复杂权限系统；---使用shiro；

实际需要了再为之；先只是分析；

用户权限模型2： ---10

资源； ---5

角色；

用户；

角色资源表；

用户角色表；

~

区域；

部门；

岗位；

部门岗位表；

岗位角色表；

区域：

用户选择区域；

需要的内容表中添加区域字段；

区域管理员，添加的用户就是本区域的；| 添加用户时的区域下拉框只有本区域；

用户登录后，绑定本区域内容；

部门： ---oa等管理系统中需要；

管理部门，管理岗位；

为岗位选择角色；

部门主管维护本部门用户；---添加，修改；

部门主管创建用户，选择岗位；用户自动拥有岗位角色；

用户可以多部门，多岗位；---即是：用户可以多角色；

如：a既是技术4主管，也是公司副总；

分析：

在医院系统实现中：

部门大类分两类：本医院用户，外部用户；

二级类中，医院用户的岗位；外部用户的类别；

--------------------------------

资源：

模块---一级；

Action---二级；| 对应模块；

~

页面，等价于action，在action记录中关联；

·当页面url时，找到对应的action，判断权限；

//////////////////////////////////////////////////////////////////////////////

表：

资源表：---无限级；自外键；

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| resource\_id | bigint | 自增主键 | 自增、无符号、not null | 主键 not null |
| sup\_resource\_id | bigint | 父类id | s=superior | 自外键 |
| res\_name | varchar(50) | 资源名 |  | not null |
| res\_action | varchar(100) | 动作 |  |  |
| res\_url | varchar(100) | url |  |  |
| res\_code | varchar(20) | 代码 |  |  |
| res\_sequence | int | 排序 | 默认0 | not null |
| res\_remarks | varchar(200) | 备注 |  |  |

~

模块，对应的action、url为空；

角色表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| role\_id | bigint |  |  | 主键 not null |
| role\_name | varchar(50) | 角色名 |  | not null |
| role\_status | Tinyint | 状态 | 0关闭1正常 | not null |
| role\_sequence | int | 排序 |  | not null |
| role\_remarks | varchar(200) | 备注 |  |  |

用户表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user\_id | bigint |  |  | 主键 not null |
| area\_id | bigint | 区域 | 暂时清除外键，给定默认值0； | 外键 not null |
| dept\_id | bigint | 部门id | 暂时清除外键，给定默认值0； | 外键 not null |
| Jobs\_id | bigint | 岗位id | 暂时清除外键，给定默认值0； | 外键 not null |
| user\_name | varchar(50) | 用户名 |  | not null |
| user\_pwd | varchar(50) | 密码 |  | not null |
| user\_addTime | datetime | 建立时间 |  | not null |
| user\_updateTime | datetime | 修改时间 |  | not null |
| user\_status | Tinyint | 状态 | 0锁定1正常2注销 | not null |
|  |  | 真实姓名 |  |  |
|  |  | 身份证号 |  |  |
|  |  | 是否实名认证 |  |  |
|  |  | 邮箱 |  |  |
|  |  | 是否邮箱认证 |  |  |
|  |  | 手机号 |  |  |
|  |  | 是否手机认证 |  |  |
| user\_remarks | varchar(200) | 备注 |  |  |

角色-资源表：t\_sys\_role\_resource

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| role\_resource\_id | bigint |  |  | 主键 not null |
| role\_id | bigint | 角色id |  | 外键 not null |
| resource\_id | bigint | 资源id |  | 外键 not null |

用户-角色表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| user\_role\_id | bigint |  |  | 主键 not null |
| user\_id | bigint | 用户id |  | 外键 not null |
| role\_id | bigint | 角色id |  | 外键 not null |

----------------------------

区域表：---无限级；自外键；

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| area\_id | bigint |  |  | 主键 not null |
| sup\_area\_id | bigint | 父类id |  | 自外键 |
| area\_name | varchar(50) | 区域名 |  | not null |
| area\_code | varchar(20) | 区域代码 |  |  |
| area\_sequence | int | 排序 | 默认0 | not null |
| area\_remarks | varchar(200) | 备注 |  |  |

部门表：---无限级；自外键；

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| dept\_id | bigint |  |  | 主键 not null |
| dept\_name | varchar(50) | 部门名 |  | not null |
| sup\_dept\_id | bigint | 父类id |  | 自外键 |
| dept\_sequence | int | 排序 | 默认0 | not null |
| dept\_remarks | varchar(500) | 备注 |  |  |

岗位表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| jobs\_id | bigint |  |  | 主键 not null |
| jobs\_name | varchar(50) | 岗位名 |  | not null |
| jobs\_sequence | int | 排序 | 默认0 | not null |
| jobs\_remarks | varchar(500) | 备注 |  |  |

部门-岗位表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| dept\_jobs\_id | bigint |  |  | 主键 not null |
| dept\_id | bigint | 部门id |  | 外键 not null |
| jobs\_id | bigint | 岗位id |  | 外键not null |

岗位-角色表：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| jobs\_role\_id | bigint |  |  | 主键 not null |
| jobs\_id | bigint | 部门id |  | 外键 not null |
| role\_id | bigint | 角色id |  | 外键not null |

///////////////////////////////////////////////////////////////////////////////

实现：