表包括预约记录编号，违约记录编号，进出记录编号，学号，团体编号，可以根据学号或团体编号进行排序，方便查询。

* 系统监督功能
  + 有关学生的监督

1. 本校学生预定座位成功后，需根据预约时填写的到位时间，刷卡通过进出控制系统完成座位确认，如果半个小时内未到达，则认定为“迟到”，该座位的状态重新调整为“未预约”，同时该学生将被记一次违约。
   * 有关团队的监督
2. 团体负责人预定讨论室成功后，需按照预约时填写的到位时间，刷卡通过进出控制系统完成讨论室确认。如果超过30分钟仍未到达，则取消该此预约，并且为该团体记一次违约。
3. 团体负责人应合理安排会议时间，如果会议时长超出了预约时长，即刷卡通过进出控制系统时，已超出预约的时间段，则该团体被记一次违约。

* 违约处理
  + 若学生违约达到3次，则7天内无法登录该系统。
  + 若团体违约达到2次，则2个月内无法登录该系统。

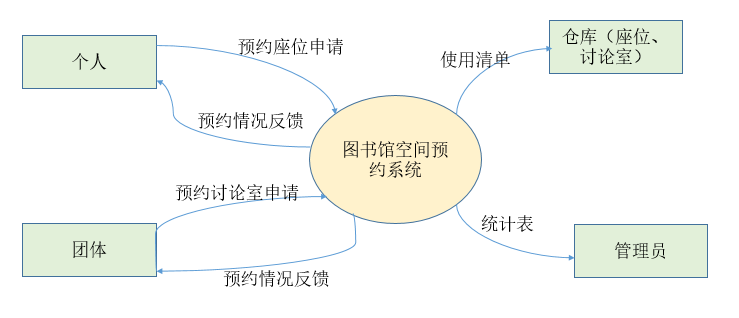
#### 安全性及完整性约束

系统通过分类，定义了学生、团体、管理员的权限，从而保证重要信息不会得到非法的修改，在管理的层面上进行了安全性约束。

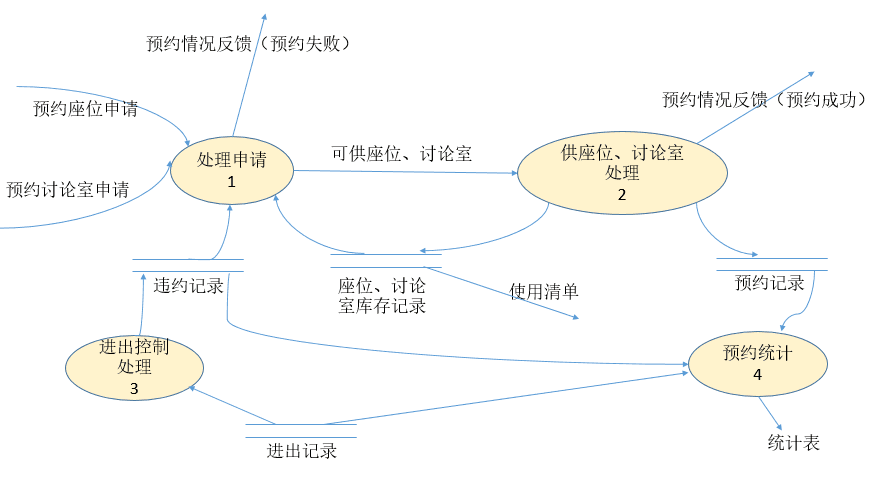
此外，系统在设计过程中按照实体完整性和参照完整性条件的约束逐步深入设计，满足了完整性的约束条件。

### 数据流图

#### 顶层数据流图

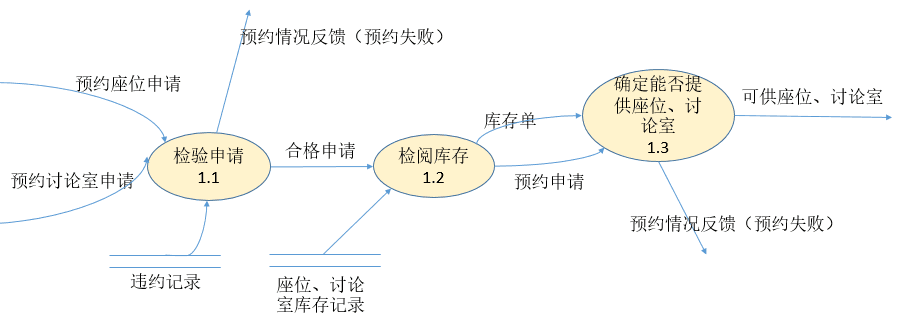


#### 0层数据流图

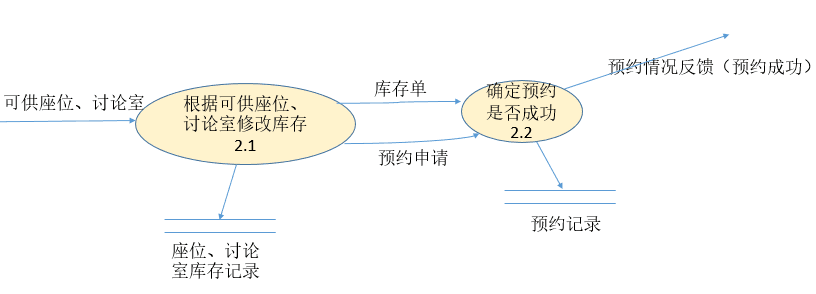


#### 1层数据流图

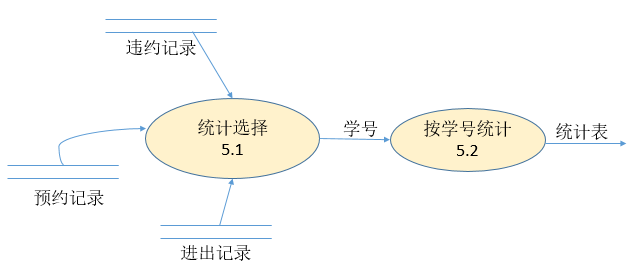
（1）处理申请数据流图



（2）供座位、讨论室处理数据流图



（3）预约统计数据流图



### 数据元素表

#### 数据项

##### 本校学生信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E1 | stu\_id | int | 10 | 学号 |
| E2 | stu\_name | string | 10 | 姓名 |
| E3 | stu\_sex | string | 2 | 性别 |
| E4 | stu\_psw | string | 20 | 登录密码 |
| E5 | stu\_pho | string | 20 | 联系电话 |
| E6 | stu\_dep | int | 2 | 所属系号 |
| E7 | stu\_exp | datetime | 20 | 学生身份有效期 |

##### 本校团体信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E8 | team\_id | int | 6 | 团体编号 |
| E9 | team\_pro | string | 20 | 团体性质 |
| E10 | team\_psw | string | 20 | 登录密码 |
| E11 | team\_pho | string | 20 | 联系电话 |
| E12 | team\_cre | string | 20 | 成立时间 |

##### 管理员信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E13 | staff\_id | int | 8 | 管理员工号 |
| E14 | staff\_name | string | 10 | 姓名 |
| E15 | staff\_sex | string | 2 | 性别 |
| E16 | staff\_psw | string | 20 | 登录密码 |
| E17 | staff\_pho | string | 20 | 联系电话 |
| E18 | staff\_exp | datetime | 20 | 管理员身份有效期 |

##### 单人座位信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E19 | seat\_id | int | 10 | 座位编号 |
| E20 | seat\_sta | string | 10 | 当前状态 |

##### 讨论室信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E21 | room\_id | int | 5 | 房间号 |
| E22 | room\_sta | string | 10 | 当前状态 |
| E23 | room\_sch | string | 20 | 空闲时间段编号 |

##### 违约记录表信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E24 | bre\_rec | string | 20 | 违约记录编号 |
| E1 | stu\_id | int | 10 | 学号 |
| E8 | team\_id | int | 6 | 团体编号 |
| E25 | bre\_time | datetime | 20 | 违约时间 |
| E26 | bre\_rea | string | 20 | 违约理由 |

##### 进出控制系统信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E27 | inout\_rec | string | 20 | 进出记录编号 |
| E28 | mec\_id | int | 2 | 机器编号 |
| E1 | stu\_id | int | 10 | 学号 |
| E8 | team\_id | int | 6 | 团体编号 |
| E29 | in\_time | datetime | 20 | 进入时间 |
| E30 | out\_time | datetime | 20 | 离开时间 |

##### 座位预约申请信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E31 | con\_rec | string | 30 | 预约记录编号 |
| E1 | stu\_id | int | 10 | 学号 |
| E19 | seat\_id | int | 10 | 座位编号 |
| E32 | arr\_time | datetime | 20 | 到位时间 |

##### 讨论室预约申请信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E31 | con\_rec | string | 30 | 预约记录编号 |
| E8 | team\_id | int | 6 | 团体编号 |
| E21 | room\_id | int | 5 | 房间号 |
| E33 | start\_time | datetime | 20 | 开始时间 |
| E34 | duration | int | 2 | 预约时长 |

##### 统计表信息相关项

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据类型 | 长度 | 备注 |
| E31 | con\_rec | string | 30 | 预约记录编号 |
| E24 | bre\_rec | string | 20 | 违约记录编号 |
| E27 | inout\_rec | string | 20 | 进出记录编号 |
| E1 | stu\_id | int | 10 | 学号 |
| E8 | team\_id | int | 6 | 团体编号 |

#### 数据结构

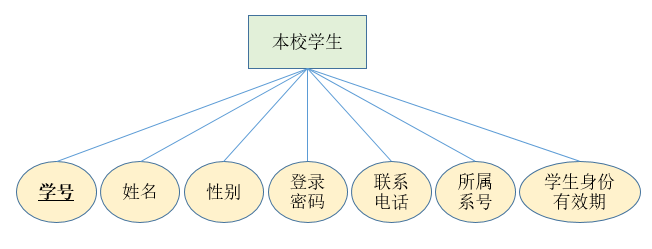
|  |  |  |
| --- | --- | --- |
| 编号 | 数据结构名 | 所用数据项 |
| T1 | 本校学生信息表 | E1-E7 |
| T2 | 本校团体信息表 | E8-E12 |
| T3 | 管理员信息表 | E13-E18 |
| T4 | 单人座位信息表 | E19-E20 |
| T5 | 讨论室信息表 | E21-E23 |
| T6 | 违约记录表 | E24-E26、E1、E8 |
| T7 | 进出信息表 | E27-E30、E1、E8 |
| T8 | 座位预约申请表 | E1、E19、E31、E32 |
| T9 | 讨论室预约申请表 | E8、E21、E31、E33、E34 |
| T10 | 统计信息表 | E31、E24、E27、E1、E8 |

## 数据库概念模式设计

### 系统初步 E-R图

#### 实体属性信息

##### 本校学生：



##### 本校团体：



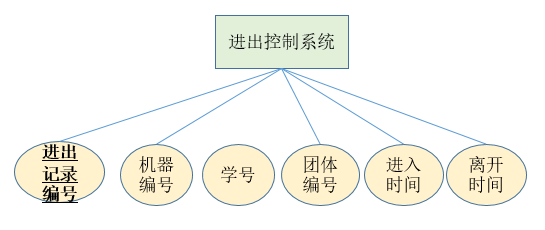
##### 管理员：



##### 违约记录：



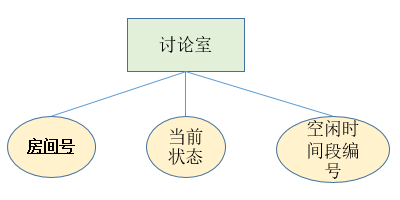
##### 进出控制系统：



##### 单人座位：

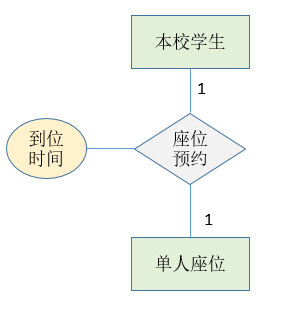


##### 讨论室：

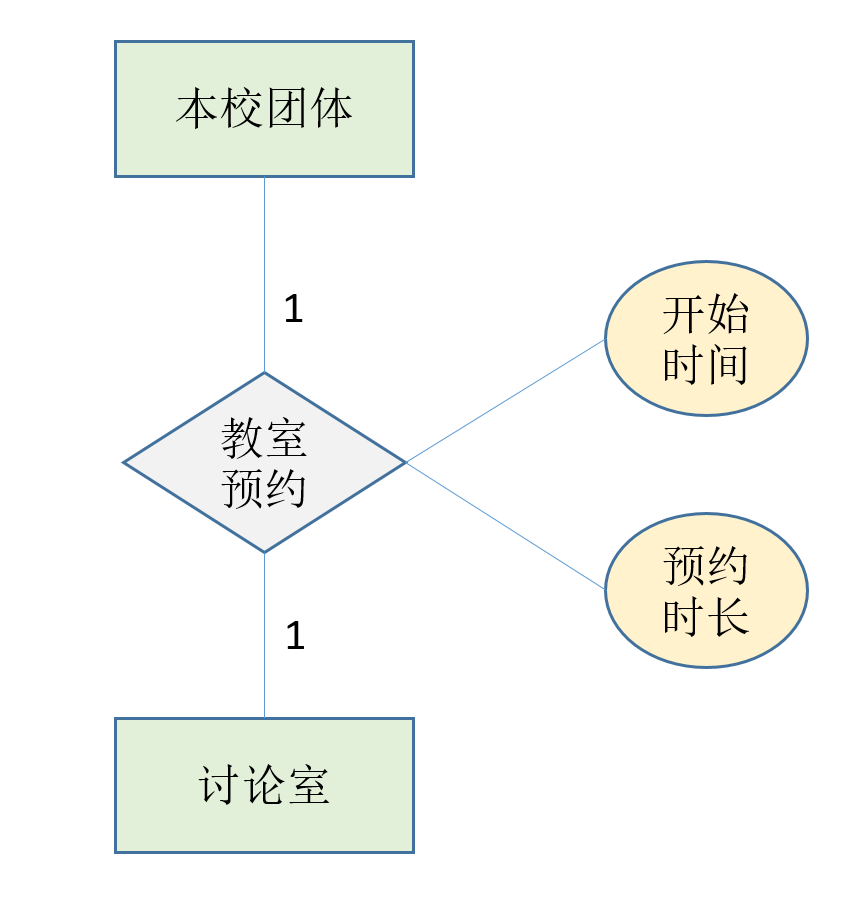


#### 局部E-R图

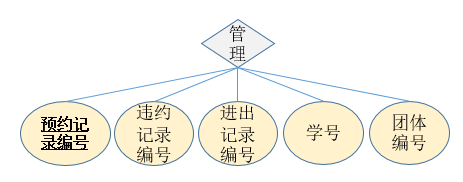
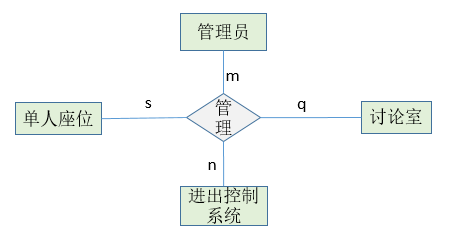
##### 与单人座位有关：



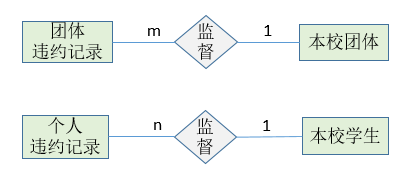
##### 与讨论室有关：



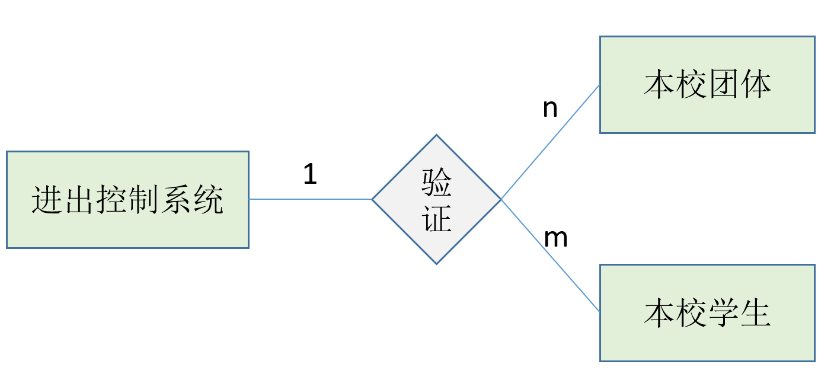
##### 与管理员有关：



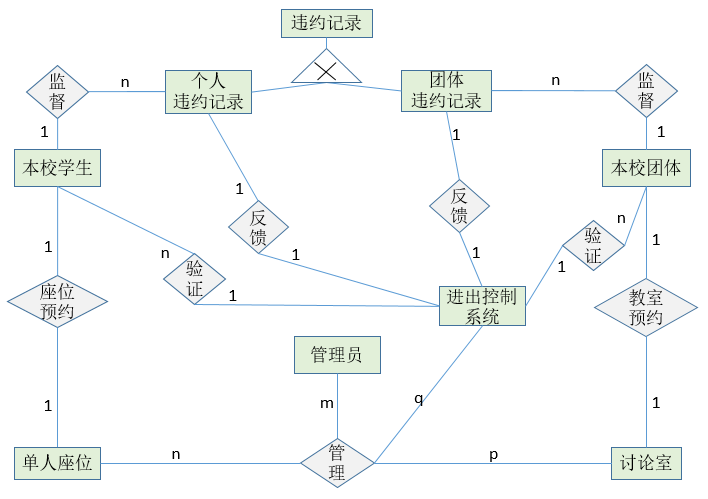
##### 与违约记录有关：



##### 与进出控制系统有关：



### 系统基本E-R图



## 数据库逻辑模式设计

### 数据库关系模式

#### 与实体相关的关系模式

* 本校学生（学号，姓名，性别，登录密码，联系电话，所属系号，学生身份有效期）
* 本校团体（团体编号，团体性质，登录密码，联系电话，成立时间）
* 管理员（管理员工号，姓名，性别，登录密码，联系电话，管理员身份有效期）
* 单人座位（座位编号，当前状态）
* （补充说明：此关系模式作为单人座位的库存记录，当前状态为已被预约、保留、空闲这三种状态之一。）
* 讨论室（房间号，当前状态，空闲时间段编号）
* （补充说明：此关系模式作为讨论室的库存记录。当前状态为已被预约和空闲这两种状态之一。为了方便理解与记录，空闲时间段编号可以是多个编号组合形成的字符串，具体对应如下：

|  |  |
| --- | --- |
| 编号 | 时间段 |
| A | 08:00-09:00 |
| B | 09:00-10:00 |
| C | 10:00-11:00 |
| D | 11:00-12:00 |
| E | 12:00-13:00 |
| F | 13:00-14:00 |
| G | 14:00-15:00 |
| H | 15:00-16:00 |
| I | 16:00-17:00 |
| J | 17:00-18:00 |
| K | 18:00-19:00 |
| L | 19:00-20:00 |
| M | 20:00-21:00 |

比如空闲时间段编号为AC，表示讨论室的空闲时间段为8:00-9:00和10:00-11:00，其余时间段已被预约）

#### 与联系相关的关系模式

* 违约记录（违约记录编号，学号，团体编号，违约时间，违约理由）
* （补充说明：由实际情况可知，任何一个元组中，学号和团体编号必有一个属性值为空，但两个属性值不能同时为空。违约理由可为：迟到——进入图书馆的时间晚于预约时规定的到位时间半个小时以上，比如预约座位时的到位时间为8:00，但是实际到位时间是9:00，则记为违约一次，违约理由为迟到，但是实际到位时间是8:20则不算违约；占座——学生在暂离或离开时，未选择其座位对应状态，系统则认定为恶意占座；超时——离开讨论室的时间晚于预约时规定的离开时间。其中迟到是对于座位和讨论室而言，但占座仅针对单人座位，超时仅针对讨论室而言，即预约座位违约后，违约理由可以是迟到或占座，预约讨论室违约后，违约理由可以是迟到或超时）
* 进出信息记录（进出记录编号，机器编号，学号，团体编号，进入时间，离开时间）
* （补充说明：由实际情况可知，任何一个元组中，学号和团体编号必有一个属性值为空，但两个属性值不能同时为空）
* 座位预约（预约记录编号，学号，座位编号，到位时间）
* （补充说明：为了使座位预约和讨论室预约区分开，在这里预约记录编号规定为“sc\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”，其中\*代表数字0到9，即“sc+28位数字”。到位时间为学生预约时填写的到位时间，若实际到位时间晚于规定时间半个小时以上，则视为违约一次）
* 讨论室预约（预约记录编号，团体编号，房间号，开始时间，预约时长）
* （补充说明：为了使座位预约和讨论室预约区分开，在这里预约记录编号规定为“rc\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”，其中\*代表数字0到9，即“rc+28位数字”。便于统计和记录，开始时间只能为整点，且预约时长只能为小时的整数倍，比如开始时间为8:30则不符合要求，若为8:00则符合要求。预约时长为一个半小时则不符合要求，若为一个小时则符合要求）
* 统计表（预约记录编号，违约记录编号，进出记录编号，学号，团体编号）
* （补充说明：在这里选择预约记录编号为主码，因为违约记录和进出记录是在预约的基础上形成的，即按照实际情况，只有预约了才可能会出现违约的现象，才会产生进出记录，所以违约记录编号和进出记录编号存在可能为空的情况，但是完整性约束要求主码不能为空，因此在这里选择预约记录编号为主码。当违约记录编号为空时表明本次预约后学生准时到位准时离开，当进出记录编号为空时表明本次预约后学生没有到位。且由实际情况可知，任何一个元组中，学号和团体编号必有一个属性值为空，但两个属性值不能同时为空）

### 关系模式范式等级的判定与规范化

* 本校学生（学号，姓名，性别，登录密码，联系电话，系号，学生身份有效期）
* 码：学号，联系电话
* 主码：学号
* 主属性：{学号，联系电话}
* 非主属性：{姓名，性别，登录密码，系号，学生身份有效期}
* 函数依赖：
* 学号 → 姓名
* 学号 → 性别
* 学号 → 登录密码
* 学号 → 联系电话
* 学号 → 系号
* 学号 → 学生身份有效期
* 联系电话 → 学号
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖（说明：候选码学号和联系电话之间有着相互决定的关系，使得传递函数依赖不成立），所以此关系模式属于3NF。

* 本校团体（团体编号，团体性质，登录密码，联系电话，成立时间）
* 码：团体编号，联系电话
* 主码：团体编号
* 主属性：{团体编号，联系电话}
* 非主属性：{团体性质，登录密码，成立时间}
* 函数依赖：
* 团体编号 → 团体性质
* 团体编号 → 登录密码
* 团体编号 → 联系电话
* 团体编号 → 成立时间
* 联系电话 → 团体编号
* 结论

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖（说明：候选码团体编号和联系电话之间有着相互决定的关系，使得传递函数依赖不成立），所以此关系模式属于3NF。

* 管理员（管理员工号，姓名，性别，登录密码，联系电话，管理员身份有效期）
* 码：管理员工号，联系电话
* 主码：管理员工号
* 主属性：{管理员工号，联系电话}
* 非主属性：{姓名，性别，登录密码，管理员身份有效期}
* 函数依赖：
* 管理员工号 → 姓名
* 管理员工号 → 性别
* 管理员工号 → 登录密码
* 管理员工号 → 联系电话
* 管理员工号 → 管理员身份有效期
* 联系电话 → 管理员工号
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖（说明：候选码管理员工号和联系电话之间有着相互决定的关系，使得传递函数依赖不成立），所以此关系模式属于3NF。

* 单人座位（座位编号，当前状态）
* 码：座位编号
* 主码：座位编号
* 主属性：{座位编号}
* 非主属性：{当前状态}
* 函数依赖：
* 座位编号 → 当前状态
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 讨论室（房间号，当前状态，空闲时间段编号）
* 码：房间号
* 主码：房间号
* 主属性：{房间号}
* 非主属性：{当前状态，空闲时间段编号}
* 函数依赖：
* 房间号 → 当前状态
* 房间号 → 空闲时间段编号
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 违约记录（违约记录编号，学号，团体编号，违约时间，违约理由）
* 码：违约记录编号
* 主码：违约记录编号
* 主属性：{违约记录编号}
* 非主属性：{学号，团体编号，违约时间，违约理由}
* 函数依赖：
* 违约记录编号 → 学号
* 违约记录编号 → 团体编号
* 违约记录编号 → 违约时间
* 违约记录编号 → 违约理由
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 进出信息记录（进出记录编号，机器编号，学号，团体编号，进入时间，离开时间）
* 码：进出记录编号
* 主码：进出记录编号
* 主属性：{进出记录编号}
* 非主属性：{机器编号，学号，团体编号，进入时间，离开时间}
* 函数依赖：
* 进出记录编号 → 机器编号
* 进出记录编号 → 学号
* 进出记录编号 → 团体编号
* 进出记录编号 → 进入时间
* 进出记录编号 → 离开时间
* 结论：

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 座位预约（预约记录编号，学号，座位编号，到位时间）
* 码：预约记录编号
* 主码：预约记录编号
* 主属性：{预约记录编号}
* 非主属性：{学号，座位编号，到位时间}
* 函数依赖：
* 预约记录编号 → 学号
* 预约记录编号 → 座位编号
* 预约记录编号 → 到位时间
* 结论

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 讨论室预约（预约记录编号，团体编号，房间号，开始时间，预约时长）
* 码：预约记录编号
* 主码：预约记录编号
* 主属性：{预约记录编号}
* 非主属性：{团体编号，房间号，开始时间，预约时长}
* 函数依赖：
* 预约记录编号 → 团体编号
* 预约记录编号 → 房间号
* 预约记录编号 → 开始时间
* 预约记录编号 → 预约时长
* 结论

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

* 统计表（预约记录编号，违约记录编号，进出记录编号，学号，团体编号）
* 码：预约记录编号
* 主码：预约记录编号
* 主属性：{预约记录编号}
* 非主属性：{违约记录编号，进出记录编号，学号，团体编号}
* 函数依赖：
* 预约记录编号 → 违约记录编号
* 预约记录编号 → 进出记录编号
* 预约记录编号 → 学号
* 预约记录编号 → 团体编号
* 结论

从上面的函数依赖关系及各属性分量可以得出，每个分量都是不可分的数据项，所以此关系模式满足1NF的要求，且关系模式中不存在非主属性对主属性的传递函数依赖，所以此关系模式属于3NF。

### 数据库设计优化

#### 多余表单优化

#### 消除冗余数据

## 最终版修改说明