数据库设计说明书

# 1引言

## 1.1编写目的

* 本数据库设计说明书是关于此博客系统数据库设计，主要包括数据概念结构的设计、逻辑结构设计、物理结构设计以及数据字典、运行环境、安全设计等。
* 读者范围：用户、系统设计人员、系统开发人员、系统测试人员、系统维护人员。
* 本数据库设计说明书根据系统需求分析设计所编写。
* 为团队开发提供参考基础

## 1.2背景

随着互联网发展，数字化智能化在我们的生活中越来越普遍，使我们的生活越来越便利。为提高医疗过程中治疗的效率和患者的体验实现该医院治疗流程管理系统。

## 1.3定义

* MySQL：一个关系型数据库管理系统，在 WEB 应用方面，MySQL是最好的RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。
* 医院治疗流程管理系统：系统允许医生和病人在治疗过程中进行有效的协作。系统支持两种类型的治疗环境：病床和诊疗室。病人根据预先制定的治疗流程在医院进行治疗，系统负责安排医生和诊疗室，并管理治疗流程的执行过程.

# 2外部设计

## 2.1标识符和状态

数据库名称：Assiginment

* Patients:病人表；标识符p\_id；用于记录病人信息
* Doctors：医生表；标识符d\_id；用于记录医生信息
* Items：项目表；标识符i\_id;用于记录项目信息
* Processes：流程表；标识符pr\_id;用于记录医疗流程信息
* Process\_Items:流程所拥有项目；标识符id；用于记录医疗流程所有的项目
* Doctor\_Item：医生项目关系表；标识符di\_id;用于记录医生可完成的项目

## 2.2使用它的程序

|  |  |  |
| --- | --- | --- |
| 应用程序 | 访问的数据表 | 版本号 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 2.4约定

|  |  |  |
| --- | --- | --- |
| 文卷 | 记录 | 数据项的命名约定 |
| 病人表 | Patients | 无 |
| 医生表 | Doctors | 无 |
| 项目表 | Items | 无 |
| 流程表 | Processes | 无 |
| 流程所拥有项目 | Process\_Items | 无 |
| 医生项目关系表 | Doctor\_Item | 无 |

## 2.4专门指导

* 字符集采用utf8
* 排序规则用utf8\_general\_ci
* 按设计编辑字段名、类型、长度、不是null、键等
* 注意数据库名和表名的大小写

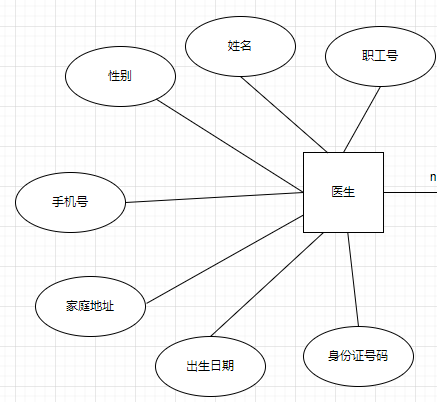
## 2.5支持软件

|  |  |  |
| --- | --- | --- |
| 支持软件名称 | 版本号 | 主要功能 |
| Mysql | 8.0 | 建立数据库并提供数据库维护与管理功能 |
| jdk | 17 | 代码编写 |
| DataGrip |  | 数据库管理工具 |

# 3结构设计

## 3.1概念结构设计

医生（职工号，姓名，性别，手机号，家庭地址，出生日期，身份证号码）

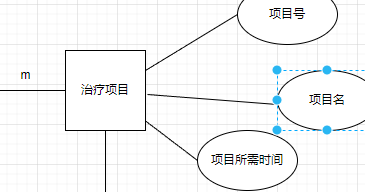


病人（病人号，姓名，性别，身份证号，出生日期，手机号，家庭地址，紧急联系人，紧急联系人手机号）

图示

描述已自动生成

项目（项目号，项目名，项目所需的时间）



治疗流程（流程号，病人号，治疗日期）

图示

描述已自动生成

医生项目对应（id，项目号，医生职工号）

图示, 示意图

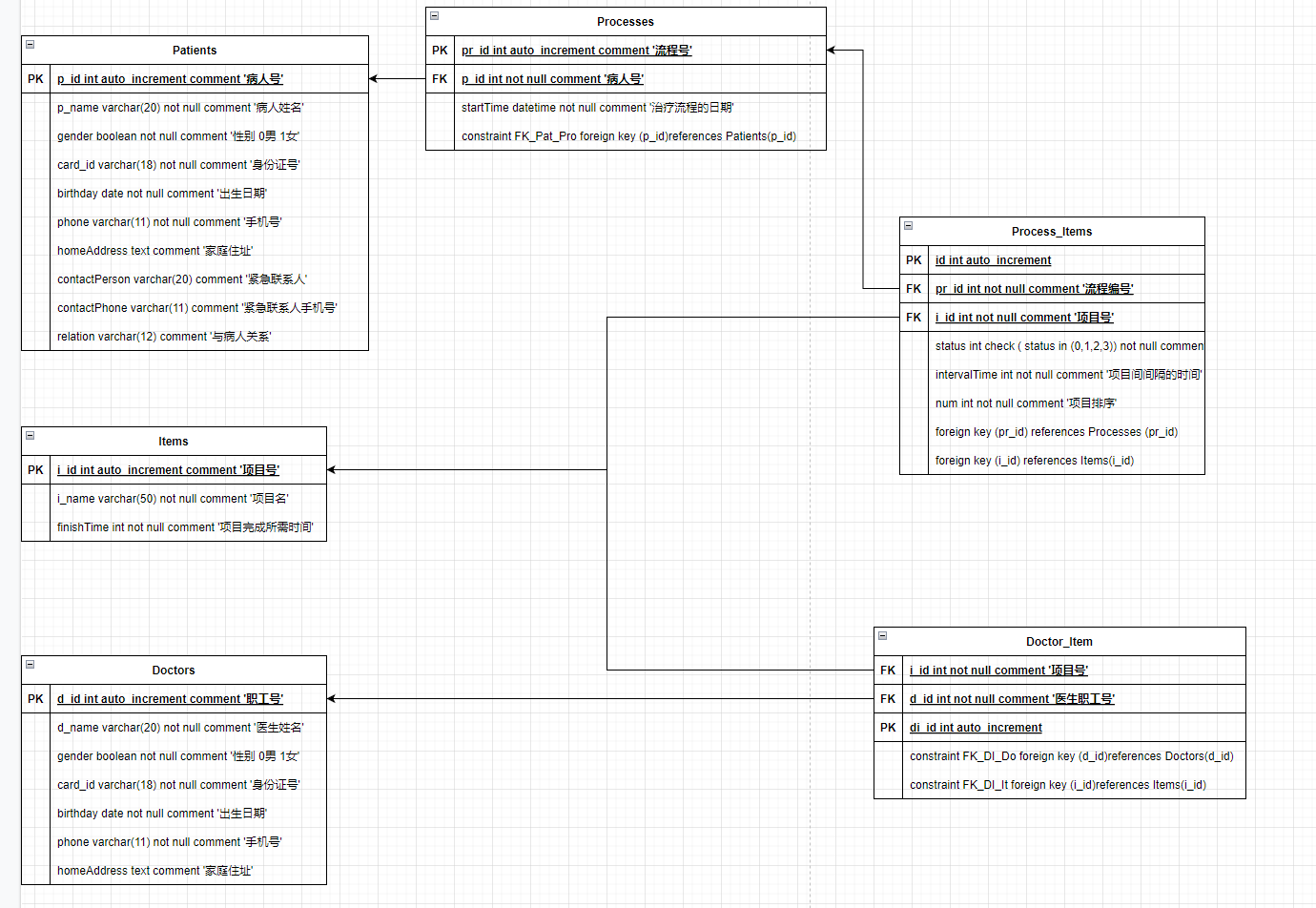
描述已自动生成

流程项目（id，流程号，项目号，项目间隔，项目排序，项目状态）

图示

描述已自动生成

**全局E-R图**

****

## 3.2逻辑结构设计

**病人表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| p\_id | int |  | 是 | 是 |
| p\_name | varchar | 20 |  |  |
| gender | Boolean |  |  |  |
| card\_id | date | 18 |  |  |
| birthday | varchar |  |  |  |
| phone | text | 11 |  |  |
| homeAddress | varchar |  |  |  |
| contactPerson | varchar | 20 |  |  |
| contactPhone | varchar | 11 |  |  |
| relation | varchar | 12 |  |  |

**医生表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| d\_id | int |  | 是 | 是 |
| d\_name | varchar | 20 |  |  |
| gender | Boolean |  |  |  |
| card\_id | date | 18 |  |  |
| birthday | varchar |  |  |  |
| phone | text | 11 |  |  |
| homeAddress | varchar |  |  |  |

**项目表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| i\_id | int |  | 是 | 是 |
| i\_name | varchar | 50 |  |  |
| finishTime | int |  |  | 是 |

**流程表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| pr\_id | int |  | 是 | 是 |
| startTime | datetime |  |  | 是 |
| p\_id | int |  |  | 是 |

**流程项目表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| id | int |  | 是 | 是 |
| pr\_id | int |  |  | 是 |
| i\_id | int |  |  | 是 |
| status | int |  |  | 是 |
| intervalTime | int |  |  | 是 |
| num | int |  |  | 是 |

**医生项目表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 主键 | 非空 |
| di\_id | int |  | 是 | 是 |
| d\_id | int |  |  | 是 |
| i\_id | int |  |  | 是 |

## 3.3物理结构设计

**创建Patients表**

create table Patients  
(  
 p\_id int primary key auto\_increment comment '病人号',  
 p\_name varchar(20) not null comment '病人姓名',  
 gender boolean not null comment '性别 0男 1女',  
 card\_id varchar(18) not null comment '身份证号',  
 birthday date not null comment '出生日期',  
 phone varchar(11) not null comment '手机号',  
 homeAddress text comment '家庭住址',  
 contactPerson varchar(20) comment '紧急联系人',  
 contactPhone varchar(11) comment '紧急联系人手机号',  
 relation varchar(12) comment '与病人关系'  
)comment '病人';

**创建Doctors表**  
create table Doctors  
(  
 d\_id int primary key auto\_increment comment '职工号',  
 d\_name varchar(20) not null comment '医生姓名',  
 gender boolean not null comment '性别 0男 1女',  
 card\_id varchar(18) not null comment '身份证号',  
 birthday date not null comment '出生日期',  
 phone varchar(11) not null comment '手机号',  
 homeAddress text comment '家庭住址'  
)comment '医生';

**创建Items表**  
create table Items  
(  
 i\_id int primary key auto\_increment comment '项目号',  
 i\_name varchar(50) not null comment '项目名',  
 finishTime int not null comment '项目完成所需时间'  
)comment '项目';

**创建Processes表**  
create table Processes  
(  
 pr\_id int primary key auto\_increment comment '流程号',  
 startTime datetime not null comment '治疗流程的日期',  
 p\_id int not null comment '病人号',  
 constraint FK\_Pat\_Pro foreign key (p\_id)references Patients(p\_id)  
)comment '流程';

**创建Process\_Items表**  
create table Process\_Items  
(  
 id int primary key auto\_increment,  
 pr\_id int not null comment '流程编号',  
 i\_id int not null comment '项目号',  
 status int check ( status in (0,1,2,3)) not null comment '当前状态 0异常 1未启动 2进行中 3 结束',  
 intervalTime int not null comment '项目间间隔的时间',  
 num int not null comment '项目排序',  
 foreign key (pr\_id) references Processes (pr\_id),  
 foreign key (i\_id) references Items(i\_id)  
)comment '流程所拥有项目';

**创建Doctor\_Item表**  
create table Doctor\_Item  
(  
 di\_id int primary key auto\_increment,  
 d\_id int not null comment '医生职工号',  
 i\_id int not null comment '项目号',  
 constraint FK\_DI\_Do foreign key (d\_id)references Doctors(d\_id),  
 constraint FK\_DI\_It foreign key (i\_id)references Items(i\_id)  
)comment '医生项目关系表';

# 4运用设计

## 4.1安全保密设计

* 在云服务中对安全组进行配置，防止不明地址的对数据库的请求和攻击。
* 严格分配数据库用户权限。
* 对不同功能模块授予不同的角色管理。
* 将数据库管理用户分为管理员组、开发组、测试组。