《数据库原理及应用》课程团队大作业

数据库设计报告

团队（项目组）名称：男上加男队

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 角色 | 班级 | 学号 | 姓名 | 成绩 |
| 组长 | 2022CST | 22213020470137 | 付庭喜 |  |
| 成员 | 2022CST | 20202134047250 | 张成双 |  |

【此处插入一张团队设计过程的合影照片】

2024年6月1日

目 录

[2 引言 7](#_Toc28041)

[2.1 编写目的 7](#_Toc10019)

[2.2 背景 7](#_Toc21667)

[2.3 团队（项目组）要求 7](#_Toc15976)

[2.4 数据库设计过程 7](#_Toc26299)

[2.5 ER模型转换为关系模式的规则 8](#_Toc18314)

[2.6 处理多值属性的方法 9](#_Toc14265)

[2.7 处理复合属性的方法 9](#_Toc3981)

[2.8 优化关系模式的方法 10](#_Toc11869)

[3 业务简述 11](#_Toc16565)

[4 业务架构分析 11](#_Toc22592)

[5 局部概念建模1：付庭喜 12](#_Toc5400)

[5.1 业务模块的详细描述 12](#_Toc18068)

[5.2 分析实体及其属性 13](#_Toc25288)

[5.2.1 实体集 13](#_Toc546)

[5.2.2 实体的属性 13](#_Toc30307)

[5.3 实体集之间的分类关系（父子集关系） 16](#_Toc22563)

[5.4 实体集之间的联系 18](#_Toc5896)

[5.4.1 实体集之间可能存在的联系矩阵勾核表 18](#_Toc3805)

[5.4.2 【患者表】相关的联系 18](#_Toc9981)

[5.4.3 【病历记录表】相关的联系 19](#_Toc18480)

[5.4.4 【预约信息表】相关的联系 20](#_Toc11920)

[5.4.5 【药品表】相关的联系 20](#_Toc18801)

[5.4.6 【处方表】相关的联系 20](#_Toc19195)

[5.4.7 【账单表】相关的联系 21](#_Toc23371)

[5.5 集成的局部概念模型 21](#_Toc2375)

[5.5.1 集成得到的ER简图 21](#_Toc18813)

[5.5.2 去除冗余联系得到的ER简图 21](#_Toc13710)

[5.5.3 联系的属性 22](#_Toc2476)

[5.6 验证模型的正确性 23](#_Toc10510)

[5.6.1 【患者-拥有-病历记录】 23](#_Toc2469)

[5.6.2 【患者-拥有-处方】 24](#_Toc13787)

[5.6.3 【患者-拥有-账单】 25](#_Toc26812)

[5.7 ER模型中不能表达的业务规则 25](#_Toc1342)

[6 局部概念建模2：张成双 26](#_Toc15670)

[6.1 业务模块的详细描述 26](#_Toc9734)

[6.2 分析实体及其属性 26](#_Toc19928)

[6.2.1 实体集 26](#_Toc12881)

[6.2.2 实体的属性 27](#_Toc10509)

[6.3 实体集之间的分类关系（父子集关系） 28](#_Toc28816)

[6.4 实体集之间的联系 29](#_Toc3665)

[6.4.1 实体集之间可能存在的联系矩阵勾核表 29](#_Toc12767)

[6.4.2 【科室】相关的联系 29](#_Toc949)

[6.4.3 【医生】相关的联系 30](#_Toc32740)

[6.4.4 【诊室】相关的联系 31](#_Toc183)

[6.4.5 【医生请假】相关的联系 31](#_Toc32461)

[【医生排班】相关的联系 32](#_Toc8399)

[6.5 集成的局部概念模型 32](#_Toc19500)

[6.5.1 集成得到的ER简图 32](#_Toc9047)

[6.5.2 去除冗余联系得到的ER简图 33](#_Toc10689)

[6.5.3 联系的属性 33](#_Toc27575)

[6.6 验证模型的正确性 34](#_Toc31593)

[6.6.1 【医生-所属-科室】 34](#_Toc3074)

[6.6.2 【医生-排班在-诊室】 35](#_Toc6616)

[6.6.3 【医生-产生-请假信息】 35](#_Toc30052)

[6.6.4 【诊室-归属于-科室】 36](#_Toc18371)

[6.7 ER模型中不能表达的业务规则 36](#_Toc1296)

[7 局部概念建模3：蒋文栋 36](#_Toc17820)

[7.1 业务模块的详细描述 36](#_Toc6544)

[7.2 分析实体及其属性 37](#_Toc8221)

[7.2.1 实体集 37](#_Toc31653)

[7.2.2 实体的属性 37](#_Toc30768)

[7.3 实体集之间的分类关系（父子集关系） 38](#_Toc32537)

[7.4 实体集之间的联系 39](#_Toc14061)

[7.4.1 实体集之间可能存在的联系矩阵勾核表 39](#_Toc9011)

[7.4.2 【排班】相关的联系 40](#_Toc30414)

[7.4.3 【诊所】相关的联系 41](#_Toc30632)

[7.5 集成的局部概念模型 41](#_Toc29515)

[7.5.1 集成得到的ER简图 41](#_Toc29248)

[7.5.2 去除冗余联系得到的ER简图 41](#_Toc6772)

[7.5.3 联系的属性 42](#_Toc656)

[7.6 验证模型的正确性 42](#_Toc30792)

[7.6.1 【医生-排班】 43](#_Toc23658)

[7.6.2 【诊所-排班】 43](#_Toc12761)

[7.7 ER模型中不能表达的业务规则 43](#_Toc30457)

[8 全局概念模型 44](#_Toc13952)

[8.1 对整个业务的详细说明 44](#_Toc28072)

[8.2 实体及其属性 45](#_Toc16294)

[8.2.1 实体集 45](#_Toc6461)

[8.2.2 实体的属性 46](#_Toc15347)

[8.3 实体集之间的分类关系（父子集关系） 50](#_Toc10962)

[8.4 全局概念模型 51](#_Toc26851)

[8.4.1 ER简图 51](#_Toc8082)

[8.4.2 去除冗余联系后的ER简图 52](#_Toc13840)

[8.4.3 联系的属性 53](#_Toc23396)

[8.5 验证模型的正确性 55](#_Toc10159)

[8.5.1 【医生-排班-诊室】 55](#_Toc27114)

[8.5.2 【诊所-排班-医生】 55](#_Toc23059)

[8.5.3 【医生-请假-时间】 56](#_Toc8319)

[8.5.4 【医生-开处方-患者】 56](#_Toc5685)

[8.5.5 【患者-预约-医生】 57](#_Toc28951)

[8.6 ER模型中不能表达的业务规则 57](#_Toc10129)

[9 逻辑结构设计 58](#_Toc32048)

[9.1 实体集转换为关系（表） 58](#_Toc26075)

[9.1.1 表：患者 58](#_Toc12595)

[9.1.2 表：医疗记录 60](#_Toc14673)

[9.1.3 表：预约信息 60](#_Toc13301)

[9.1.4 表：药品 61](#_Toc22204)

[9.1.5 表：处方 61](#_Toc4927)

[9.1.6 表：账单 62](#_Toc6238)

[9.1.7 表：医生 63](#_Toc7549)

[9.1.8 表：诊室 63](#_Toc17241)

[9.1.9 表：医生排班 63](#_Toc6666)

[9.1.10 表：医生请假 64](#_Toc17754)

[9.1.11 表：排班 64](#_Toc23728)

[9.1.12 表：医生工作时间表 64](#_Toc5259)

[9.1.13 表：诊所 65](#_Toc22100)

[9.1.14 表：诊所排班 65](#_Toc27669)

[9.2 联系集转换为表 65](#_Toc5103)

[9.2.1 科室-包含-医生 66](#_Toc12733)

[9.2.2 医生-排班-诊室 66](#_Toc26164)

[9.2.3 医生-记录-病历 66](#_Toc30047)

[9.2.4 医生-请假-时间 67](#_Toc32669)

[9.2.5 患者-拥有-病历 67](#_Toc16727)

[9.2.6 患者-预约-医生 67](#_Toc951)

[9.2.7 医生-开具-处方 67](#_Toc27418)

[9.2.8 处方-包含-药品 68](#_Toc6622)

[9.2.9 患者-产生-账单 68](#_Toc19837)

[9.3 优化关系模式 68](#_Toc26314)

[9.3.1 垂直拆分表 68](#_Toc25235)

[9.3.2 无水平拆分表 68](#_Toc11166)

[9.3.3 合并表 69](#_Toc8442)

[9.4 优化后的关系模式 69](#_Toc32346)

[10 逻辑结构中不能表达的业务规则 69](#_Toc24850)

[11 在MySQL平台中实施的数据库设计 69](#_Toc28374)

[11.1 表 70](#_Toc28791)

[11.2 需要用触发器实现的业务规则 75](#_Toc17593)

[12 在MySQL平台中实现数据库 75](#_Toc28335)

[12.1 数据库关系图 75](#_Toc9738)

[12.2 表的结构 76](#_Toc9201)

[12.2.1 科室表（Department） 76](#_Toc13124)

[12.2.2 医生表（Doctor） 76](#_Toc19304)

[12.2.3 诊室表（Room） 77](#_Toc19735)

[12.2.4 医生排班信息表（Doctor\_Schedule） 77](#_Toc21610)

[12.2.5 医生请假信息表（Doctor\_Leave） 77](#_Toc6467)

[12.2.6 患者表（Patient） 77](#_Toc31345)

[12.2.7 病历记录表（Medical\_Record） 78](#_Toc18219)

[12.2.8 预约信息表（Appointment） 78](#_Toc20958)

[12.2.9 药品表（Medication） 78](#_Toc23056)

[12.2.10 处方表（Prescription） 78](#_Toc6091)

[12.2.11 账单表（Bill） 79](#_Toc11797)

[12.2.12 排班信息表（Schedule） 79](#_Toc17784)

[12.2.13 医生工作时间表（Time\_Table） 79](#_Toc22140)

[12.2.14 诊所表（Clinic） 80](#_Toc11288)

[12.2.15 诊所排班信息表（Clinic\_Schedule） 80](#_Toc9919)

[12.3 勾核数据库实现情况 80](#_Toc17414)

[12.4 表的实现情况 80](#_Toc1390)

[13 数据库应用 81](#_Toc29358)

[13.1 患者管理模块的增删改查（付庭喜同学负责） 81](#_Toc24312)

[13.2 科室+医生管理模块业务的增删改查（张成双同学负责） 82](#_Toc25771)

[13.3 诊室管理模块+排班业务的增删改查（蒋文栋同学负责） 82](#_Toc32717)

[14 设计工作总结 83](#_Toc17898)

# 引言

## 编写目的

记录数据库设计过程、设计约束和设计结果，以便系统设计、实现、测试和维护人员作为参考手册。系统上线后，每一次维护要形成新的设计报告版本，并记录所做的任何更改。

## 背景

1. 拟开发的软件（平台）：**医院管理系统**。该平台的目标：构建一个功能完备的医疗管理系统，涵盖科室管理、医生管理、患者管理、排班管理、病历记录、预约管理、药品管理和账单管理等多个核心模块。
2. 数据库名称：Hospital。
3. 任务提出者：潘文林（数据库课程授课教师）
4. 系统用户：管理员 医生 护士/医护人员 患者 药师 财务人员 技术支持人员

## 团队（项目组）要求

1. 组建2-4人的设计团队（也可称为项目组），3人最佳，最好有一位异性同学。
2. 选出项目组长，项目组长负责组织、召集和督促项目组成员完成相应任务。认真履职的项目组长可获得3-5分的加分。

## 数据库设计过程

1. 项目组全体人员根据下面给出的“业务简述”梳理出业务架构（即业务模块）。
2. 按业务模块分工，项目组成员独立完成自己负责的业务模块的概念建模工作。步骤如下，前4步要求在先在纸面上完成（拍照作为本设计报告的附件），然后再写成电子文档。ER模型用Crow’s Foot表示法表示。
   1. 根据自己的理解，详细描述自己负责的业务模块，补充业务简述不足或不够清晰的内容。
   2. 分析并列出实体集。
   3. 逐一分析各类实体的属性，明确标识属性、复合属性、多值属性、派生属性。
   4. 分析实体集之间的分类关系或父子集关系，说明父子集关系的业务约束，即分类是否是完全划分的、子集是否是互斥的。
   5. 分析并列出实体之间的联系及其属性，明确联系的基数与强制性。
   6. 分析是否需要将将联系实体化（即关系对象化）。
   7. 分析各实体集是否是弱实体集（其参与的联系也是弱联系）。
   8. 画出局部概念模型的EER简图（不含实体集和联系集的属性）。
   9. 用结构化自然语言写出局部概念模型蕴含的业务规则。
   10. 验证模型的正确性，列出相应的假设数据用例。
   11. 分析并列出局部概念模型无法表达出的业务规则，例如属性的约束、属性之间的约束、对象之间的约束、联系之间的约束等。
3. 项目组全体人员一起，将项目组成员提交多个局部概念模型集成为全局概念模型。全局概念建模步骤同局部概念建模，但每一步都需要说明合并局部模型时是否存在各局部模型之间的术语冲突、结构冲突等问题，说明全局模型是如何解决这些问题的。
4. 项目组全体人员一起，严格按照转换规则（见1.5），将全局概念模型ER模型转换为与DBMS无关的逻辑模型（关系模式），处理多值属性（处理方法见1.6），处理复合属性（处理方法见1.7）。优化关系模式，合并或分拆表等，写出优化理由，优化方法见1.8。
5. 根据MySQL数据库管理系统的特性，将上一步得到的与平台无关的关系模式转换为MySQL支持的关系模式。明确属性的数据类型等，派生属性的计算表达式等。
6. 在MySQL数据库中实现数据库。生成数据库关系图、数据库表的结构及已实现的约束，列出MySQL不能支持的业务规则。
7. 按概念建模工作分工，项目成员列出自己所负责业务模块需要的数据增删改查信息处理需求，写出相应的SQL语句。
8. 设计工作总结。每人一段，团队一段。

## ER模型转换为关系模式的规则

ER模型转换为关系模式的基本规则如下：

1. 一个实体集E1转换为一个关系（表）TE1。
   1. 实体E1的属性转换为表TE1的属性；标记出多值属性、复合属性和派生（导出/计算）属性，要优化。
   2. 实体E1的标识属性转换为表TE1的候选码（选出一个作为主码）。如果实体集E1是实体集E0的子集（两者是父子关系），则TE0的主码是TE1的主码（继承父集的主码）也是其外码（引用TE0.PK）。
   3. 处理多值属性。方法见1.6节。
   4. 处理复合属性。方法见1.7节。
2. 一个联系集（E1-R1-E2）转换为一个关系（表）T\_E1\_E2\_R1。
   1. 表T\_E1\_E2\_R1的属性由三部分构成，TE1.PK + TE2.PK + 联系R1的属性。
   2. 表T\_E1\_E2\_R1的外码有两个：FK1=（TE1.PK）、FK2=（TE2.PK）。
   3. 表T\_E1\_E2\_R1的候选码：
      1. 若联系的基数是1:1，则有两个候选码，分别是CK1=（TE1.PK）、CK2=（TE2.PK），选联系的主导者为主码。
      2. 若联系的基数是1（E1）:n（E2），则有1个候选码，即CK=（TE2.PK）=PK。
      3. 若联系的基数是m（E1）:n（E2），则有1个候选码，即CK=（TE1.PK ，TE2.PK）=PK。
   4. 处理多值属性。方法见1.6节。
   5. 处理复合属性。方法见1.7节。

## 处理多值属性的方法

检查表的属性中是否存在多值属性，如果有按照如下规则处理：将多值属性从表T中拆分出来为一个表S。参考如下：

| T（原始表） | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| A | PK |  | ● |  |  |  |  |
| B |  |  | ● |  |  |  |  |
| C |  |  |  | ● |  |  |  |

多值属性C单独拆分成一个表，拆分后原始表成为如下两个表：

| T | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | PK | FK | 非空 | 派生 | 说明 |
| A | PK |  | ● |  |  |
| B |  |  | ● |  |  |

| TC | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | PK | FK | 非空 | 派生 | 说明 |
| A | PK | Fk1 | ● |  | 引用T.PK |
| C | PK |  | ● |  |  |

## 处理复合属性的方法

检查每个表的属性中是否存在复合属性，如果有按照如下规则处理：将复合属性分拆为多个简单单值属性，复合属性成为一个派生属性。参考如下：

| TC（原始） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | PK | FK | 非空 | 复合 | 派生 | 说明 |
| A | PK | Fk1 | ● |  |  | 引用T.PK |
| **C** | **PK** |  | ● | ● |  | C=D+E+F |

TC表的复合属性C分拆成D、E、F三个属性，C成为派生属性，处理后的表如下：

| TC（处理复合属性后） | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | PK | FK | 非空 | 派生 | 说明 |
| A | PK | Fk1 | ● |  | 引用T.PK |
| **D** | **PK** |  | ● |  |  |
| **E** | **PK** |  | ● |  |  |
| **F** | **PK** |  | ● |  |  |
| C |  |  |  | ● | C=D+E+F |

## 优化关系模式的方法

优化关系模式的方法如下：

1. 垂直拆分表。可以把一张表拆分为一个主表和多个子表，所有表的主码都是拆分前表的主码，子表的主码同时也是是外码（引用主表的主码）。拆分的依据：（1）主表的列过多，或数据量很大；（2）主表经常被查询，子表偶尔被查询；（3）主表是主信息，子表对应相应的业务模块独有的属性；（4）其它设计上的考虑。
2. 水平拆分表。把一张表拆分成多张表，车分后的表的关系模式和车分前的表的相同。要求：拆分后的表的并集是拆分前的表，拆分的表无交集。拆分依据：（1）表的数量很大；（2）不用子集的用途有区别，例如一个表的数据中有正在处理的业务数据和已处理完毕不会再修改的历史数据，则可以拆分成业务数据表和历史数据表。
3. 合并表。
   1. 可以把1:1联系集转换得到的表S合并到与其主码相同的、其外码引用的表T中。除S表的主码（也是S表的外码）外，其它列全部合并到T表中，这些列的数据合并到T表后是否是强制（非空）的，需要看S表的数据是否是T表的真子集，如果是，则列的值可空，否则保留其在S表中的特性。合并后S的候选码（也是S表的另外一个外码）也成为T表的候选码（其值可能允许空），同时该候选码也成为T表的一个外码。
   2. 可以把1:n联系集转换得到的表S合并到与其主码相同的、其外码引用的表T中。除S表的主码（也是S表的外码）外，其它列全部合并到T表中，这些列的数据合并到T表后是否是强制（非空）的，需要看S表的数据是否是T表的真子集，如果是，则列的值可空，否则保留其在S表中的特性。合并后S的另外一个外码成为T表的一个外码。
   3. 可以把子集转换得到的TE1表的所有列（主码除外）合并到期父集转换得到的表TE0中，增加一个子集标识列“是否是E1子集”。如果子集表TE1除了主码外没有其它属性，则应合并到其父集表TE0中。合并后，引用子集表TE1的表的外码引用父集表TE0的主码。
4. 抽取枚举类型编码或可数名称编码表。
   1. 抽取类型代码或编码表。表TE的属性A（例如“证照类型”），其属性值是枚举值、枚举值有三种以上、枚举值是一个字符串：增加一个关系（表）TTA来存放枚举值，该表由至少2列（A编码,A名称）构成，例如“证照类型（证照编码，证照名称）”，视需要可增加“备注”列或 “别名”列之类的；TE的属性A更名为“A编码”（外码），引用TTA表的主码（A编码），此类引用关系可以不用实现。对于分类编码是国际标准、国家标准或行业标准，则类型编码表可以设计为“证照类型代码（证照编码，证照名称，标准代码，标准名称，备注）”。
   2. 抽取名称编码表。表TE的属性A（例如“毕业中学名称”），属性值是一个取值种类可数、数据长期不变的的字符串，例如属性值是“怒江州民族中学”、“云南师范大学附中”等，这些属性值是长期不变的且可数的。可以增加一个表TTA来存储此类属性值，例如“毕业中学代码（学校编码，学校名称，学校简称，备注）”。表TE的属性A更改为“A编码”，引用TTA.编码，引用关系可不实现。如果名称编码是国际标准、国家标准或行业标准，处理方法同上。

# 业务简述

平台支持的业务（即目标系统应具备的业务支持能力）描述如下：

本次业务架构涵盖了多个关键模块。科室管理模块专注于维护科室的基本信息，包括科室 ID 和名称。医生管理模块全面存储医生的个人信息，如姓名、职称、电话等，并明确其所属科室，同时记录排班和请假信息。诊室管理模块有效管理诊室的名称及其所属科室。患者管理模块妥善保存患者的基本资料，涵盖姓名、性别、年龄、地址和电话，记录病历相关信息及预约状态。药品管理模块详细登记药品的名称、用法和剂量等。排班管理模块精心安排医生的具体排班并记录工作时间。诊所管理模块准确记录诊所的基本信息，管理其排班详情。财务相关模块负责生成患者账单，涵盖诊疗费、药品费用和总费用等，并记录处方信息，关联患者、医生和药品。各个模块相互协作，共同构建起一个完善且高效的医疗业务体系。

# 业务架构分析

【说明：项目组全体人员根据“业务简述”梳理出业务架构（即业务模块）。】

通过分析，平台主要业务模块及其局部概念模型建模及外模式设计负责人如下表。

表 1 平台的主要业务模块及建模任务分配表

| 编号 | 业务模块名称 | 相关业务工作说明 | 负责人 |
| --- | --- | --- | --- |
| B2 | 患者管理模块 |  | 付庭喜 |
| B3 | 医生管理模块 |  | 张成双 |
| B4 | 排班管理模块 |  | 蒋文栋 |

# 局部概念建模1：付庭喜

【说明：付庭喜是本部分概念建模的负责人】。

## 业务模块的详细描述

【说明：根据自己的理解，详细描述自己负责的业务模块，补充业务简述不足或不够清晰的内容。】

业务模块的详细描述如下表。

表 2 局部业务模块的详细描述

| 编号 | 业务模块名称 | 详细说明 |
| --- | --- | --- |
| B3 | 患者管理模块 | Patient（患者表）:  存储患者的基本信息，如姓名、性别、年龄、地址和电话号码。  Medical\_Record（病历记录表）:  记录每位患者的医疗记录，包括日期、诊断、处方等信息，并通过患者ID和医生ID与患者表和医生表关联。  Appointment（预约信息表）:  包含患者与医生的预约信息，包括预约日期和状态（例如预约成功或取消），通过患者ID和医生ID与患者表和医生表关联。  Medication（药品表）:  存储可用药品的信息，包括药品名称、用法说明和剂量信息。  Prescription（处方表）:  记录医生为患者开具的处方信息，包括处方日期、相关患者和医生的信息以及药品ID，通过外键与患者表、医生表和药品表关联。  Bill（账单表）:  包含患者的费用信息，包括诊疗费、药品费用和总费用，同时记录了账单日期，并通过患者ID、医生ID和处方ID与患者表、医生表和处方表关联。 |

## 分析实体及其属性

### 实体集

业务模块中包含的实体如下表【提示：一个实体只能用一种量词，如果在不同语境中需要使用不同的量词，则说明不是一个概念】。

表 3 业务模块中包含的实体集

| 编号 | 实体 | 量词 | 实体集含义说明（概念定义） |
| --- | --- | --- | --- |
| E1\_1 | 患者 | 位 | 该表包括所有到访医院或诊所的个体，需记录姓名、性别、年龄、地址和电话等基本信息。 |
| E1\_2 | 医疗记录 | 条 | 包括每位患者在就诊时记录的病历信息，如诊断、处方及相关医生信息。 |
| E1\_3 | 预约信息 | 条 | 包含患者与医生之间的预约记录，记录预约日期及状态，用于安排诊疗服务。 |
| E1\_4 | 药品 | 种 | 用于治疗或管理疾病的药品信息，包括药品名称、用法说明和剂量信息。 |
| E1\_5 | 处方 | 张 | 记录医生为患者开具的处方信息，包括处方日期、相关患者和医生的信息以及药品信息。 |
| E1\_6 | 账单 | 项 | 包含患者的费用信息，如诊疗费、药品费用及总费用，用于结算医疗服务的成本。 |

### 实体的属性

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |

| E1\_2医疗记录 (Medical Records) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| record\_id (医疗记录的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于跟踪每条医疗记录。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该医疗记录的患者的唯一标识符。 |
| date (就诊日期) |  | ● |  |  |  | 输入就诊日期，记录患者每次的就诊时间。 |
| diagnosis (诊断信息) |  |  |  |  |  | 输入诊断信息，描述患者的疾病或健康问题。 |
| prescription (处方信息) |  |  |  |  |  | 输入医生开具的处方信息。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该医疗记录的医生的唯一标识符。 |

| E1\_3预约信息 (Appointments) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| appointment\_id (预约信息的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个预约的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该预约的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该预约的医生的唯一标识符。 |
| appointment\_date (预约日期) |  | ● |  |  |  | 输入预约的日期和时间。 |
| status (预约状态，如预约成功或取消) |  |  |  |  |  | 输入预约的状态，例如预约成功或取消。 |

| E1\_4药品 (Medications) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| medication\_id (药品的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每种药品的唯一标识。 |
| medication\_name (药品的名称) |  | ● |  |  |  | 输入药品的名称。 |
| usage\_instructions (药品的用法说明) |  | ● |  |  |  | 输入药品的使用说明，如用药频次和方法。 |
| dosage (药品的剂量信息) |  | ● |  |  |  | 输入药品的剂量信息，如每次服用的量。 |

| E1\_5处方 (Prescriptions) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| prescription\_id (处方的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个处方的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该处方的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | *关联开具该处方的医生的唯一标识符。* |
| medication\_id (关联的药品ID) |  | ● |  |  |  | 关联的药品ID |
| prescription\_date (处方开具日期) |  | ● |  |  |  | 输入处方开具的日期。 |

| E1\_6账单 (Bills) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| bill\_id (账单的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每张账单的管理和跟踪。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该账单的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联的医生ID |
| prescription\_id (关联的处方ID) |  | ● |  |  |  | 关联的处方ID |
| consultation\_fee (诊疗费用) |  |  |  |  |  | 输入与诊疗相关的费用。 |
| medication\_fee (药品费用) |  |  |  |  |  | 输入与药品相关的费用。 |
| total\_amount (总费用) |  |  |  |  |  | 输入账单的总费用。 |
| bill\_date (账单日期) |  | ● |  |  |  | 输入账单的日期。 |

## 实体集之间的分类关系（父子集关系）

在患者管理模块中，存在以下实体集之间的分类关系（父子集关系）：患者表（Patient）是其他几个表的父集：

1. 病历记录表（Medical\_Record）通过 patient\_id 外键关联到患者表，表明每个病历 记录属于特定的患者。
2. 预约信息表（Appointment）通过 patient\_id 外键关联到患者表，说明每个预约是由 特定患者发起的。
3. 医生表（Doctor）与其他表的关系：

病历记录表（Medical\_Record）通过 doctor\_id 外键关联到医生表，确定负责每个病历 的医生。

预约信息表（Appointment）通过 doctor\_id 外键关联到医生表，明确每个预约对应的 医生。

1. 药品表（Medication）与处方表（Prescription）的关系：

处方表（Prescription）通过 medication\_id 外键关联到药品表，指出每个处方中包含的 具体药品。

1. 处方表（Prescription）与其他表的关系：

通过 patient\_id 外键关联到患者表，表明处方所属的患者。

通过 doctor\_id 外键关联到医生表，确认开具处方的医生。

1. 账单表（Bill）与其他表的关系：

通过 patient\_id 外键关联到患者表，确定账单对应的患者。

通过 doctor\_id 外键关联到医生表，表明为患者服务的医生。

通过 prescription\_id 外键关联到处方表，明确账单所基于的处方。

患者管理模块分类关系如下EER图。



图 1 患者管理模块关系图

患者管理模块关系的业务规则如下：

* 完全划分：患者诊疗信息 = 病历记录 ∪ 预约信息 ∪ 处方 ∪ 账单
* 子集互斥：病历记录 ∩ 预约信息 ∩ 处方 ∩ 账单 = Φ
* 完全划分：处方 = 患者相关处方 ∪ 医生相关处方 ∪ 药品相关处方
* 子集互斥：患者相关处方 ∩ 医生相关处方 ∩ 药品相关处方 = Φ
* 完全划分：患者 = 有病历患者 ∪ 有预约患者 ∪ 有处方患者 ∪ 有账单患者
* 子集互斥：有病历患者 ∩ 有预约患者 ∩ 有处方患者 ∩ 有账单患者 = Φ

## 实体集之间的联系

### 实体集之间可能存在的联系矩阵勾核表

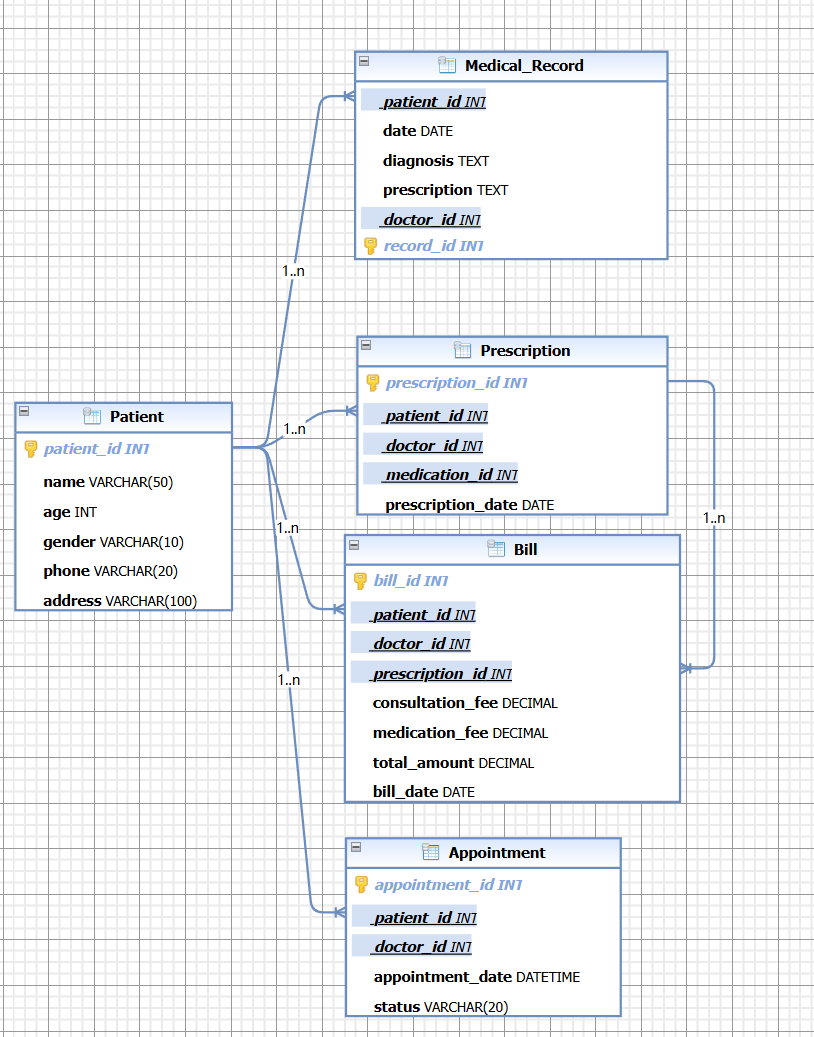
实体集之间可能存在的联系矩阵如下表（●表示实体集之间至少存在一种联系）。

|  | 患者表 | 病历记录表 | 预约信息表 | 药品表 | 处方表 | 账单表 |
| --- | --- | --- | --- | --- | --- | --- |
| 患者表 |  | ● | ● |  | ● | ● |
| 病历记录表 | ● |  | ● |  |  |  |
| 预约信息表 | ● |  |  |  |  |  |
| 药品表 |  |  | ● |  | ● |  |
| 处方表 | ● |  |  | ● |  |  |
| 账单表 | ● |  |  |  | ● |  |

### 【患者表】相关的联系

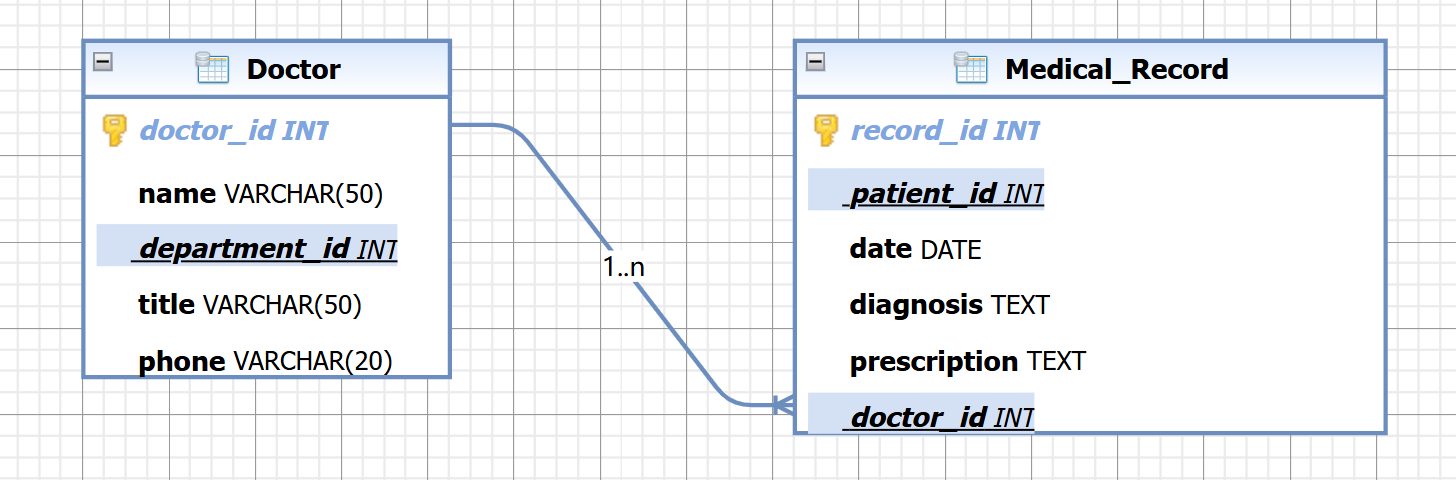
【患者表】相关的联系如下图。

【有的实体集可能依赖另一实体集才能区别开其实例，这样的实体集是弱实体集，要用弱实体集及弱联系集符号来表达出来】



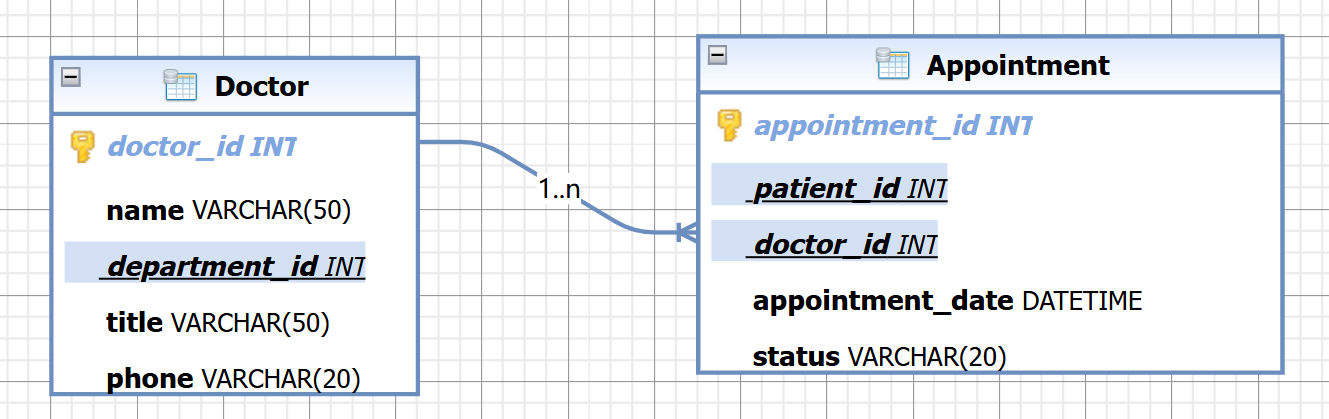
### 【病历记录表】相关的联系

【病历记录表】相关的联系如下图。



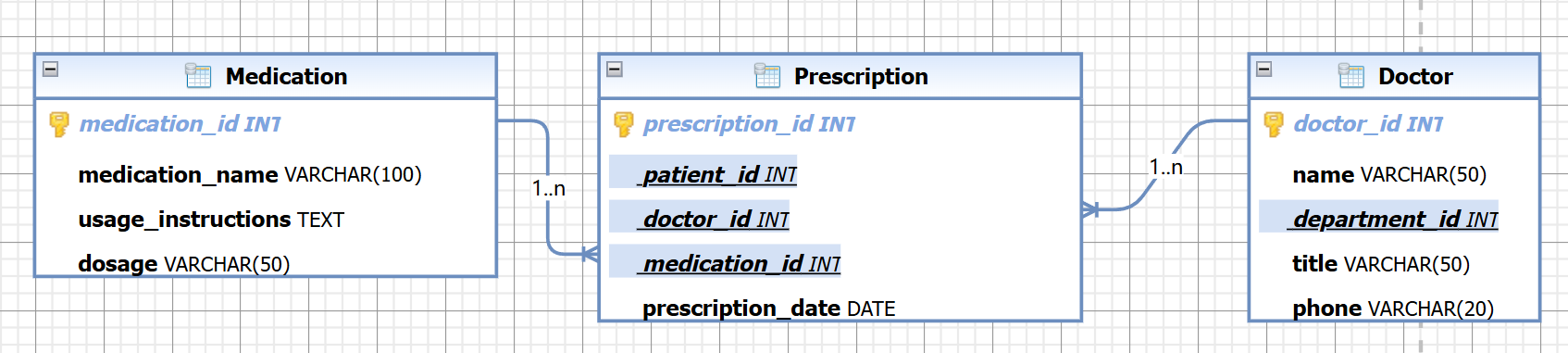
### 【预约信息表】相关的联系

【预约信息表】相关的联系如下图。



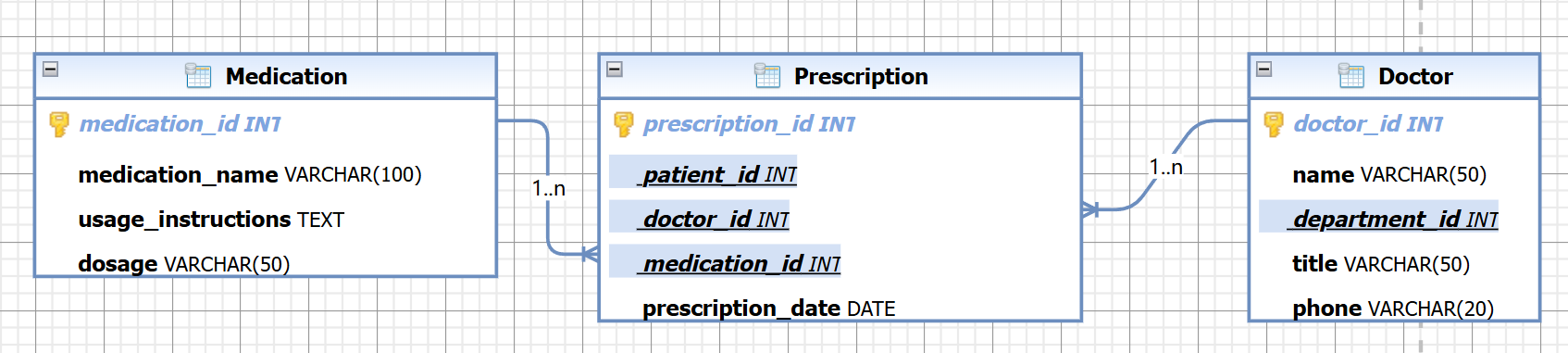
### 【药品表】相关的联系

【药品表】相关的联系如下图。



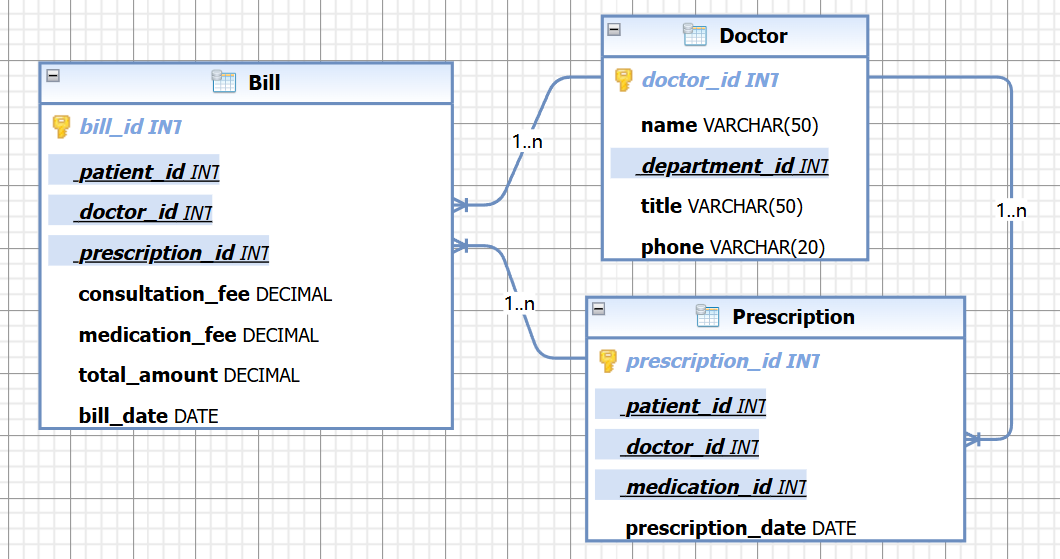
### 【处方表】相关的联系

【处方表】相关的联系如下图。



### 【账单表】相关的联系

【账单表】相关的联系如下图。



## 集成的局部概念模型

### 集成得到的ER简图

将上面分析出来各个局部的ER模型合并（集成）在一起，形成一个完整的局部概念模型，ER简图如下。



### 去除冗余联系得到的ER简图

分析集成后的局部概念模型中存在冗余的联系，冗余联系是可以由其它联系推导出来的或者说是由其它联系蕴含的联系。以下联系是冗余的：

* 在实际业务中，通过预约信息表中的“appointment\_date”（预约日期）和患者表中的某些信息，能够完全确定患者的预约状态，那么“Appointment”表中的“status”（状态）字段存在冗余。
* 通过病历记录表中的详细诊断描述和处方内容，能够精确推断出患者所使用的药品，那么处方表中的部分药品相关信息可能存在冗余。
* 通过患者表中的年龄和性别等信息，能够大致推测出患者可能患有的常见疾病类型，那么病历记录表中的某些常见疾病诊断字段有冗余。
* 在系统中，通过账单表中的处方 ID 能够获取到关于处方的所有详细费用信息，那么账单表中直接记录的部分药品费用相关字段存在冗余。

去除冗余联系后的ER简图如下。



### 联系的属性

【患者-拥有-病历记录】联系的属性如下：

| 患者-拥有-病历记录 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| date（病历日期） | ● |  |  |  | 精确记录病历创建的具体日期，用于跟踪和追溯患者的诊疗时间线，为后续的病情分析和治疗决策提供时间参考 |

【患者-预约-医生】联系的属性如下：

| 患者-预约-医生 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| appointment\_date（预约日期） | ● |  |  |  | 明确患者预约医生的具体日期和时间，便于医院进行资源调配和患者有序就诊安排。 |
| status（预约状态） | ● |  |  |  | 标识预约的当前状态，如“已预约”“已取消”“已就诊”等，有助于医院和患者及时了解预约的进展和情况，以便进行相应的处理和调整。 |

【患者 - 持有 - 处方】联系的属性如下：

| 患者 - 持有 - 处方 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| prescription\_date（处方开具日期） | ● |  |  |  | 准确记录处方的开具时间，对于药品的使用期限、治疗周期的规划以及后续处方的调整和跟踪具有重要意义。 |

提示：逐一分析各联系是否有属性，如果有，参照上述格式列出来。

## 验证模型的正确性

验证局部模型中每个联系集的正确性：用结构化自然语言描述每个联系表达的业务规则，用假设数据验证是否存在逻辑矛盾。

### 【患者-拥有-病历记录】

【患者 - 拥有 - 病历记录】表达的业务规则如下：

* （1 个）患者 - 拥有 - （n 个）病历记录；
* （1 个）病历记录 - 属于 - （1 个）患者。

假设的数据如下（含正反例）：

| 参与联系的实体 | | 联系的属性 | 事实判定 |
| --- | --- | --- | --- |
| 患者 |  | 病历记录 |
| 张三 |  | 2024 年 5 月 1 日病历 | ✔ |
| 张三 |  | 2024 年 5 月 1 日病历 | ✔ |
| 张三 |  | 2024 年 5 月 1 日病历 | ✔ |
| **李四** |  | 2024 年 5 月 1 日病历 | ✔ |
| **李四** |  | 2023 年 12 月 30 日病历 | ✔ |
| **赵六** |  | 2024 年 5 月 1 日病历（同一患者重复） | ✔ |
| 马奇 |  | 2024 年 6 月 1 日病历（不存在的患者） | ✘ |
| 吴迪 |  | 2024 年 6 月 1 日病历（患者与病历不对应） | ✘ |

### 【患者-拥有-处方】

【患者-拥有-处方】表达的业务规则如下：

* （1 个）患者 - 拥有 - （n 个）处方；
* （1 个）处方 - 属于 - （n 个）患者；

假设的数据如下（含正反例）：

| **参与联系的实体** | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **患者** | 处方 |  |
| 张三 | 2024年7月5 日处方 | ✔ |
| **李四** | 2024年8月 10 日处方 | ✔ |
| 王五 | 2024年9 月 15 日处方（同一患者重复） | ✔ |
| 赵六 | 2024年10 月 20 日处方（不存在的患者） | ✘ |
| **张三** | 2024年7 月 6 日处方（患者与处方不对应） | ✘ |

### 【患者-拥有-账单】

【患者-拥有-账单】表达的业务规则如下：

* （1 个）患者 - 拥有 - （n 个）账单；
* （1 个）账单 - 属于 - （1 个）患者；

假设的数据如下（含正反例）：

| **参与联系的实体** | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **患者** | 账单 |  |
| 张三 | 2024年7月5 日账单 | ✔ |
| **李四** | 2024年8月 10 日账单 | ✔ |
| 王五 | 2024年9 月 15 日账单（同一患者重复） | ✔ |
| 赵六 | 2024年10 月 20 日账单（不存在的患者） | ✘ |
| **张三** | 2024年7 月 6 日账单（患者与账单不对应） | ✘ |

## ER模型中不能表达的业务规则

提示：分析上述模型中不能表达出来的业务规则，例如：【学生-重修课程】是【学生-选修-课程】的子集。

ER模型中不能表达或描述的业务规则如下表。

| 序号 | 约束类别 | 业务规则说明 |
| --- | --- | --- |
| 1 | 实体集内部实体间约束 | 同一患者在一定时间内的预约次数不能超过 5 次 |
| 2 | 实体集的统计约束 | 一个医生在一天内接诊的患者数量不能超过 30 人 |
| 3 | 实体集之间的约束 | 同一患者在一个月内的处方数量不能超过 10 个 |
| 4 | 联系集之间的约束 | 【患者 - 重新预约】是【患者 - 预约】的子集，只有原预约取消或变更的才能重新预约 |

# 局部概念建模2：张成双

【说明：张成双是本部分概念建模的负责人】。

## 业务模块的详细描述

【说明：根据自己的理解，详细描述自己负责的业务模块，补充业务简述不足或不够清晰的内容。】

业务模块的详细描述如下表。

表 2 局部业务模块的详细描述

| 编号 | 业务模块名称 | 详细说明 |
| --- | --- | --- |
| B3 | 科室管理模块 | 科室是医疗系统的基本组织单元。 科室表通过科室ID主键唯一标识科室，科室名称记录科室名称，明确各科室的区分。 |
| B4 | 医生管理模块 | 医生是医疗服务的核心。  1、医生表以医生ID主键标识医生， `department\_id`关联所属科室，`title`表明职称，`phone`为联系方式。  2、诊室表用诊室ID主键标识诊室，诊室名称记录诊室名称，科室ID关联所属科室。  3、医生排班信息表的`schedule\_id`主键安排医生工作时间，`doctor\_id`关联医生，`date`和`time\_slot`确定工作时段，`clinic\_room`关联诊室。  4、医生请假信息表的请假ID主键记录请假，`doctor\_id`关联医生，`start\_date`和`end\_date`确定请假起止，`reason`记录理由。 |

## 分析实体及其属性

### 实体集

业务模块中包含的实体如下表【提示：一个实体只能用一种量词，如果在不同语境中需要使用不同的量词，则说明不是一个概念】。

表 3 业务模块中包含的实体集

| 编号 | 实体 | 量词 | 实体集含义说明（概念定义） |
| --- | --- | --- | --- |
| E1\_1 | 科室 | 个 | 医疗机构中按照特定医疗领域划分的组织单位集合 |
| E1\_2 | 医生 | 个 | 在医疗机构中为患者提供医疗服务的专业人员的集合 |
| E1\_3 | 诊室 | 间 | 医疗机构中用于医生为患者进行诊断和治疗的特定房间的集合 |
| E1\_4 | 医生排班 | 次 | 对医生工作时间和任务安排的记录的集合 |
| E1\_5 | 医生请假 | 次 | 医生因个人原因申请暂停工作的记录的集合 |

### 实体的属性

| E1\_1：科室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 科室ID | ID1 | ● |  |  |  | 科室的唯一标识符 |
| 科室名称 |  | ● |  |  |  | 科室的具体名称 |

| E1\_2：医生 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 医生ID | ID1 | ● |  |  |  | 医生的唯一标识符 |
| 姓名 |  | ● |  |  |  | 医生的称呼 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |
| 职称 |  | ● |  |  |  | 医生的专业职称 |
| 电话 |  |  |  |  |  | 医生的联系电话 |

| E1\_3：诊室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 诊室ID | ID1 | ● |  |  |  | 诊室的唯一标识符 |
| 诊室名称 |  | ● |  |  |  | 诊室的具体名称 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |

| E1\_4：医生排班 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班ID | ID1 | ● |  |  |  | 医生排班的唯一标识符 |
| 医生ID |  | ● |  |  |  | 参与排班的医生的标识符 |
| 日期 |  | ● |  |  |  | 排班的具体日期 |
| 时间段 |  | ● |  |  |  | 排班的时间区间 |
| 诊室ID |  | ● |  |  |  | 排班对应的诊室标识符 |

| E1\_5：医生请假 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假ID | ID1 | ● |  |  |  | 医生请假的唯一标识符 |
| 医生ID |  | ● |  |  |  | 请假医生的标识符 |
| 开始日期 |  | ● |  |  |  | 请假的起始日期 |
| 结束日期 |  | ● |  |  |  | 请假的结束日期 |
| 请假理由 |  |  |  |  |  | 医生请假的原因说明 |

## 实体集之间的分类关系（父子集关系）

医疗机构的人员和事务包括科室 、 医生 、 诊室 、 医生排班 、 医生请假等

医疗机构的人员和事务分类关系如下EER图。

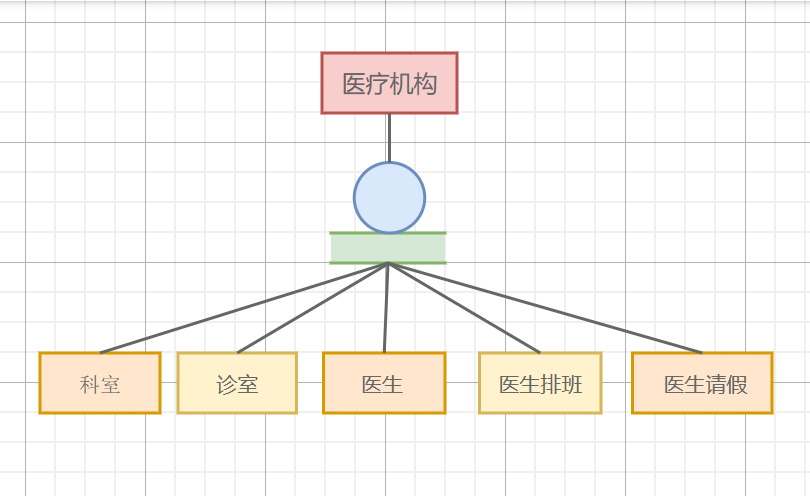


图 1 学生分类关系图

医疗机构的人员和事务分类关系的业务规则如下：

* 完全划分：医疗机构的人员和事务 = 科室 ∪ 医生 ∪ 诊室 ∪ 医生排班 ∪ 医生请假
* 子集互斥：科室 ∩ 医生 ∩ 诊室 ∩ 医生排班 ∩ 医生请假 = Φ

## 实体集之间的联系

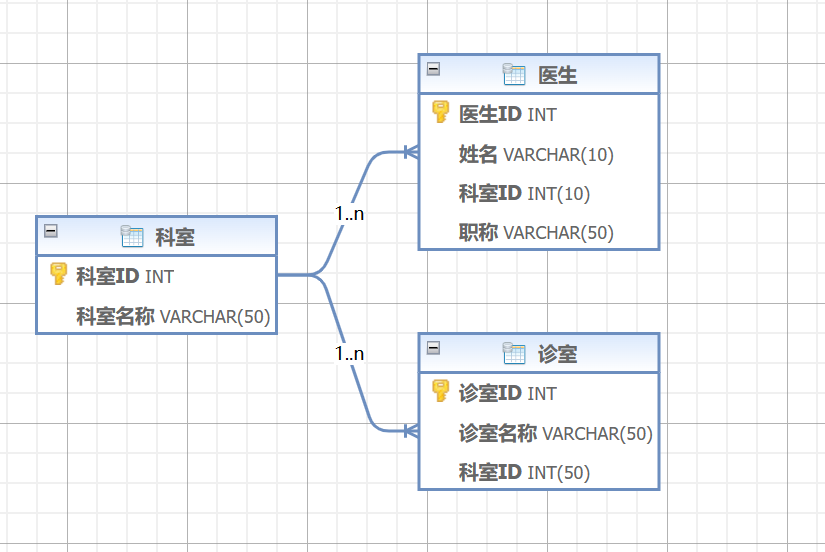
### 实体集之间可能存在的联系矩阵勾核表

实体集之间可能存在的联系矩阵如下表（●表示实体集之间至少存在一种联系）。

|  | 科室[个] | 医生[个] | 诊室[间] | 医生排班[次] | 医生请假[次] |
| --- | --- | --- | --- | --- | --- |
| 科室[个] | ● | ● | ● | ● | ● |
| 医生[个] | ● | ● | ● | ● | ● |
| 诊室[间] | ● | ● | ● | ● | ● |
| 医生排班[次] | ● | ● | ● | ● | ● |
| 医生请假[次] | ● | ● | ● | ● | ● |

### 【科室】相关的联系

【科室】相关的联系如下图。

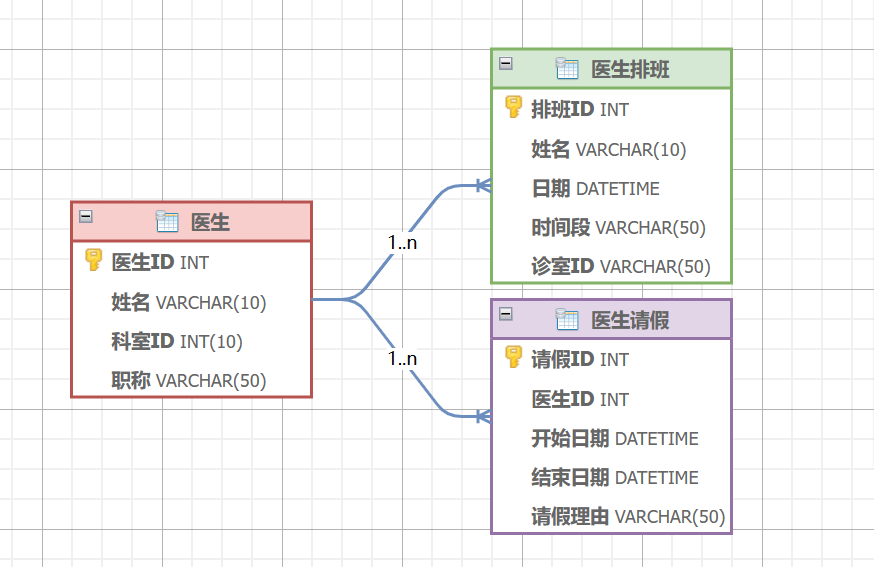


提示：对每个实体集，分析其自身内部对象之间的关系，分析其余其它实体集之间的联系，两个实体集之间的联系可能存在多种，例如学生与课程之间的联系至少存在“初修”、“重修”两种联系。

【有的实体集可能依赖另一实体集才能区别开其实例，这样的实体集是弱实体集，要用弱实体集及弱联系集符号来表达出来】

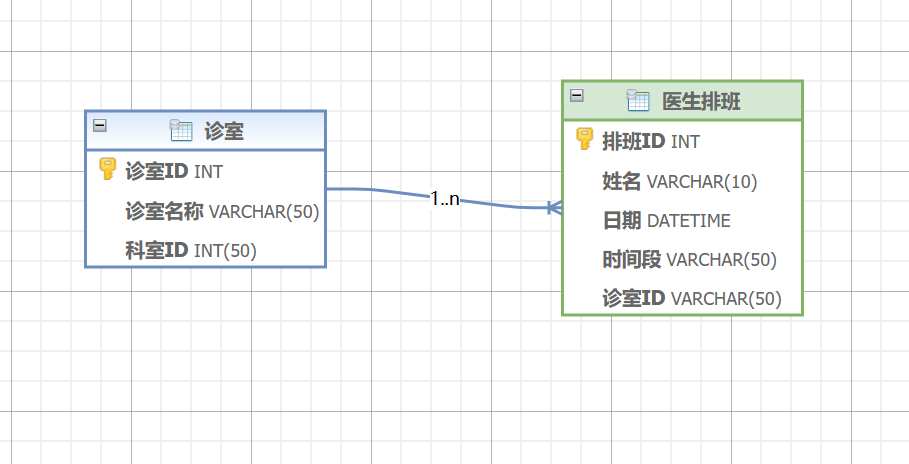
### 【医生】相关的联系

【医生】相关的联系如下图。



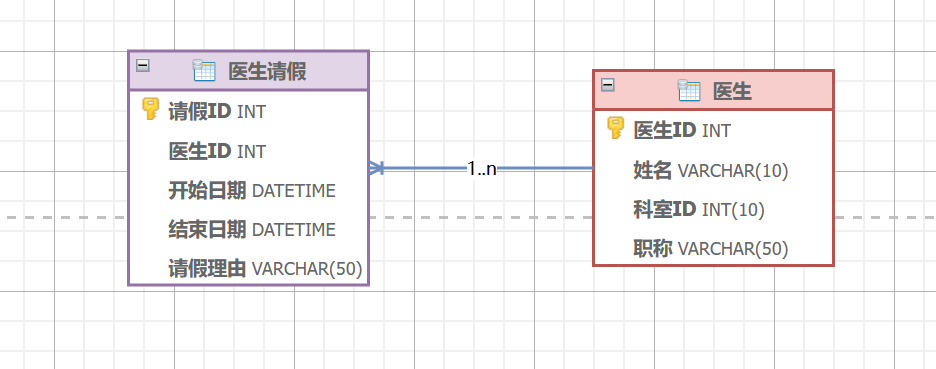
### 【诊室】相关的联系

【诊室】相关的联系如下图。



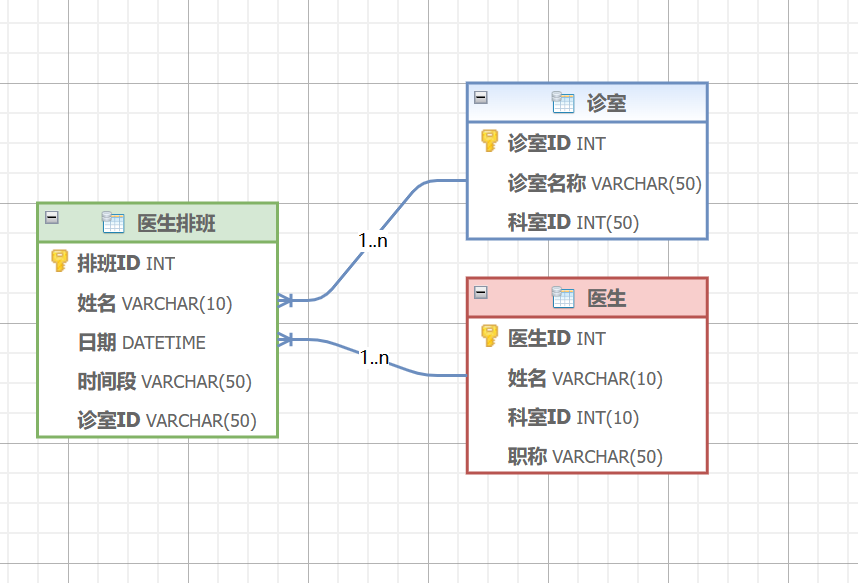
### 【医生请假】相关的联系

【医生请假】相关的联系如下图。



### 【医生排班】相关的联系

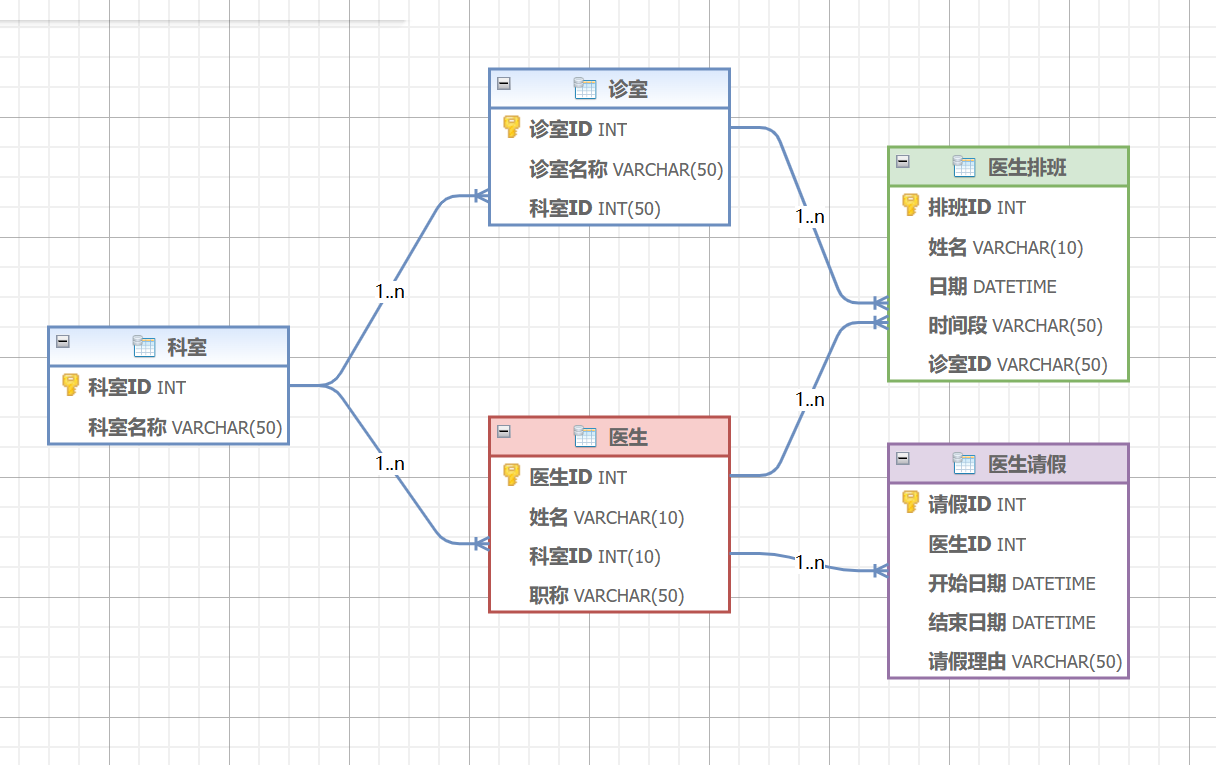
【医生排班】相关的联系如下图。



## 集成的局部概念模型

### 集成得到的ER简图

将上面分析出来各个局部的ER模型合并（集成）在一起，形成一个完整的局部概念模型，ER简图如下。



### 去除冗余联系得到的ER简图

已是无冗余 ER 简图。

### 联系的属性

【**医生-所属-科室**】联系的属性如下：

| 医生-所属-科室 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 入职年份 | ● |  |  |  | 医生加入所属科室的年份 |
| 在科时长 | ● |  |  |  | 医生在该科室工作的时长 |
| 在科时长 | ● |  |  |  | 医生在所属科室负责诊治的病种 |
| 科室考核结果 |  |  |  |  | 医生在所属科室的考核成绩或结果 |

【医生-排班在-诊室】联系的属性如下：

| 医生-排班在-诊室 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班状态 | ● |  |  |  | 表示医生在诊室的排班状态，如已排、未排 |
| 特殊需求 | ● |  |  |  | 医生在该诊室排班时的特殊需求 |

【医生-产生-请假信息】联系的属性如下：

| 医生-产生-请假信息 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假类型 | ● |  |  |  | 区分请假的类型，如病假、事假 |
| 审批状态 | ● |  |  |  | 显示请假申请的审批情况，如已批准、待批准 |

【诊室-归属于-科室】联系的属性如下：

| 诊室-归属于-科室 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 归属时间 | ● |  |  |  | 诊室归属于该科室的起始时间 |

## 验证模型的正确性

验证局部模型中每个联系集的正确性：用结构化自然语言描述每个联系表达的业务规则，用假设数据验证是否存在逻辑矛盾。

### 【医生-所属-科室】

【医生-所属-科室】表达的业务规则如下：

* （1个）医生 - 所属 - （1 个）科室；
* （1个）科室 - 包含 - （多个）医生。

假设的数据如下（含正反例）：

| 参与联系的实体 | | 联系的属性 | 事实判定 |
| --- | --- | --- | --- |
| 医生ID | 科室ID |  |
| 医生1 | 科室A |  | ✔ |
| **医生2** | 科室B |  | ✔ |
| 医生3 | 科室C |  | ✔ |
| 医生3 | 科室A |  | ✔ |

### 【医生-排班在-诊室】

【医生-排班在-诊室】表达的业务规则如下：

* （1个）医生 - 排班在 - （1 个）诊室。
* （1门）诊室 - 被（1 个）医生排班在。

假设的数据如下（含正反例）：

| **参与联系的实体** | | 联系的属性 | | | 事实判定 |
| --- | --- | --- | --- | --- | --- |
| **医生ID** | **诊室ID** |  |  | …… |
| 医生A | 诊室1 |  |  |  | ✔ |
| 医生B | 诊室2 |  |  |  | ✔ |
| 医生D | 诊室0 |  |  |  | ✔ |
| 医生C | 诊室5 |  |  |  | ✔ |
| 医生E | 诊室6 |  |  |  | ✔ |
| 医生T | 诊室8 |  |  |  | ✔ |

### 【医生-产生-请假信息】

【医生-产生-请假信息】表达的业务规则如下：

* （1个）医生 - 产生 - （1 个或多个）请假信息。
* （1门）请假信息 - 由（1 个）医生产生。

假设的数据如下（含正反例）：

| **参与联系的实体** | | 联系的属性 | | | 事实判定 |
| --- | --- | --- | --- | --- | --- |
| **医生ID** | **请假ID** |  |  | …… |
| S1 | C1 |  |  |  | ✔ |
| **S1** | **C2** |  |  |  | ✔ |
| S2 | **C3** |  |  |  | ✔ |

### 【诊室-归属于-科室】

【诊室-归属于-科室】表达的业务规则如下：

* （1个）诊室 - 归属于 - （1 个）科室。。
* （1门）科室 - 拥有 - （多个）诊室。
* 假设的数据如下（含正反例）：

| **参与联系的实体** | | 联系的属性 | | | 事实判定 |
| --- | --- | --- | --- | --- | --- |
| **学生** | **课程** |  |  | …… |
| S1 | C1 |  |  |  | ✔ |
| **S2** | **C2** |  |  |  | ✔ |
| S3 | **C1** |  |  |  | ✔ |
| S4 | C3 |  |  |  | ✔ |

## ER模型中不能表达的业务规则

提示：分析上述模型中不能表达出来的业务规则，例如：【学生-重修课程】是【学生-选修-课程】的子集。

ER模型中不能表达或描述的业务规则如下表。

| 序号 | 约束类别 | 业务规则说明 |
| --- | --- | --- |
| 1 | 实体集内部实体间约束 | 同一科室的医生职称不能完全相同 |
| 2 | 实体集的统计约束 | 一个科室的医生人数不能超过 50 人 |
| 3 | 实体集之间的约束 | 同一时间段内，一个医生不能在不同科室的诊室排班 |
| 4 | 联系集之间的约束 | 【医生 - 请假】是【医生 - 排班】的补集，即请假期间不能有排班 |

# 局部概念建模3：蒋文栋

【说明：蒋文栋是本部分概念建模的负责人】。

## 业务模块的详细描述

【说明：根据自己的理解，详细描述自己负责的业务模块，补充业务简述不足或不够清晰的内容。】

业务模块的详细描述如下表。

表 2 局部业务模块的详细描述

| 编号 | 业务模块名称 | 详细说明 |
| --- | --- | --- |
| B3 | 排班管理模块 | 提高排班效率：手动添加或修改班制，可按部门或员工工作性质为其安排相应班制。当员工工作安排有调整时，也能在已有班制基础上进行修改。对于复杂的多班制情况，系统可通过设置班制组来统一分组整理，方便管理。 |
| B4 | 诊所管理模块 | 患者管理：方便快捷地录入、查看、更新和删除患者信息，包括患者的基本资料、诊疗记录等，有助于医生全面了解患者情况，提供更精准的医疗服务。  医生管理：对医生信息进行有效管理，如添加、查看、更新和删除医生的相关信息，同时能结合排班管理，合理安排医生的工作时间。  优化就诊流程：通过预约挂号管理，患者可以提前选择医生和就诊时间，减少等待时间，提高就诊效率；诊所也能更好地规划资源，确保医疗服务的有序进行。  病历管理：创建、查看、更新和删除病历记录，实现病历的电子化和规范化，方便医生随时查阅患者的病史，为诊断和治疗提供参考。 |

## 分析实体及其属性

### 实体集

业务模块中包含的实体如下表【提示：一个实体只能用一种量词，如果在不同语境中需要使用不同的量词，则说明不是一个概念】。

表 3 业务模块中包含的实体集

| 编号 | 实体 | 量词 | 实体集含义说明（概念定义） |
| --- | --- | --- | --- |
| E1\_1 | 排班 | 个 | 包括医生的日常工作排班、临时排班等各种形式的工作安排 |
| E1\_2 | 诊所 | 个 | 提供医疗服务的场所 |

### 实体的属性

| E1\_1：排班（Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| date |  | ● |  |  |  | 排班的日期 |
| time\_slot |  | ● |  |  |  | 排班的时间段 |
| doctor\_id |  | ● |  |  |  | 负责该排班的医生标识 |

| E1\_2：医生工作时间表（Time\_Table） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| time\_table\_id | ID1 | ● |  |  |  | 时间表的唯一标识 |
| doctor\_id | ID2 | ● |  |  |  | 医生的标识 |
| date |  | ● |  |  |  | 医生工作日期 |
| time\_slot |  | ● |  |  |  | 医生工作时间段 |

| E1\_3：诊所（Clinic） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| clinic\_id | ID1 | ● |  |  |  | 诊所的唯一标识 |
| clinic\_name |  |  |  |  |  | 诊所的名称 |

| E1\_4：诊所排班（Clinic\_Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| clinic\_id |  | ● |  |  |  | 诊所的标识 |
| date |  | ● |  |  |  | 日期 |
| time\_slot |  | ● |  |  |  | 时间段 |
| doctor\_id |  | ● |  |  |  | 医生的标识 |

## 实体集之间的分类关系（父子集关系）

分类关系如下EER图。

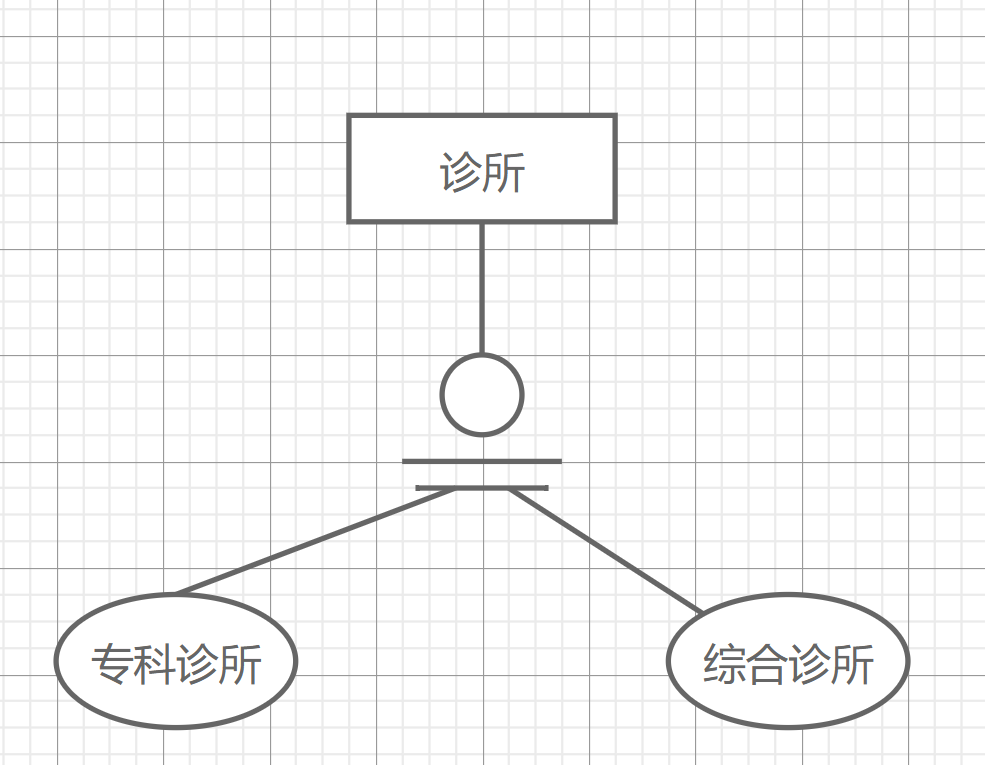


图 1 诊所分类关系图

排班管理模块业务规则如下：

* 完全划分：排班 = 医生排班 ∪ 诊所排班
* 子集互斥：正常排班 ∩ 临时排班 = Φ
* 完全划分：医生排班 = 正常排班 ∪ 临时排班
* 子集互斥：正常排班 ∩ 临时排班 = Φ

诊所管理模块业务规则如下：

* 完全划分：诊所 = 综合诊所 ∪ 专科诊所
* 子集互斥：综合诊所 ∩ 专科诊所 = Φ
* 完全划分：诊所排班 = 工作日排班 ∪ 周末排班
* 子集互斥：工作日排班 ∩ 周末排班 = Φ

## 实体集之间的联系

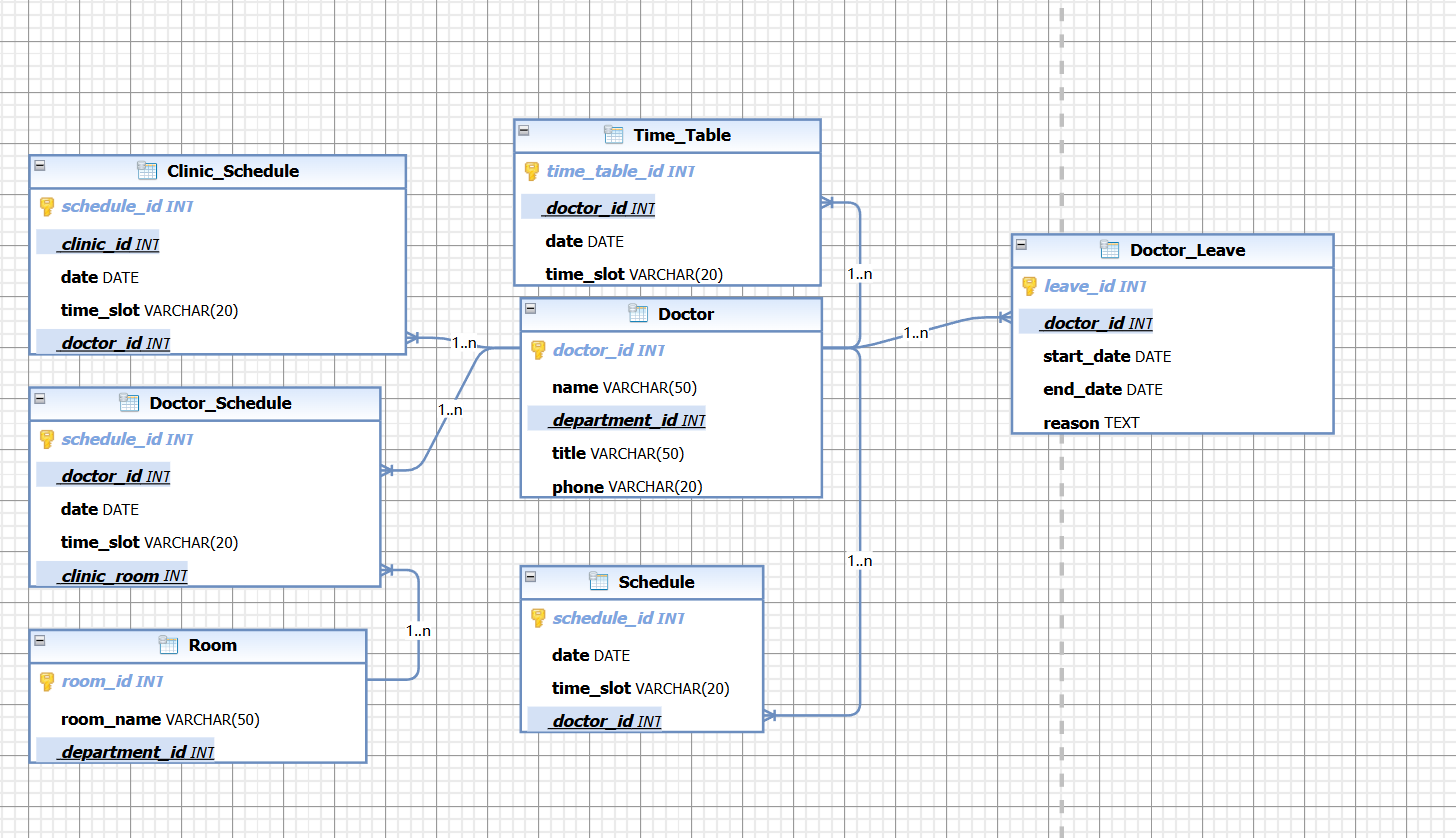
### 实体集之间可能存在的联系矩阵勾核表

实体集之间可能存在的联系矩阵如下表（●表示实体集之间至少存在一种联系）。

|  | 排班 | 医生工作时间表 | 诊所 | 诊所排班 |
| --- | --- | --- | --- | --- |
| 排班 |  |  |  | ● |
| 医生工作时间表 |  |  |  | ● |
| 诊所 |  |  |  | ● |
| 诊所排班 | ● | ● | ● |  |

### 【排班】相关的联系

【排班】相关的联系如下图。

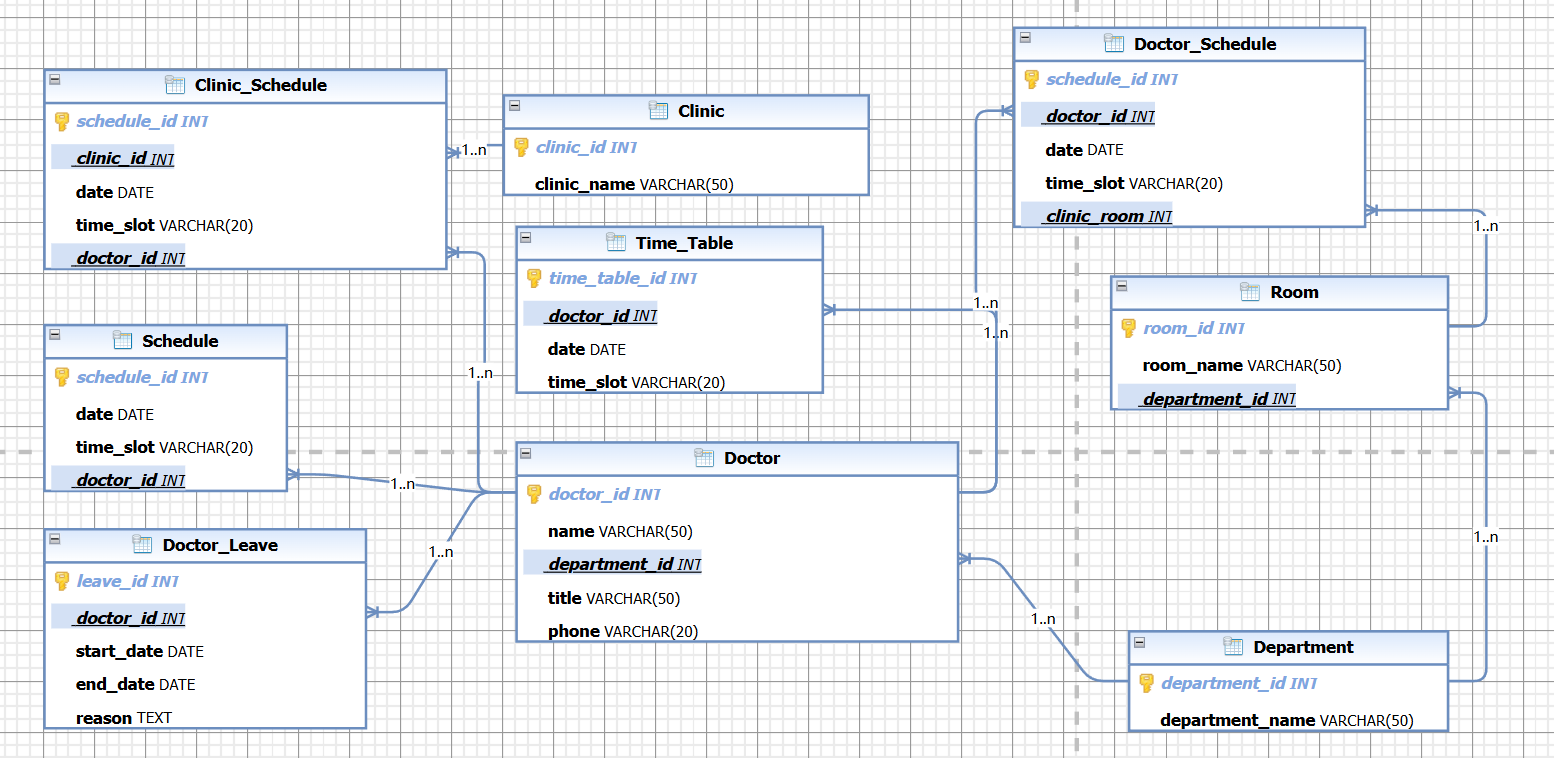


提示：对每个实体集，分析其自身内部对象之间的关系，分析其余其它实体集之间的联系，两个实体集之间的联系可能存在多种，例如学生与课程之间的联系至少存在“初修”、“重修”两种联系。

【有的实体集可能依赖另一实体集才能区别开其实例，这样的实体集是弱实体集，要用弱实体集及弱联系集符号来表达出来】

### 【诊所】相关的联系

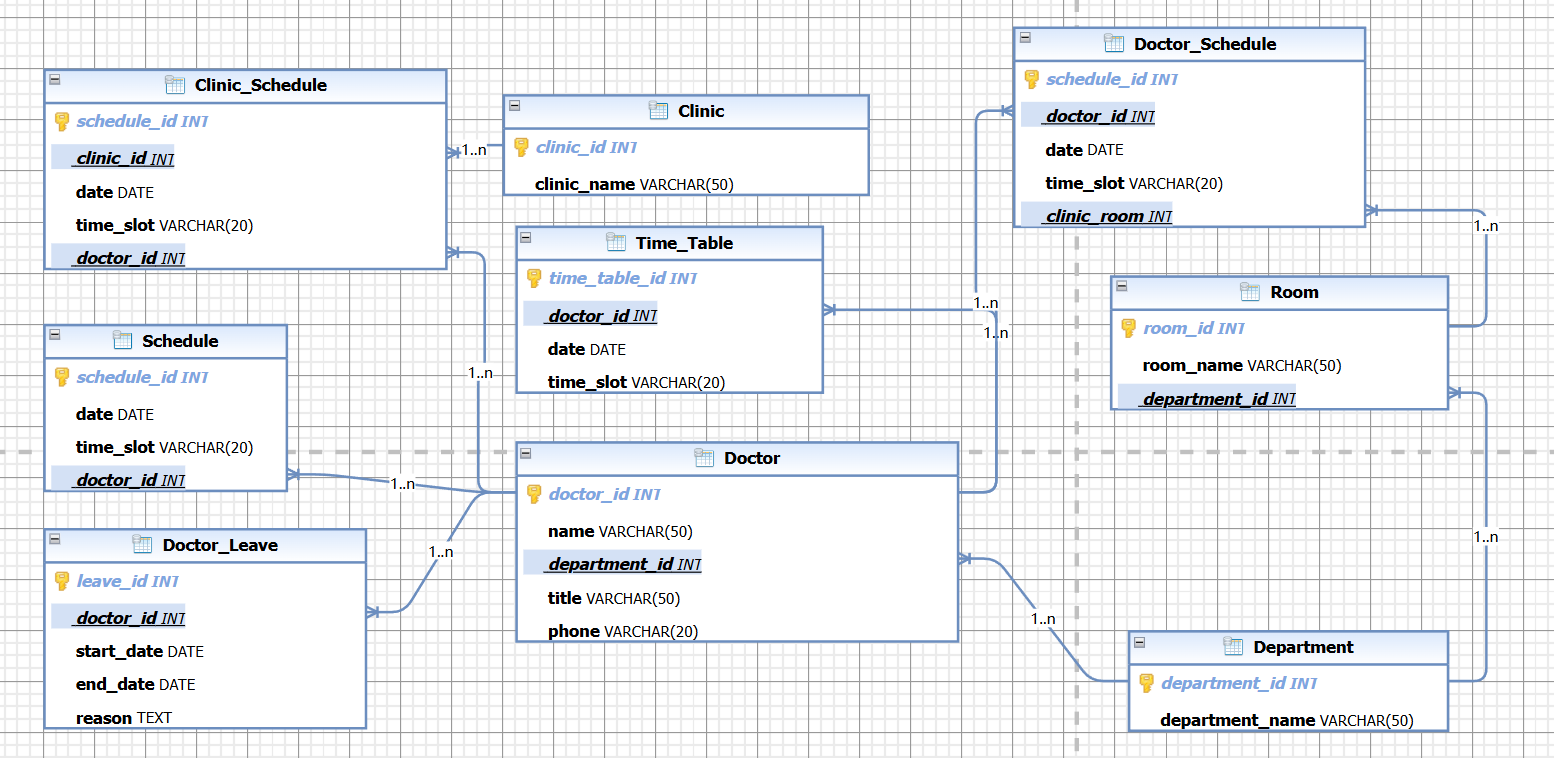
【诊所】相关的联系如下图。



## 集成的局部概念模型

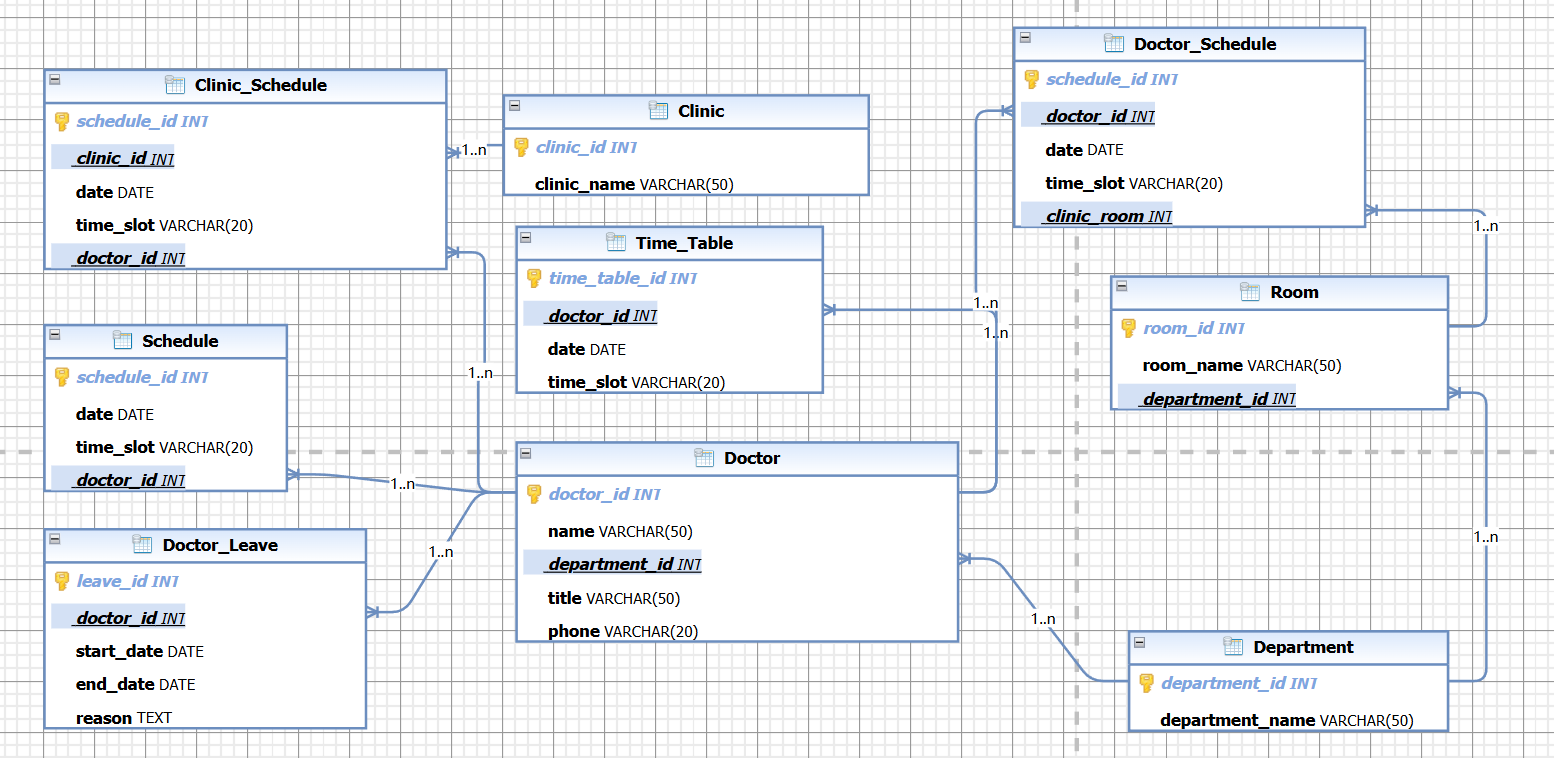
### 集成得到的ER简图

将上面分析出来各个局部的ER模型合并（集成）在一起，形成一个完整的局部概念模型，ER简图如下。



### 去除冗余联系得到的ER简图

分析集成后的局部概念模型中存在冗余的联系，冗余联系是可以由其它联系推导出来的或者说是由其它联系蕴含的联系。以下联系是无冗余的。



### 联系的属性

【医生 - 排班】联系的属性如下：

| 医生 - 排班 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班周期 | ● |  |  |  | 表示医生排班的时间周期，例如按周、按月等，这是必须记录的信息（强制）。 |
| 工作时长 | ● |  |  |  | 明确医生每次排班的工作时间长度，是重要的强制记录项。 |
| 是否加班 | ● |  |  |  | 用于标记医生的此次排班是否属于加班情况，是一个强制且具有固定枚举值（是或否）的属性。 |

【诊所 - 排班】联系的属性如下：

| 诊所 - 排班 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班人数限制 | ● |  |  |  | 规定每次诊所排班所能容纳的患者数量上限，是强制记录的属性，以确保医疗服务的质量和效率。 |
| 特殊要求 | ● |  |  |  | 可能包括需要特定的医疗设备、特殊的环境准备等，是强制记录且可能为多值的属性，以满足诊所排班的特殊条件 |

提示：逐一分析各联系是否有属性，如果有，参照上述格式列出来。

## 验证模型的正确性

验证局部模型中每个联系集的正确性：用结构化自然语言描述每个联系表达的业务规则，用假设数据验证是否存在逻辑矛盾。

### 【医生-排班】

【学院-开设-专业】表达的业务规则如下：

* （1个）医生-（n个）排班；
* （1个）排班-（1个）医生。

假设的数据如下（含正反例）：

| 参与联系的实体 | 联系的属性 | 事实判定 |
| --- | --- | --- |
| 医生 | 排班 |  |
| 李医生 | 周一上午 | ✔ |
| **李医生** | 周三上午 | ✔ |
| 王医生 | 周无上午 | ✔ |
| **王医生** | 周一上午 | ✘ |
| 李医生 | 周五上午 | ✔ |

### 【诊所-排班】

【学生-选修-课程】表达的业务规则如下：

* （1个）诊所-（n个）排班。
* （1个）排班-（n所）诊所。

假设的数据如下（含正反例）：

| **参与联系的实体** | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **诊所** | 排班 |  |
| 仁爱诊所 | 周一至周五全天 | ✔ |
| **仁爱诊所** | 周六上午 | ✔ |
| 惠民诊所 | 周日全天 | ✔ |
| 仁爱诊所 | 周日全天 | ✘ |
| **惠民诊所** | 周一至周五全天 | ✔ |

## ER模型中不能表达的业务规则

提示：分析上述模型中不能表达出来的业务规则，例如：【学生-重修课程】是【学生-选修-课程】的子集。

ER模型中不能表达或描述的业务规则如下表。

| 序号 | 约束类别 | 业务规则说明 |
| --- | --- | --- |
| 1 | 实体集内部实体间约束 | 正常排班和临时排班的时间不能重叠  综合诊所和专科诊所的名称不能相同 |
| 2 | 实体集的统计约束 | 每位医生每天的排班数量不超过一定限度每个诊所的诊室数量有上限 |
| 3 | 实体集之间的约束 | 医生排班需符合医生所在科室的工作时间安排诊所排班需考虑诊所的规模和设施条件 |
| 4 | 联系集之间的约束 | 医生工作时间表中的排班不能与请假时间冲突诊所排班不能与诊所的维修时间冲突 |

# 全局概念模型

项目组全体成员一起，将上述项目成员各自完成概念模型及相关成果进行集成，得到全局概念模型。

## 对整个业务的详细说明

将各局部概念模型报告中对各业务模块的详细描述集成在一起后，经过项目组分析与讨论，确定各业务模块的细节，得出业务详细说明如下表。

表 8 业务模块的详细描述

| 编号 | 业务模块 | 详细说明 |
| --- | --- | --- |
| B1 | 科室管理模块 | 负责管理医院的科室信息，通过“Department”表存储科室的 ID 和名称。确保科室信息的准确记录，为其他相关业务模块提供基础数据支持。 |
| B2 | 医生管理模块 | 对医生的个人信息、所属科室、职称、电话等进行管理。“Doctor”表记录了医生的详细资料，同时与“Department”表关联获取科室信息，与“Doctor\_Schedule”表和“Doctor\_Leave”表关联了解医生的排班和请假情况。 |
| B3 | 诊室管理模块 | 管理诊室的基本信息，包括诊室 ID、名称和所属科室。“Room”表用于存储这些数据，并与“Department”表建立关联。 |
| B4 | 医生排班管理模块 | 要处理医生的排班安排，“Doctor\_Schedule”表记录了医生的排班 ID、日期、时间段和诊室 ID 等，与“Doctor”表和“Room”表关联，实现全面的排班管理。 |
| B5 | 医生请假管理模块 | 用于记录医生的请假信息，“Doctor\_Leave”表包含请假 ID、医生 ID、开始日期、结束日期和请假理由，便于对医生请假进行统筹和安排。 |
| B6 | 患者管理模块 | 管理患者的基本资料，如姓名、性别、年龄、地址和电话等，“Patient”表为核心存储表。 |
| B7 | 病历记录管理模块 | 记录患者的病历信息，包括诊断、处方和就诊日期等。“Medical\_Record”表关联患者和医生，确保病历数据的完整性和准确性。 |
| B8 | 预约管理模块 | 处理患者的预约信息，“Appointment”表记录预约 ID、患者 ID、医生 ID、预约日期和状态，便于安排就诊。 |
| B9 | 药品管理模块 | 管理药品的名称、用法、剂量等信息，“Medication”表作为药品数据的存储载体。 |
| B10 | 处方管理模块 | 记录处方详情，“Prescription”表关联患者、医生和药品，明确处方的开具日期等信息。 |
| B11 | 账单管理模块 | 处理费用相关事务，“Bill”表记录账单 ID、患者 ID、医生 ID、处方 ID 以及各项费用和总费用等。 |
| B12 | 排班管理模块 | 包含排班信息和医生工作时间的管理。“Schedule”表记录具体的排班情况，“Time\_Table”表记录医生的工作时间表。 |
| B13 | 诊所管理模块 | 管理诊所的基本信息，“Clinic”表存储诊所 ID 和名称。“Clinic\_Schedule”表则负责记录诊所的排班信息，与医生和诊所关联。 |

## 实体及其属性

### 实体集

业务中包含的实体如下表【提示：一个实体只能用一种量词，如果在不同语境中需要使用不同的量词，则说明不是一个概念】。

表 3 业务模块中包含的实体集

| 编号 | 实体 | 量词 | 实体集含义说明（概念定义） |
| --- | --- | --- | --- |
| E1\_1 | 科室 | 个 | 包括内科、外科、妇产科、儿科等各种不同职能的医疗科室。 |
| E1\_2 | 医生 | 位 | 从事医疗工作，具有相应职称和专业技能的人员。 |
| E1\_3 | 诊室 | 间 | 医院中供医生为患者进行诊疗的房间 |
| E1\_4 | 患者 | 个 | 到医院寻求医疗服务的个人。 |
| E1\_5 | 病历 | 门 | 记录患者病情、诊断、治疗等信息的文档。 |
| E1\_6 | 药品 | 种 | 用于治疗、预防或诊断疾病的物质 |
| E1\_7 | 处方 | 张 | 医生为患者开具的用药和治疗的书面指示。 |
| E1\_8 | 账单 | 份 | 记录患者医疗费用明细的单据 |
| E1\_9 | 诊所 | 家 | 提供医疗服务的场所。 |
| E1\_10 | 排班 | 次 | 对医生工作时间和任务的安排。 |
| E1\_11 | 请假 | 次 | 医生因特殊原因申请的暂时离岗 |
| E1\_12 | 时间槽 | 段 | 划分的具体时间段，如上午、下午、晚上等。 |

### 实体的属性

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |

| E1\_2医疗记录 (Medical Records) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| record\_id (医疗记录的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于跟踪每条医疗记录。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该医疗记录的患者的唯一标识符。 |
| date (就诊日期) |  | ● |  |  |  | 输入就诊日期，记录患者每次的就诊时间。 |
| diagnosis (诊断信息) |  |  |  |  |  | 输入诊断信息，描述患者的疾病或健康问题。 |
| prescription (处方信息) |  |  |  |  |  | 输入医生开具的处方信息。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该医疗记录的医生的唯一标识符。 |

| E1\_3预约信息 (Appointments) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| appointment\_id (预约信息的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个预约的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该预约的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该预约的医生的唯一标识符。 |
| appointment\_date (预约日期) |  | ● |  |  |  | 输入预约的日期和时间。 |
| status (预约状态，如预约成功或取消) |  |  |  |  |  | 输入预约的状态，例如预约成功或取消。 |

| E1\_4药品 (Medications) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| medication\_id (药品的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每种药品的唯一标识。 |
| medication\_name (药品的名称) |  | ● |  |  |  | 输入药品的名称。 |
| usage\_instructions (药品的用法说明) |  | ● |  |  |  | 输入药品的使用说明，如用药频次和方法。 |
| dosage (药品的剂量信息) |  | ● |  |  |  | 输入药品的剂量信息，如每次服用的量。 |

| E1\_5处方 (Prescriptions) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| prescription\_id (处方的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个处方的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该处方的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | *关联开具该处方的医生的唯一标识符。* |
| medication\_id (关联的药品ID) |  | ● |  |  |  | 关联的药品ID |
| prescription\_date (处方开具日期) |  | ● |  |  |  | 输入处方开具的日期。 |

| E1\_6账单 (Bills) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| bill\_id (账单的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每张账单的管理和跟踪。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该账单的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联的医生ID |
| prescription\_id (关联的处方ID) |  | ● |  |  |  | 关联的处方ID |
| consultation\_fee (诊疗费用) |  |  |  |  |  | 输入与诊疗相关的费用。 |
| medication\_fee (药品费用) |  |  |  |  |  | 输入与药品相关的费用。 |
| total\_amount (总费用) |  |  |  |  |  | 输入账单的总费用。 |
| bill\_date (账单日期) |  | ● |  |  |  | 输入账单的日期。 |

| E1\_7：科室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 科室ID | ID1 | ● |  |  |  | 科室的唯一标识符 |
| 科室名称 |  | ● |  |  |  | 科室的具体名称 |

| E1\_8：医生 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 医生ID | ID1 | ● |  |  |  | 医生的唯一标识符 |
| 姓名 |  | ● |  |  |  | 医生的称呼 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |
| 职称 |  | ● |  |  |  | 医生的专业职称 |
| 电话 |  |  |  |  |  | 医生的联系电话 |

| E1\_9：诊室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 诊室ID | ID1 | ● |  |  |  | 诊室的唯一标识符 |
| 诊室名称 |  | ● |  |  |  | 诊室的具体名称 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |

| E1\_10：医生排班 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班ID | ID1 | ● |  |  |  | 医生排班的唯一标识符 |
| 医生ID |  | ● |  |  |  | 参与排班的医生的标识符 |
| 日期 |  | ● |  |  |  | 排班的具体日期 |
| 时间段 |  | ● |  |  |  | 排班的时间区间 |
| 诊室ID |  | ● |  |  |  | 排班对应的诊室标识符 |

| E1\_11：医生请假 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假ID | ID1 | ● |  |  |  | 医生请假的唯一标识符 |
| 医生ID |  | ● |  |  |  | 请假医生的标识符 |
| 开始日期 |  | ● |  |  |  | 请假的起始日期 |
| 结束日期 |  | ● |  |  |  | 请假的结束日期 |
| 请假理由 |  |  |  |  |  | 医生请假的原因说明 |

## 实体集之间的分类关系（父子集关系）

科室分类关系如下EER图。

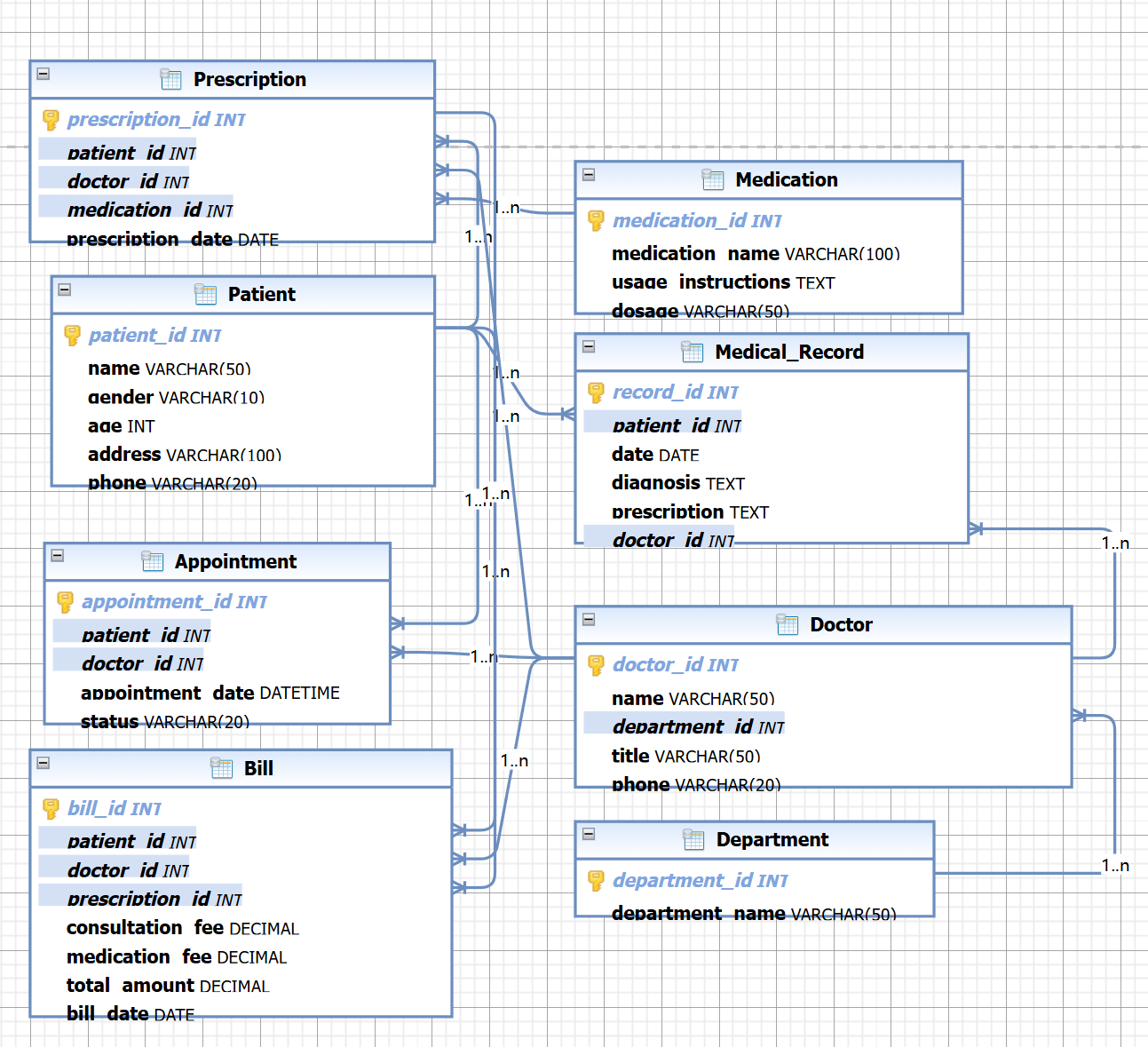


图 1 科室分类关系图

科室分类关系的业务规则如下：

* 科室 = 内科 ∪ 外科 ∪ 妇产科 ∪ 儿科 ∪ 其他科室（如五官科、皮肤科等）
* 子集互斥：内科 ∩ 外科 ∩ 妇产科 ∩ 儿科 ∩ 其他科室 = Φ

医生分类关系的业务规则如下：

* 完全划分：医生 = 主任医师 ∪ 副主任医师 ∪ 主治医师 ∪ 住院医师留学生 ∩ 本国 = Φ
* 完全划分：主任医师 ∩ 副主任医师 ∩ 主治医师 ∩ 住院医师 = Φ

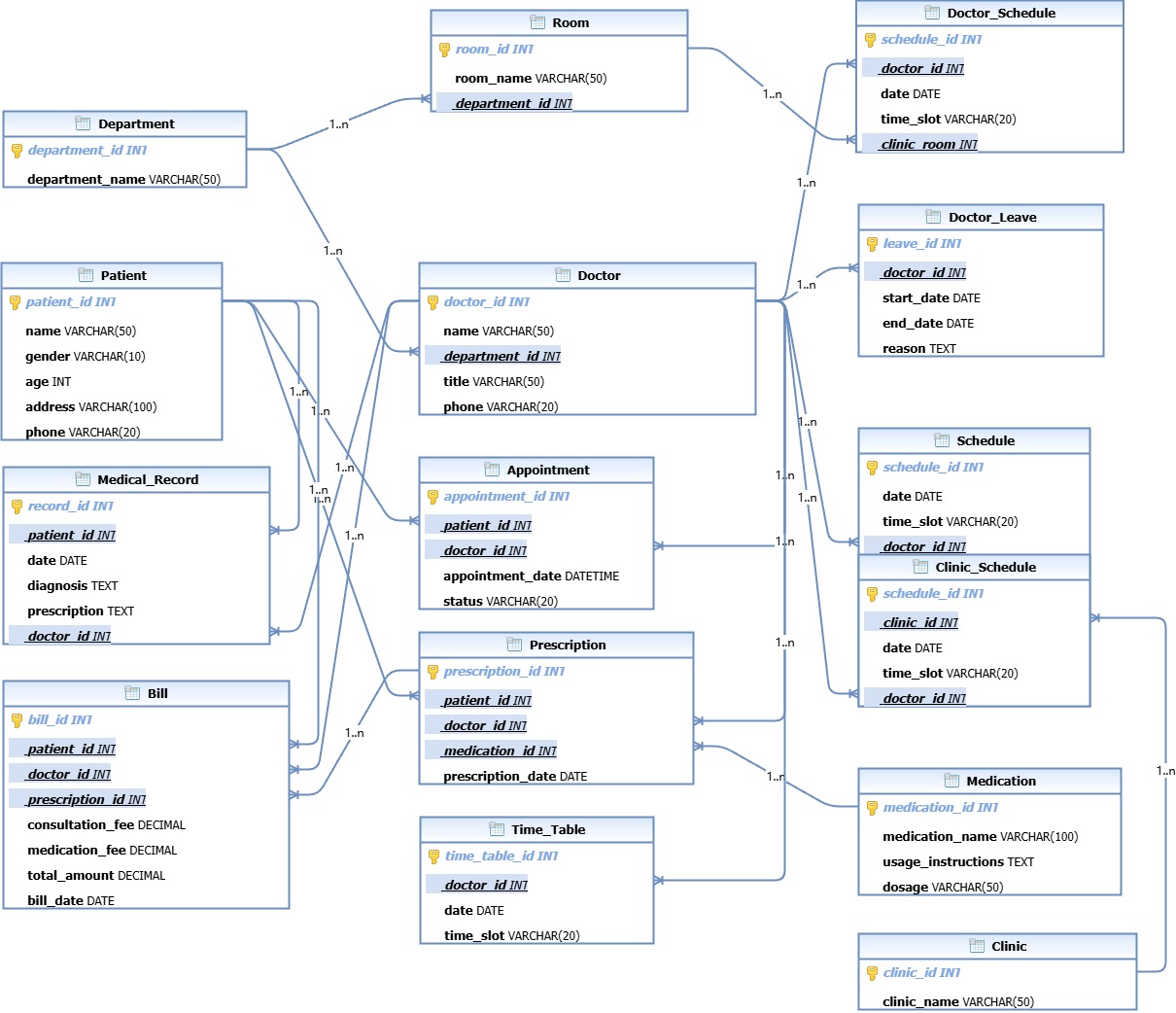
患者分类关系的业务规则如下：

* 完全划分：患者 = 成人患者 ∪ 儿童患者
* 完全划分：成人患者 ∩ 儿童患者 = Φ

## 全局概念模型

### ER简图

将上面分析出来各个局部的ER模型合并（集成）在一起，形成一个完整的全局概念模型，ER简图如下。

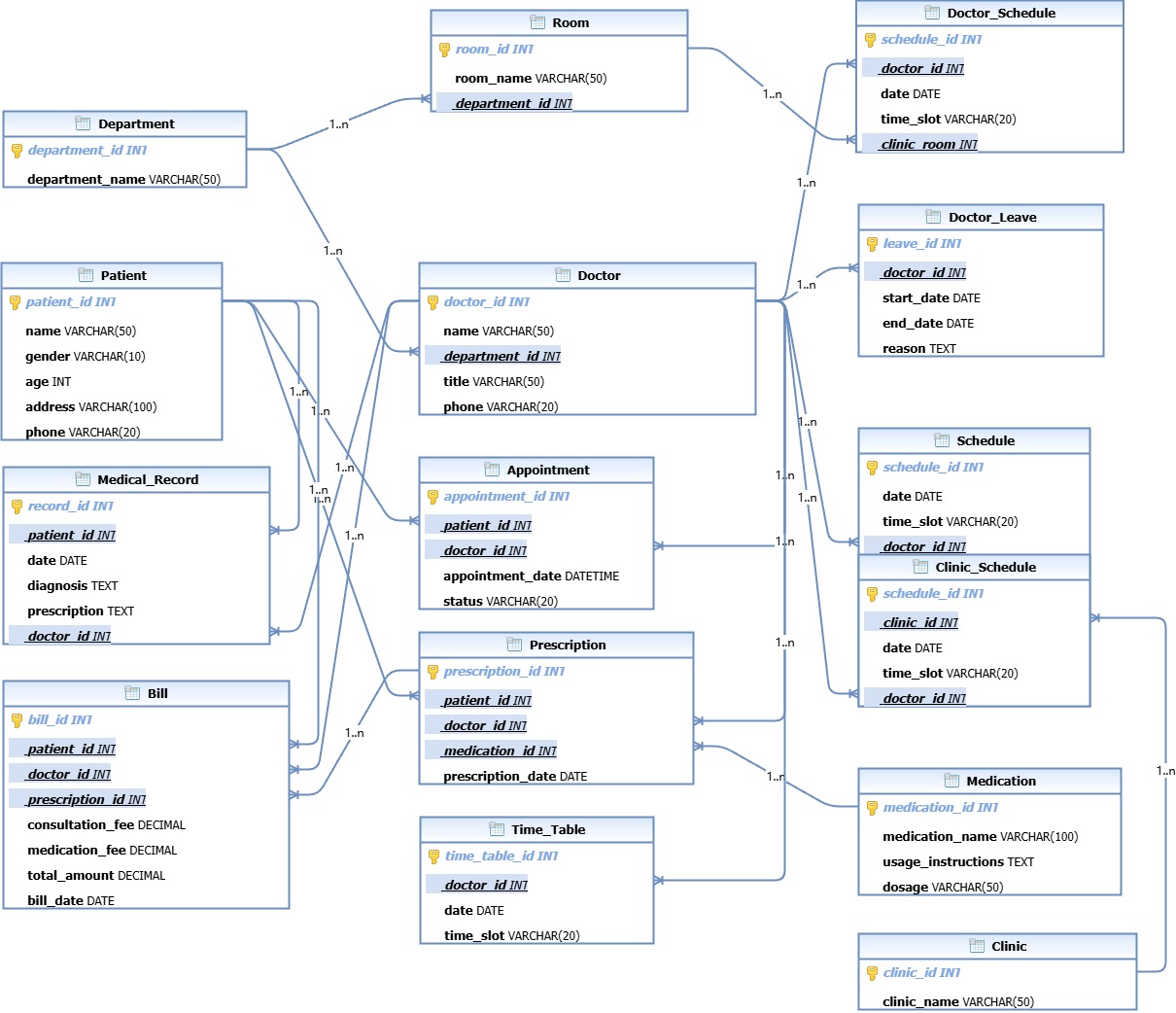


### 去除冗余联系后的ER简图

分析集成后的局部概念模型中存在冗余的联系，冗余联系是可以由其它联系推导出来的或者说是由其它联系蕴含的联系。以下联系是冗余的：

* 【医生-排班-诊室】：【（1 个）医生 - 排班 - （n 个）诊室】可由【（1 个）医生 - 排班 - （n 个）时间】和【（1 个）诊室 - 分配 - （n 个）时间】这两个联系推导出来；【（1 个）诊室 - 属于 - （1 个）医生】可由【（1 个）诊室 - 属于 - （1 个）科室】、【（1 个）科室 - 包含 - （1 个）医生】这两个联系推导出来。
* 【诊所-排班-医生】：【（1 个）诊所 - 排班 - （n 个）医生】可由【（1 个）诊所 - 包含 - （n 个）诊室】、【（1 个）诊室 - 分配 - （n 个）医生】这两个联系推导出来；【（1 个）医生 - 属于 - （1 个）诊所】可由【（1 个）医生 - 属于 - （1 个）科室】、【（1 个）科室 - 位于 - （1 个）诊所】这两个联系推导出来。
* 【医生-请假-时间】：【（1 个）医生 - 请假 - （n 个）时间】可由【（1 个）医生 - 排班 - （n 个）时间】推导出来；【（1 个）时间 - 属于 - （1 个）医生的请假时段】可由【（1 个）时间 - 属于 - （1 个）医生的排班时段】推导出来。

去除冗余联系后的ER简图如下。



### 联系的属性

【医生 - 排班 - 诊室】联系的属性如下：

| 医生 - 排班 - 诊室 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班日期 | ● |  |  | ● | 具体的排班日期 |
| 时间段 | ● |  |  | ● | 如上午、下午、晚上等 |
| 诊室编号 | ● |  |  | ● | 所排诊室的唯一编号 |

【诊所 - 排班 - 医生】联系的属性如下：

| 诊所 - 排班 - 医生 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班日期 | ● |  |  | ● | 具体的排班日期 |
| 时间段 | ● |  |  | ● | 如上午、下午、晚上等 |
| 医生编号 | ● |  |  | ● | 所排医生的唯一编号 |

【医生 - 请假 - 时间】联系的属性如下：

| 医生 - 请假 - 时间 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假开始日期 | ● |  |  | ● | 请假的起始日期 |
| 请假结束日期 | ● |  |  | ● | 请假的结束日期 |
| 请假原因 | ● |  |  | ● | 说明请假的具体原因 |

【医生 - 开处方 - 患者】联系的属性如下：

| 医生 - 开处方 - 患者 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 处方编号 | ● |  |  | ● | 处方的唯一编号 |
| 开具日期 | ● |  |  | ● | 处方开具的时间 |

【患者 - 预约 - 医生】联系的属性如下：

| 患者 - 预约 - 医生 | | | | | |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 预约编号 | ● |  |  | ● | 预约的唯一标识 |
| 预约时间 | ● |  |  | ● | 预约的具体时刻 |
| 预约状态 | ● |  |  | ● | 如已确认、已取消等 |

提示：逐一分析各联系是否有属性，如果有，参照上述格式列出来。

## 验证模型的正确性

验证局部模型中每个联系集的正确性：用结构化自然语言描述每个联系表达的业务规则，用假设数据验证是否存在逻辑矛盾。

### 【医生-排班-诊室】

【医生 - 排班 - 诊室】表达的业务规则如下：

* （1 个）医生 - 排班 - （n 个）诊室；
* （1 个）诊室 - 被安排给 - （1 个）医生。

假设的数据如下（含正反例）：

| 参与联系的实体 | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **医生** | 诊室 |  |
| 李医生 | 1 号诊室 | ✔ |
| 王医生 | 2号诊室 | ✔ |
| 赵医生 | 3号诊室 | ✔ |
| 李医生 | 4号诊室 | ✔ |
| 李医生 | 2 号诊室 （同一医生重复安排） | ✘ |
| 孙医生 | 1 号诊室 | ✘ |

### 【诊所-排班-医生】

【【诊所 - 排班 - 医生】】表达的业务规则如下：

* （1 个）诊所 - 排班 - （n 个）医生；
* （1 个）医生 - 被诊所排班 - （1 个）诊所。

假设的数据如下（含正反例）：

| 参与联系的实体 | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **诊所** | **医生** |  |
| 仁爱诊所 | 张医生 | ✔ |
| 康复诊所 | 李医生 | ✔ |
| 仁爱诊所 | 刘医生 | ✔ |
| 仁爱诊所 | 刘医生 （同一诊所重复安排） | ✘ |

### 【医生-请假-时间】

【医生 - 请假 - 时间】表达的业务规则如下：

* （1 个）医生 - 请假 - （1 个）时间段；
* （1 个）时间段 - 属于 - （1 个）医生的请假时间。

假设的数据如下（含正反例）：

| 参与联系的实体 | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **医生** | **请假时间** |  |
| 陈医生 | 2024-07-01 至 2024-07-05 | ✔ |
| 吴医生 | 2024-07-10 至 2024-07-15 | ✔ |
| 陈医生 | 2024-07-03 至 2024-07-07 （同一医生请假时间冲突） | ✘ |
| 郑医生 | 2024-07-01 至 2024-07-05 | ✘ |

### 【医生-开处方-患者】

【学生-选修-课程】表达的业务规则如下：

* （1 个）医生 - 开处方 - （n 个）患者；
* （1 个）处方 - 属于 - （1 个）患者。

假设的数据如下（含正反例）：

| **参与联系的实体** | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **医生** | 患者 |  |
| 王医生 | 李四 | ✔ |
| 赵医生 | 王五 | ✔ |
| 李医生 | 李四 （同一医生对同一患者重复开处方） | ✘ |
| 孙医生 | 张三 | ✘ |

### 【患者-预约-医生】

【学生-选修-课程】表达的业务规则如下：

* （1 个）患者 - 预约 - （n 个）医生；
* （1 个）预约 - 属于 - （1 个）患者。

假设的数据如下（含正反例）：

| 参与联系的实体 | 联系的属性 | 事实判定 |
| --- | --- | --- |
| **患者** | **医生** |  |
| 张三 | 李医生 | ✔ |
| 李四 | 王医生 | ✔ |
| 王五 | 赵医生 | ✔ |
| 王五 | 李医生 （同一患者重复预约） | ✘ |
| 赵六 | 孙医生 | ✘ |

## ER模型中不能表达的业务规则

提示：分析上述模型中不能表达出来的业务规则，例如：【学生-重修课程】是【学生-选修-课程】的子集。

ER模型中不能表达或描述的业务规则如下表。

| 序号 | 约束类别 | 业务规则说明 |
| --- | --- | --- |
| 1 | 实体集内部实体之间的约束 | 同一医生在一天内的接诊患者数量不能超过 30 人。若超过，系统将自动提示并拒绝为新患者安排该医生的就诊。 |
| 2 | 实体集的统计约束 | 一个诊室在一周内的使用次数不能超过 50 次。这是为了保证诊室有足够的时间进行清洁和设备维护。 |
| 3 | 实体集之间的约束 | 同一患者在一个月内不能在同一家诊所的不同科室就诊超过 5 次。此规则旨在防止过度医疗和资源浪费。 |
| 4 | 联系集之间的约束 | 【医生 - 请假】是【医生 - 排班】的特殊情况，只有在有排班的前提下才能请假。医生请假需要提前 3 天申请，且请假期间的排班将自动取消，重新分配给其他医生。 |

# 逻辑结构设计

## 实体集转换为关系（表）

每个实体集转换为一个关系（表）。**多步处理后最终得到的表名前加\*。**

### 表：患者

**第1步：**实体集转换为表。注意：实体集是子集的转换规则。

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |

**第2步：对上一步的每个表，**处理多值属性后表如下。

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |

**第3步：对上一步得到的每个表，**处理复合属性后表如下。

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |

### 表：医疗记录

| E1\_2医疗记录 (Medical Records) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| record\_id (医疗记录的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于跟踪每条医疗记录。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该医疗记录的患者的唯一标识符。 |
| date (就诊日期) |  | ● |  |  |  | 输入就诊日期，记录患者每次的就诊时间。 |
| diagnosis (诊断信息) |  |  |  |  |  | 输入诊断信息，描述患者的疾病或健康问题。 |
| prescription (处方信息) |  |  |  |  |  | 输入医生开具的处方信息。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该医疗记录的医生的唯一标识符。 |

### 表：预约信息

| E1\_3预约信息 (Appointments) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| appointment\_id (预约信息的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个预约的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该预约的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该预约的医生的唯一标识符。 |
| appointment\_date (预约日期) |  | ● |  |  |  | 输入预约的日期和时间。 |
| status (预约状态，如预约成功或取消) |  |  |  |  |  | 输入预约的状态，例如预约成功或取消。 |

### 表：药品

| E1\_4药品 (Medications) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| medication\_id (药品的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每种药品的唯一标识。 |
| medication\_name (药品的名称) |  | ● |  |  |  | 输入药品的名称。 |
| usage\_instructions (药品的用法说明) |  | ● |  |  |  | 输入药品的使用说明，如用药频次和方法。 |
| dosage (药品的剂量信息) |  | ● |  |  |  | 输入药品的剂量信息，如每次服用的量。 |

### 表：处方

| E1\_5处方 (Prescriptions) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| prescription\_id (处方的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个处方的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该处方的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | *关联开具该处方的医生的唯一标识符。* |
| medication\_id (关联的药品ID) |  | ● |  |  |  | 关联的药品ID |
| prescription\_date (处方开具日期) |  | ● |  |  |  | 输入处方开具的日期。 |

### 表：账单

| E1\_6账单 (Bills) | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | | 标识 | | 强制 | | 多值 | | 复合 | | 派生 | | 说明 |
| bill\_id (账单的唯一标识符) | | ID1 | | ● | |  | |  | |  | | 提供一个唯一的标识符，用于每张账单的管理和跟踪。 |
| patient\_id (关联的患者ID) | |  | | ● | |  | |  | |  | | 关联该账单的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) | |  | | ● | |  | |  | |  | | 关联的医生ID |
| prescription\_id (关联的处方ID) | |  | | ● | |  | |  | |  | | 关联的处方ID |
| consultation\_fee (诊疗费用) | |  | |  | |  | |  | |  | | 输入与诊疗相关的费用。 |
| medication\_fee (药品费用) | |  | |  | |  | |  | |  | | 输入与药品相关的费用。 |
| total\_amount (总费用) | |  | |  | |  | |  | |  | | 输入账单的总费用。 |
| bill\_date (账单日期) | |  | | ● | |  | |  | |  | | 输入账单的日期。 |
| E1\_1：科室 | | | | | | | | | | | | |
| 属性名称 | 标识 | | 强制 | | 多值 | | 复合 | | 派生 | | 说明 | |
| 科室ID | ID1 | | ● | |  | |  | |  | | 科室的唯一标识符 | |
| 科室名称 |  | | ● | |  | |  | |  | | 科室的具体名称 | |

### 表：医生

| E1\_2：医生 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 医生ID | ID1 | ● |  |  |  | 医生的唯一标识符 |
| 姓名 |  | ● |  |  |  | 医生的称呼 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |
| 职称 |  | ● |  |  |  | 医生的专业职称 |
| 电话 |  |  |  |  |  | 医生的联系电话 |

### 表：诊室

| E1\_3：诊室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 诊室ID | ID1 | ● |  |  |  | 诊室的唯一标识符 |
| 诊室名称 |  | ● |  |  |  | 诊室的具体名称 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |

### 表：医生排班

| E1\_4：医生排班 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班ID | ID1 | ● |  |  |  | 医生排班的唯一标识符 |
| 医生ID |  | ● |  |  |  | 参与排班的医生的标识符 |
| 日期 |  | ● |  |  |  | 排班的具体日期 |
| 时间段 |  | ● |  |  |  | 排班的时间区间 |
| 诊室ID |  | ● |  |  |  | 排班对应的诊室标识符 |

### 表：医生请假

| E1\_5：医生请假 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假ID | ID1 | ● |  |  |  | 医生请假的唯一标识符 |
| 医生ID |  | ● |  |  |  | 请假医生的标识符 |
| 开始日期 |  | ● |  |  |  | 请假的起始日期 |
| 结束日期 |  | ● |  |  |  | 请假的结束日期 |
| 请假理由 |  |  |  |  |  | 医生请假的原因说明 |

### 表：排班

| E1\_1：排班（Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| date |  | ● |  |  |  | 排班的日期 |
| time\_slot |  | ● |  |  |  | 排班的时间段 |
| doctor\_id |  | ● |  |  |  | 负责该排班的医生标识 |

### 表：医生工作时间表

| E1\_2：医生工作时间表（Time\_Table） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| time\_table\_id | ID1 | ● |  |  |  | 时间表的唯一标识 |
| doctor\_id | ID2 | ● |  |  |  | 医生的标识 |
| date |  | ● |  |  |  | 医生工作日期 |
| time\_slot |  | ● |  |  |  | 医生工作时间段 |

### 表：诊所

| E1\_3：诊所（Clinic） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| clinic\_id | ID1 | ● |  |  |  | 诊所的唯一标识 |
| clinic\_name |  |  |  |  |  | 诊所的名称 |

### 表：诊所排班

| E1\_4：诊所排班（Clinic\_Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| clinic\_id |  | ● |  |  |  | 诊所的标识 |
| date |  | ● |  |  |  | 日期 |
| time\_slot |  | ● |  |  |  | 时间段 |
| doctor\_id |  | ● |  |  |  | 医生的标识 |

## 联系集转换为表

每个联系集转换为一个表。**多步处理后最终得到的表名前加\*。**

### 科室-包含-医生

**第1步：**联系集转换为表。

| **科室-包含-医生（联系）** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 科室.主码 | department\_id FK1 |  | FK1 | ● |  |  |  |  |
| 医生.主码 | doctor\_id PK FK2 | PK | FK2 | ● |  |  |  |  |
| 联系的属性 |  |  |  |  |  |  |  |  |

### 医生-排班-诊室

**第1步：**联系集转换为表。

| **医生-排班-诊室** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 医生.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 诊室.主码 | room\_id | PK | FK2 | ● |  |  |  |  |

### 医生-记录-病历

联系集【医生-记录-病历】转换为如下表。

| **医生-记录-病历** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 医生.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 病历.主码 | record\_id |  | FK2 | ● |  |  |  |  |

### 医生-请假-时间

联系集【医生-请假-时间】转换为如下表。

| **医生-请假-时间** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 医生.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 请假.主码 | record\_id |  | FK2 | ● |  |  |  |  |

### 患者-拥有-病历

联系集【患者-拥有-病历】转换为如下表。

| **患者-拥有-病历** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 医生.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 病历.主码 | record\_id |  | FK2 | ● |  |  |  |  |

### 患者-预约-医生

联系集【患者-预约-医生】转换为如下表。

| **患者-预约-医生** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 患者.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 医生.主码 | record\_id |  | FK2 | ● |  |  |  |  |

### 医生-开具-处方

联系集【医生-开具-处方】转换为如下表。

| **医生-开具-处方** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 医生.主码 | doctor\_id | PK | FK1 | ● |  |  |  |  |
| 处方.主码 | record\_id |  | FK2 | ● |  |  |  |  |

### 处方-包含-药品

联系集【处方-包含-药品】转换为如下表。

| **处方-包含-药品** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 处方.主码 | prescription\_id | PK | FK1 | ● |  |  |  |  |
| 药品.主码 | medication\_id |  | FK2 | ● |  |  |  |  |

### 患者-产生-账单

联系集【患者-产生-账单】转换为如下表。

| **患者-产生-账单** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性来源 | 属性名称 | PK  CK | FK | 非空 | 多值 | 复合 | 派生 | 说明 |
| 患者.主码 | patient\_id | PK | FK1 | ● |  |  |  |  |
| 账单.主码 | bill\_id |  | FK2 | ● |  |  |  |  |

## 优化关系模式

对上述得到的关系（表）进行优化。写出优化的原因。

### 垂直拆分表

逐一分析需要对那些表进行垂直拆分，写出拆分后表的关系模式。如果有则写出拆分理由及拆分的结果，如果无则写

无

### 无水平拆分表

逐一分析需要对那些表进行水平拆分，写出拆分后表的关系模式。如果有则写出拆分理由及拆分的结果，如果无则写无。

无

### 合并表

如果有则写出合并理由及合并的结果，如果无则写无。

无

## 优化后的关系模式

逐一列出优化后得到的所有关系模式（表）。

无

# 逻辑结构中不能表达的业务规则

逻辑结构中不能表达的业务规则如下表。

| 序号 | 业务规则（约束）说明 |
| --- | --- |
| 1 | 预约：患者预约后，若要取消，必须提前 24 小时。节假日期间不接受普通疾病的预约。 |
| 2 | 处方：同一位医生不能为同一患者在一周内开具超过 3 次处方，对于某些特殊药品，只有特定职称的医生才有开具权限。儿童患者的处方剂量不能超过成人剂量的一定比例。 |
| 3 | 账单：患者累计欠费达到一定金额时，将暂停其后续的医疗服务 |
| 4 | 病历记录：患者的病历只有在经过上级医生审核后才能归档。 |
| 5 | 患者：患者连续 3 次爽约，将被列入黑名单，半年内不能进行预约。 |
| 6 | 请假：医生在一个月内的请假次数不能超过 2 次。 |
| 7 | 医生的职称晋升必须满足一定的工作年限和业绩要求。 |
| 8 | 诊室：诊所的同一诊室在同一时间段只能安排一位医生就诊。  排班：同一医生在一周内的排班时间不能超过 40 小时 |

# 在MySQL平台中实施的数据库设计

在MySQL平台中实施的数据库设计如下。

注意：表的主码不是整型的，可以新增一个整型自动序号列AID作为主码，表原来的主码成为候选码（唯一键UK）。

## 表

| E1\_1：患者 (Patients) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| patient\_id (患者的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于识别每个患者。 |
| name (患者的姓名) |  | ● |  |  |  | 输入患者的姓名，确保准确记录每位患者的身份信息。 |
| gender(患者的性别，枚举值为 {男, 女}) |  | ● |  |  |  | 选择患者的性别，可选值为 {男, 女}。 |
| age (患者的年龄) |  | ● |  |  |  | 输入患者的年龄，这是一个衍生值，可由出生日期计算得出。 |
| address (患者的地址) |  | ● |  |  |  | 输入患者的住址，支持多个地址的情况。 |
| phone (患者的电话号码) | ID2 | ● |  |  |  | 输入患者的电话号码，用于联系或确认身份。 |

| E1\_2医疗记录 (Medical Records) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| record\_id (医疗记录的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于跟踪每条医疗记录。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该医疗记录的患者的唯一标识符。 |
| date (就诊日期) |  | ● |  |  |  | 输入就诊日期，记录患者每次的就诊时间。 |
| diagnosis (诊断信息) |  |  |  |  |  | 输入诊断信息，描述患者的疾病或健康问题。 |
| prescription (处方信息) |  |  |  |  |  | 输入医生开具的处方信息。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该医疗记录的医生的唯一标识符。 |

| E1\_3预约信息 (Appointments) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| appointment\_id (预约信息的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个预约的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该预约的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | 关联该预约的医生的唯一标识符。 |
| appointment\_date (预约日期) |  | ● |  |  |  | 输入预约的日期和时间。 |
| status (预约状态，如预约成功或取消) |  |  |  |  |  | 输入预约的状态，例如预约成功或取消。 |

| E1\_4药品 (Medications) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| medication\_id (药品的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每种药品的唯一标识。 |
| medication\_name (药品的名称) |  | ● |  |  |  | 输入药品的名称。 |
| usage\_instructions (药品的用法说明) |  | ● |  |  |  | 输入药品的使用说明，如用药频次和方法。 |
| dosage (药品的剂量信息) |  | ● |  |  |  | 输入药品的剂量信息，如每次服用的量。 |

| E1\_5处方 (Prescriptions) | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| prescription\_id (处方的唯一标识符) | ID1 | ● |  |  |  | 提供一个唯一的标识符，用于每个处方的追踪和管理。 |
| patient\_id (关联的患者ID) |  | ● |  |  |  | 关联该处方的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) |  | ● |  |  |  | *关联开具该处方的医生的唯一标识符。* |
| medication\_id (关联的药品ID) |  | ● |  |  |  | 关联的药品ID |
| prescription\_date (处方开具日期) |  | ● |  |  |  | 输入处方开具的日期。 |

| E1\_6账单 (Bills) | | | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | | 标识 | | 强制 | | 多值 | | 复合 | | 派生 | | 说明 |
| bill\_id (账单的唯一标识符) | | ID1 | | ● | |  | |  | |  | | 提供一个唯一的标识符，用于每张账单的管理和跟踪。 |
| patient\_id (关联的患者ID) | |  | | ● | |  | |  | |  | | 关联该账单的患者的唯一标识符。 |
| doctor\_id (关联的医生ID) | |  | | ● | |  | |  | |  | | 关联的医生ID |
| prescription\_id (关联的处方ID) | |  | | ● | |  | |  | |  | | 关联的处方ID |
| consultation\_fee (诊疗费用) | |  | |  | |  | |  | |  | | 输入与诊疗相关的费用。 |
| medication\_fee (药品费用) | |  | |  | |  | |  | |  | | 输入与药品相关的费用。 |
| total\_amount (总费用) | |  | |  | |  | |  | |  | | 输入账单的总费用。 |
| bill\_date (账单日期) | |  | | ● | |  | |  | |  | | 输入账单的日期。 |
| E1\_1：科室 | | | | | | | | | | | | |
| 属性名称 | 标识 | | 强制 | | 多值 | | 复合 | | 派生 | | 说明 | |
| 科室ID | ID1 | | ● | |  | |  | |  | | 科室的唯一标识符 | |
| 科室名称 |  | | ● | |  | |  | |  | | 科室的具体名称 | |

| E1\_2：医生 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 医生ID | ID1 | ● |  |  |  | 医生的唯一标识符 |
| 姓名 |  | ● |  |  |  | 医生的称呼 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |
| 职称 |  | ● |  |  |  | 医生的专业职称 |
| 电话 |  |  |  |  |  | 医生的联系电话 |

| E1\_3：诊室 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 诊室ID | ID1 | ● |  |  |  | 诊室的唯一标识符 |
| 诊室名称 |  | ● |  |  |  | 诊室的具体名称 |
| 科室ID |  | ● |  |  |  | 所属科室的标识符 |

| E1\_4：医生排班 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 排班ID | ID1 | ● |  |  |  | 医生排班的唯一标识符 |
| 医生ID |  | ● |  |  |  | 参与排班的医生的标识符 |
| 日期 |  | ● |  |  |  | 排班的具体日期 |
| 时间段 |  | ● |  |  |  | 排班的时间区间 |
| 诊室ID |  | ● |  |  |  | 排班对应的诊室标识符 |

| E1\_5：医生请假 | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| 请假ID | ID1 | ● |  |  |  | 医生请假的唯一标识符 |
| 医生ID |  | ● |  |  |  | 请假医生的标识符 |
| 开始日期 |  | ● |  |  |  | 请假的起始日期 |
| 结束日期 |  | ● |  |  |  | 请假的结束日期 |
| 请假理由 |  |  |  |  |  | 医生请假的原因说明 |

| E1\_1：排班（Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| date |  | ● |  |  |  | 排班的日期 |
| time\_slot |  | ● |  |  |  | 排班的时间段 |
| doctor\_id |  | ● |  |  |  | 负责该排班的医生标识 |

| E1\_2：医生工作时间表（Time\_Table） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| time\_table\_id | ID1 | ● |  |  |  | 时间表的唯一标识 |
| doctor\_id | ID2 | ● |  |  |  | 医生的标识 |
| date |  | ● |  |  |  | 医生工作日期 |
| time\_slot |  | ● |  |  |  | 医生工作时间段 |

| E1\_3：诊所（Clinic） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| clinic\_id | ID1 | ● |  |  |  | 诊所的唯一标识 |
| clinic\_name |  |  |  |  |  | 诊所的名称 |

| E1\_4：诊所排班（Clinic\_Schedule） | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| 属性名称 | 标识 | 强制 | 多值 | 复合 | 派生 | 说明 |
| schedule\_id | ID1 | ● |  |  |  | 排班的唯一标识 |
| clinic\_id |  | ● |  |  |  | 诊所的标识 |
| date |  | ● |  |  |  | 日期 |
| time\_slot |  | ● |  |  |  | 时间段 |
| doctor\_id |  | ● |  |  |  | 医生的标识 |

## 需要用触发器实现的业务规则

需要用触发器实现的业务规则如下表。

|  |  |  |
| --- | --- | --- |
| **表** | **触发时间** | **业务规则（约束）及其设计要求** |
| Doctor\_Schedule | 新增 | 检查新增排班的医生在同一时间是否已有其他排班或请假安排，如果有，禁止（撤销）新增本条数据。  修改 修改排班时间时，检查修改后的时间是否与该医生的其他排班或请假时间冲突，若冲突，禁止修改。 |
| Schedule | 新增 | 检查新增排班的医生在该日期和时间段是否已有其他排班安排，若有，禁止（撤销）新增本条数据。 |
| Time\_Table | 新增 | 检查新增工作时间的医生在同一日期和时间段是否已有其他安排（包括排班和请假），若有，禁止（撤销）新增本条数据。 |
| Clinic\_Schedule | 新增 | 检查新增排班的诊所和医生在该日期和时间段是否已有其他相关安排，若有，禁止（撤销）新增本条数据。  修改 修改排班信息时，确保修改后的内容不会导致与诊所或医生的其他安排冲突，若冲突，禁止修改。 |

# 在MySQL平台中实现数据库

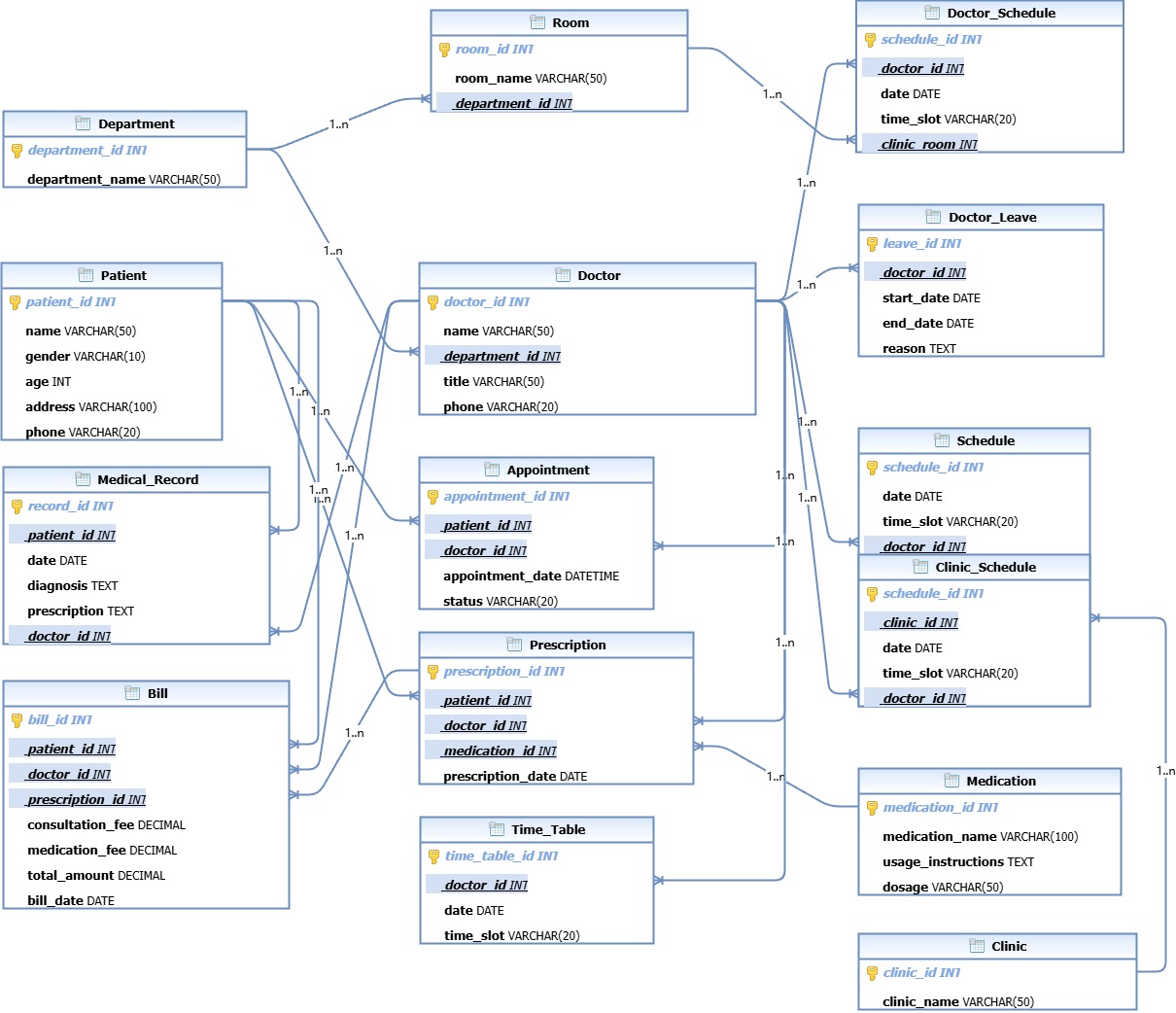
按照上一节的设计结果，在MySQL平台上实现数据库。实现数据库的过程如下：

1. 创建数据库。
2. 创建表：包括属性的数据类型、允许空否、默认值等。
3. 实体完整性约束：实现候选码（设置主码、唯一键）。
4. 引用完整性约束：实现引用关系及其**引用动作**。
5. 实现用户定义完整性约束：实现触发器【有能力的同学可试一下】。

## 数据库关系图

生成数据库关系图（可按业务模块生成相应表的数据库关系图）。

数据库关系图



## 表的结构

抓取每个表的设计图。生成创建表或更改表的MySQL代码。

### 科室表（Department）

CREATE TABLE IF NOT EXISTS Department (

department\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '科室ID',

department\_name VARCHAR(50) NOT NULL COMMENT '科室名称'

);

### 医生表（Doctor）

CREATE TABLE IF NOT EXISTS Doctor (

doctor\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '医生ID',

name VARCHAR(50) NOT NULL COMMENT '姓名',

department\_id INT NOT NULL COMMENT '科室ID',

title VARCHAR(50) NOT NULL COMMENT '职称',

phone VARCHAR(20) COMMENT '电话',

FOREIGN KEY (department\_id) REFERENCES Department(department\_id)

);

### 诊室表（Room）

CREATE TABLE IF NOT EXISTS Room (

room\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '诊室ID',

room\_name VARCHAR(50) NOT NULL COMMENT '诊室名称',

department\_id INT NOT NULL COMMENT '科室ID',

FOREIGN KEY (department\_id) REFERENCES Department(department\_id)

);

### 医生排班信息表（Doctor\_Schedule）

CREATE TABLE IF NOT EXISTS Doctor\_Schedule (

schedule\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '排班ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

date DATE NOT NULL COMMENT '日期',

time\_slot VARCHAR(20) NOT NULL COMMENT '时间段',

clinic\_room INT NOT NULL COMMENT '诊室ID',

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id),

FOREIGN KEY (clinic\_room) REFERENCES Room(room\_id)

);

### 医生请假信息表（Doctor\_Leave）

CREATE TABLE IF NOT EXISTS Doctor\_Leave (

leave\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '请假ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

start\_date DATE NOT NULL COMMENT '开始日期',

end\_date DATE NOT NULL COMMENT '结束日期',

reason TEXT COMMENT '请假理由',

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

### 患者表（Patient）

CREATE TABLE IF NOT EXISTS Patient (

patient\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '患者ID',

name VARCHAR(50) NOT NULL COMMENT '姓名',

gender VARCHAR(10) NOT NULL COMMENT '性别',

age INT NOT NULL COMMENT '年龄',

address VARCHAR(100) COMMENT '地址',

phone VARCHAR(20) COMMENT '电话'

);

### 病历记录表（Medical\_Record）

CREATE TABLE IF NOT EXISTS Medical\_Record (

record\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '病历ID',

patient\_id INT NOT NULL COMMENT '患者ID',

date DATE NOT NULL COMMENT '日期',

diagnosis TEXT COMMENT '诊断',

prescription TEXT COMMENT '处方',

doctor\_id INT NOT NULL COMMENT '医生ID',

FOREIGN KEY (patient\_id) REFERENCES Patient(patient\_id),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

### 预约信息表（Appointment）

CREATE TABLE IF NOT EXISTS Appointment (

appointment\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '预约ID',

patient\_id INT NOT NULL COMMENT '患者ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

appointment\_date DATETIME NOT NULL COMMENT '预约日期',

status VARCHAR(20) COMMENT '状态',

FOREIGN KEY (patient\_id) REFERENCES Patient(patient\_id),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

### 药品表（Medication）

CREATE TABLE IF NOT EXISTS Medication (

medication\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '药品ID',

medication\_name VARCHAR(100) NOT NULL COMMENT '药品名称',

usage\_instructions TEXT COMMENT '用法',

dosage VARCHAR(50) COMMENT '剂量'

);

### 处方表（Prescription）

CREATE TABLE IF NOT EXISTS Prescription (

prescription\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '处方ID',

patient\_id INT NOT NULL COMMENT '患者ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

medication\_id INT NOT NULL COMMENT '药品ID',

prescription\_date DATE NOT NULL COMMENT '开具日期',

FOREIGN KEY (patient\_id) REFERENCES Patient(patient\_id),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id),

FOREIGN KEY (medication\_id) REFERENCES Medication(medication\_id)

);

### 账单表（Bill）

CREATE TABLE IF NOT EXISTS Bill (

bill\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '账单ID',

patient\_id INT NOT NULL COMMENT '患者ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

prescription\_id INT NOT NULL COMMENT '处方ID',

consultation\_fee DECIMAL(10, 2) COMMENT '诊疗费',

medication\_fee DECIMAL(10, 2) COMMENT '药品费用',

total\_amount DECIMAL(10, 2) COMMENT '总费用',

bill\_date DATE NOT NULL COMMENT '账单日期',

FOREIGN KEY (patient\_id) REFERENCES Patient(patient\_id),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id),

FOREIGN KEY (prescription\_id) REFERENCES Prescription(prescription\_id)

);

### 排班信息表（Schedule）

CREATE TABLE IF NOT EXISTS Schedule (

schedule\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '排班ID',

date DATE NOT NULL COMMENT '日期',

time\_slot VARCHAR(20) NOT NULL COMMENT '时间段',

doctor\_id INT NOT NULL COMMENT '医生ID',

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

### 医生工作时间表（Time\_Table）

CREATE TABLE IF NOT EXISTS Time\_Table (

time\_table\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '时间表ID',

doctor\_id INT NOT NULL COMMENT '医生ID',

date DATE NOT NULL COMMENT '日期',

time\_slot VARCHAR(20) NOT NULL COMMENT '时间段',

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

### 诊所表（Clinic）

CREATE TABLE IF NOT EXISTS Clinic (

clinic\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '诊所ID',

clinic\_name VARCHAR(50) NOT NULL COMMENT '诊所名称'

);

### 诊所排班信息表（Clinic\_Schedule）

CREATE TABLE IF NOT EXISTS Clinic\_Schedule (

schedule\_id INT PRIMARY KEY AUTO\_INCREMENT COMMENT '排班ID',

clinic\_id INT NOT NULL COMMENT '诊所ID',

date DATE NOT NULL COMMENT '日期',

time\_slot VARCHAR(20) NOT NULL COMMENT '时间段',

doctor\_id INT NOT NULL COMMENT '医生ID',

FOREIGN KEY (clinic\_id) REFERENCES Clinic(clinic\_id),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id)

);

## 勾核数据库实现情况

## 表的实现情况

| 表 | 表、列、数据类型、默认值 | PK | CK | FK | 触发器 |
| --- | --- | --- | --- | --- | --- |
| 科室表（Department） | ✔ | ✔ | ✔ | ✔ |  |
| 医生表（Doctor） | ✔ | ✔ |  | ✔ |  |
| 诊室表（Room） | ✔ | ✔ |  | ✔ |  |
| 医生排班信息表（Doctor\_Schedule） | ✔ |  |  | ✔ |  |
| 医生请假信息表（Doctor\_Leave） | ✔ |  |  | ✔ |  |
| 患者表（Patient） | ✔ | ✔ | ✔ | ✔ |  |
| 病历记录表（Medical\_Record） | ✔ |  |  | ✔ |  |
| 预约信息表（Appointment） | ✔ |  |  | ✔ |  |
| 药品表（Medication） | ✔ | ✔ |  | ✔ |  |
| 处方表（Prescription） | ✔ |  |  | ✔ |  |
| 账单表（Bill） | ✔ |  |  | ✔ |  |
| 排班管理模块: | ✔ |  |  | ✔ |  |
| 医生工作时间表（Time\_Table） | ✔ |  |  | ✔ |  |
| 诊所管理模块: | ✔ |  |  | ✔ |  |
| 诊所排班信息表（Clinic\_Schedule） | ✔ |  |  | ✔ |  |

# 数据库应用

要求：每人至少完成1个业务模块中的增删改的SQL代码各一个，查询信息的SQL代码5个以上。查询要求自己设计。

## 患者管理模块的增删改查（付庭喜同学负责）

（1）向 Patient 患者表插入一个新患者信息。SQL 代码如下：

INSERT INTO Patient (patient\_id, name, gender, age, address, phone) VALUES (1001, '张 三', '男', 30, '云南昆明', '1234567890');

（2）修改患者姓名。SQL 代码如下：

UPDATE Patient SET name = '新姓名' WHERE patient\_id = 1;

（3）删除年龄小于 18 岁的患者数据。SQL 代码如下：

DELETE FROM Patient WHERE age < 18;

（4）查询所有患者信息。SQL 代码如下：

SELECT \* FROM Patient;

（5）查询男性患者信息。SQL 代码如下：

SELECT \* FROM Patient WHERE gender = '男';

（6）查询地址为“北京”的患者信息。SQL 代码如下：

SELECT \* FROM Patient WHERE address = '北京';

（7）统计患者总人数。SQL 代码如下：

SELECT COUNT(\*) FROM Patient;

（8）分类统计出不同性别患者的人数。SQL 代码如下：

SELECT gender, COUNT(\*) FROM Patient GROUP BY gender;

## 科室+医生管理模块业务的增删改查（张成双同学负责）

（1）向 Department 科室表插入一个新科室信息。SQL 代码如下：

INSERT INTO Department (department\_id, department\_name) VALUES (101, '内科');

（2）修改科室名称。SQL 代码如下：

UPDATE Department SET department\_name = '新科室名' WHERE department\_id = 1;

（3）删除指定科室数据。SQL 代码如下：

DELETE FROM Department WHERE department\_id = 10;

（4）查询所有科室信息。SQL 代码如下：

SELECT \* FROM Department;

1. 查询名称包含“外科”的科室信息。SQL 代码如下：

SELECT \* FROM Department WHERE department\_name LIKE '%外科%';

（6）统计科室总数量。SQL 代码如下：

SELECT COUNT(\*) FROM Department;

（7）向 Doctor 医生表插入一个新医生信息。SQL 代码如下：

INSERT INTO Doctor (doctor\_id, name, department\_id, title, phone) VALUES (2001, '李四', 1, '主任医师', '0987654321');

（8）修改医生职称。SQL 代码如下：

UPDATE Doctor SET title = '副主任医师' WHERE doctor\_id = 2;

## 诊室管理模块+排班业务的增删改查（蒋文栋同学负责）

（1）向 Room 诊室表插入一个新诊室信息。SQL 代码如下：

INSERT INTO Room (room\_id, room\_name, department\_id) VALUES (3001, '一号诊室', 2);

（2）修改诊室名称。SQL 代码如下：

UPDATE Room SET room\_name = '新诊室名' WHERE room\_id = 3;

（3）删除特定科室的诊室数据。SQL 代码如下：

DELETE FROM Room WHERE department\_id = 3;

（4）查询所有诊室信息。SQL 代码如下：

SELECT \* FROM Room;

1. 查询特定名称的诊室信息。SQL 代码如下：

SELECT \* FROM Room WHERE room\_name = '二号诊室';

（6）统计诊室总数量。SQL 代码如下：

SELECT COUNT(\*) FROM Room;

（7）向 Doctor\_Schedule 医生排班表插入一个新排班信息。SQL 代码如下：

INSERT INTO Doctor\_Schedule (schedule\_id, doctor\_id, date, time\_slot, clinic\_room) VALUES (4001, 101, '2024-07-15', '上午', 501);

（8）修改排班的时间。SQL 代码如下：

UPDATE Doctor\_Schedule SET time\_slot = '下午' WHERE schedule\_id = 1;

（9）删除特定医生的排班数据。SQL 代码如下：

DELETE FROM Doctor\_Schedule WHERE doctor\_id = 5;

# 设计工作总结

**付庭喜同学的设计工作总结如下：**

在过去的学习建模期间，我负责医疗相关的多项任务。维护患者表，录入和完善大量患者信息。管理病历记录，新增和修正众多病历。处理预约安排与调整，提高预约效率和患者满意度。协助药品管理，保障药品信息准确。通过这些工作，为这个医疗数据库项目的顺利开展贡献了自己的力量。

**张成双同学的设计工作总结如下：**

在本次工作中，我负责了科室管理模块和医生管理模块的相关工作。在科室管理模块中，我精心设计了科室表的结构，确保科室信息的准确存储和高效查询。同时，实现了科室的增删改查功能，满足了对科室信息的灵活管理需求。在医生管理模块里，构建了合理的医生表，关联了科室信息，完善了医生个人信息的存储和操作功能。通过对这两个模块的努力，为医疗系统的正常运转提供了有力的基础支撑，有效提升了管理效率和数据准确性。

**蒋文栋同学的设计工作总结如下：**

在本次工作中，我承担了排班管理模块和诊所管理模块的相关任务。对于排班管理模块，我精心规划了排班信息的存储结构，充分考虑了医生、时间、诊室等多方面因素，实现了精准、灵活且高效的排班功能。在诊所管理模块，我设计了完善的诊所信息架构，确保诊所的基本信息和相关安排能够准确记录和便捷查询，有力地保障了诊所运营的有序性和可管理性。

**项目组的设计工作总结如下：**

在过去的一段时间里，数据库项目组紧密围绕医疗信息管理的需求，开展了一系列富有成效的工作。

（一）数据库设计与构建

成功设计并创建了患者表（Patient）、病历记录表（Medical\_Record）、预约信息表 （Appointment）、药品表（Medication）、处方表（Prescription）和账单表（Bill） 等核心数据表，为医疗数据的存储和管理提供了坚实的基础。精心设置了外键约束和 索引，确保数据的一致性和查询的高效性。

（二）数据录入与维护

完成了大量初始数据的录入工作，确保患者、病历、药品等关键信息准确无误地进入 数据库。建立了定期的数据维护机制，及时更新患者信息、药品库存等动态数据，保 证数据的时效性。

（三）性能优化与安全保障

对数据库进行性能优化，调整查询语句和索引结构，显著提高了数据查询和处理的速 度。加强数据库安全管理，设置合理的用户权限，定期进行数据备份，有效防范了数 据泄露和丢失的风险。

（四）协作与沟通

与团队成员保持密切沟通，及时了解业务需求和系统变化，确保数据库的设计和使用 满足各方要求。积极参与项目组内部的技术讨论和分享，提升团队整体的技术水平和 协作能力。

二、存在问题与改进措施

（一）数据质量问题

部分数据录入存在不规范和错误，影响数据分析的准确性。今后将加强数据录入培训 和审核机制。

（二）性能瓶颈

在高并发情况下，数据库仍存在一定的性能瓶颈。将进一步优化数据库架构和服务器 配置。