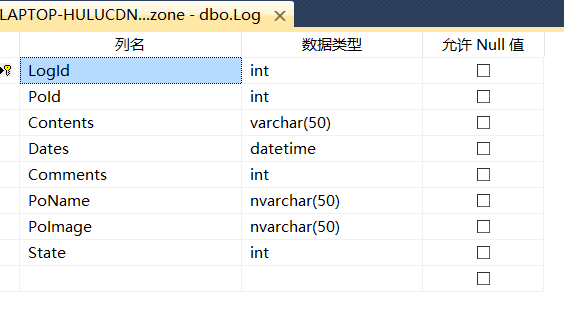
技术文档

——qq空间

1. 数据库设计
2. 记录说说内容的表

表名：Log

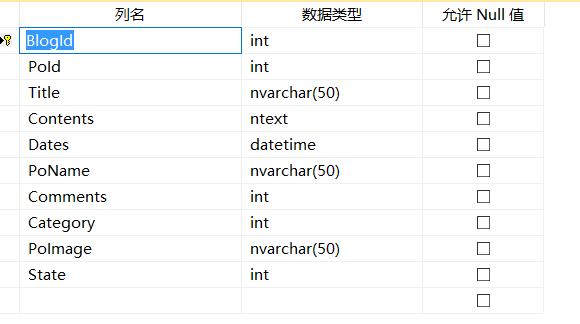


列名：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| LogId | PoId | Contents | Dates | Comments | PoName | PoImage | State |
| 主键  每条说说的id名 | 发表说说的用户的id | 发表的内容 | 发表的时间 | 该说说评论 | 发表人名字 | 发表人的头像 | 该说说的状态（若为1则是所有人可见，若为0则仅自己可见） |

1. 记录日志内容表

表名（Blog）



列名：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BlogId | PoId | Title | Contents | Dates | PoName | Comments | Category | PoImage | State |
| 主键  每条日志的id 唯一标识 | 发表人的id | 日志标题 | 日志内容 | 发表日期 | 发表人名字 | 该日志的评论数量 | 日志的分类 | 发表人的头像 | 该日志的状态（若为1则是所有人可见，若为0则仅自己可见） |

1. 记录留言内容的表

表名（message）



列名：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MessageId | PoId | Contents | Dates | PoName |
| 主键 每条留言的id 唯一标识 | 发表者的id | 留言内容 | 发表日期 | 发表者名字 |

|  |  |  |  |
| --- | --- | --- | --- |
| Comments | ParentId | ParentName | PoImange |
| 留言评论数量 | 被留言的用户id（即在谁的空间留言） | 被留言的用户姓名（即在谁的空间留言） | 发表者的头像 |

1. 记录相册的表

表名（Album）



列名

|  |  |  |  |
| --- | --- | --- | --- |
| AlbumId | AlbumName | OwnId | Describe |
| 主键 相册id 唯一标识 | 相册姓名 | 相册所有者的id | 该相册的简介 |

|  |  |  |  |
| --- | --- | --- | --- |
| AlbumImage | State | Dates | Category; |
| 存放相册封面的图片的路径 | 该相册的状态（若为1则是所有人可见，若为0则仅自己可见） | 创建相册的时间 | 该相册的分类 |

1. 记录照片的表

表名（Photo）



列名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PhotoId | PoId | Contents | Dates | Message |
| 该照片的id 主键 唯一标识 | 发表者id | 存放该照片的路径 | 发表时间 | 该照片的简介 |

|  |  |  |  |
| --- | --- | --- | --- |
| Comments | PoName | AlbumId | PoImage |
| 评论数量 | 发表者的名字 | 该照片所属相册的id | 发表者的头像 |

1. 记录说说评论的表

表名（ComLog）



列名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CommentId | ToId | ParentId | UserId | Contents |
| 该条评论的id 主键 唯一标识 | 该条评论对应的说说的id | 被回复人的id（如果是直接在说说下面评论，则是发表说说的人的id 若是回复某条评论，则为这条评论发表人的id） | 发布这条评论的用户的id | 评论的内容 |

|  |  |  |  |
| --- | --- | --- | --- |
| Dates | UserName | ParentName | Information |
| 发表时间 | 发布这条评论的用户的姓名 | 被回复人的姓名（如果是直接在说说下面评论，则是发表说说的人的姓名 若是回复某条评论，则为这条评论发表人的姓名） | 如果该条评论不是自己评论自己，那么information设定为1，直至这条消息被 被评论的人看见，information则变为-0。用来标识信息是否已读 |

1. 记录日志评论的表

表名（ComBlog）



列名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CommentId | ToId | ParentId | UserId | Contents |
| 该条评论的id 主键 唯一标识 | 该条评论对应的日志的id | 被回复人的id（如果是直接在日志下面评论，则是发表日志的人的id 若是回复某条评论，则为这条评论发表人的id） | 发布这条评论的用户的id | 评论的内容 |

|  |  |  |  |
| --- | --- | --- | --- |
| Dates | UserName | ParentName | Information |
| 发表时间 | 发布这条评论的用户的姓名 | 被回复人的姓名（如果是直接在日志下面评论，则是发表日志的人的姓名 若是回复某条评论，则为这条评论发表人的姓名） | 如果该条评论不是自己评论自己，那么information设定为1，直至这条消息被 被评论的人看见，information则变为-0。用来标识信息是否已读 |

1. 记录照片评论的表

表名（ComPhoto）



列名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CommentId | ToId | ParentId | UserId | Contents |
| 该条评论的id 主键 唯一标识 | 该条评论对应的照片的id | 被回复人的id（如果是直接在照片下面评论，则是发表照片的人的id 若是回复某条评论，则为这条评论发表人的id） | 发布这条评论的用户的id | 评论的内容 |

|  |  |  |  |
| --- | --- | --- | --- |
| Dates | UserName | ParentName | Information |
| 发表时间 | 发布这条评论的用户的姓名 | 被回复人的姓名（如果是直接在照片下面评论，则是发表照片的人的姓名 若是回复某条评论，则为这条评论发表人的姓名） | 如果该条评论不是自己评论自己，那么information设定为1，直至这条消息被 被评论的人看见，information则变为-0。用来标识信息是否已读 |

1. 记录留言评论的表

表名（ComMessage）



列名

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CommentId | ToId | ParentId | UserId | Contents |
| 该条评论的id 主键 唯一标识 | 该条评论对应的留言的id | 被回复人的id（如果是直接在留言下面评论，则是发表留言的人的id 若是回复某条评论，则为这条评论发表人的id） | 发布这条评论的用户的id | 评论的内容 |

|  |  |  |
| --- | --- | --- |
| Dates | UserName | ParentName |
| 发表时间 | 发布这条评论的用户的姓名 | 被回复人的姓名（如果是直接在留言下面评论，则是发表留言的人的姓名 若是回复某条评论，则为这条评论发表人的姓名） |

1. 记录用户信息的表

表名（Users）



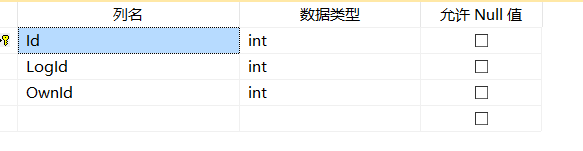
列名

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Id | Name | Password | UsersName | Sex | Email |
| 主键 用户id | 用户姓名 | 用户密码 | 用户真实姓名 | 性别 | 邮箱 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Telephone | Birthday | Blood | QzoneName | State | Image |
| 电话号码 | 生日 | 血型 | Qq空间名 | 状态（刚注册的时候为0，当通过邮箱验证的时候边为1，每次进入登录页会删除数据库中state为0的用户 ） | 用户的头像 |

1. 记录说说的赞的表

表名（GoodLog）

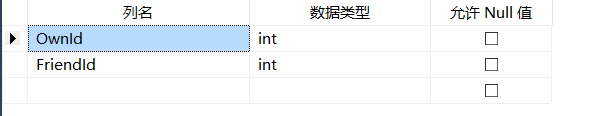


列名

|  |  |  |
| --- | --- | --- |
| Id | LogId | OwnId |
| 主键 每条赞的唯一标识 | 赞对应的说说的id | 点赞人的id |

1. 记录好友关系的表

表名（Friends）



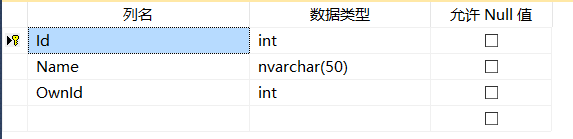
列名

|  |  |
| --- | --- |
| OwnId | FriendId |
| 用户id | 该用户关注好友的id |

（两个的关系为主键）

1. 记录相册分类的表

表名（AlbumCategory）

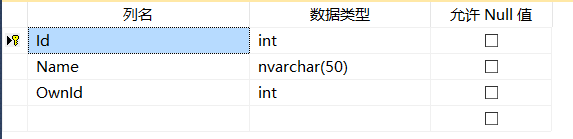


列名

|  |  |  |
| --- | --- | --- |
| Id | Name | OwnId |
| 主键 分类的唯一标识 | 分类名 | 该分类所属的用户的id |

1. 记录日志分类的表

表名（BLogCategory）

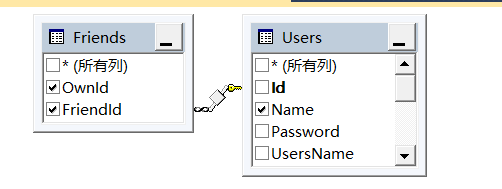


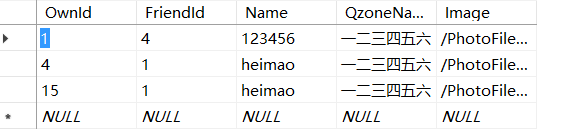
列名

|  |  |  |
| --- | --- | --- |
| Id | Name | OwnId |
| 主键 分类的唯一标识 | 分类名 | 该分类所属的用户的id |

1. 视图：有关好友的视图

视图名（View\_1）

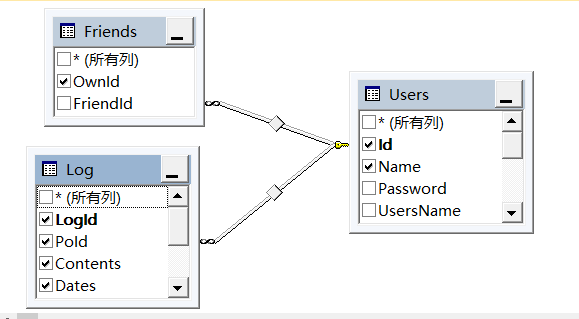




|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OwnId | FriendId | Name | QzoneName | Image |
| ；用户id | 用户关注好友的id | 用户好友的姓名 | 用户好友的空间名 | 用户好友的头像 |

1. 视图：有关好友说说动态的视图

视图名：LogF



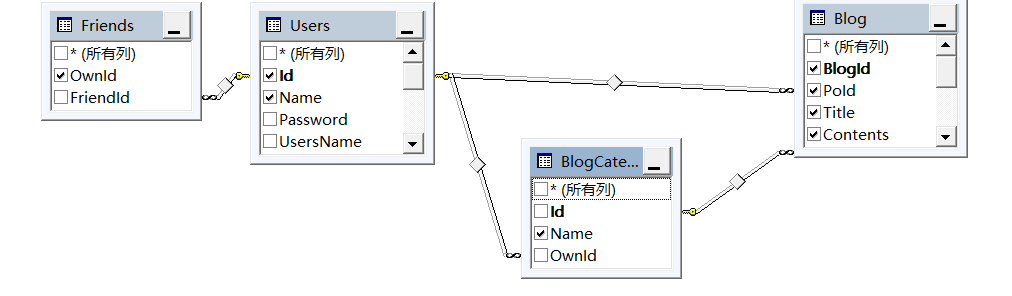


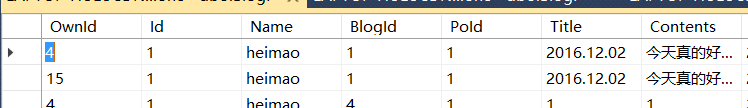
|  |  |  |
| --- | --- | --- |
| OwnId | Id | Name |
| 用户id | 用户好友的id | 用户好友名字 |

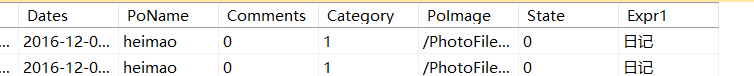
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| LogId | PoId | Contents | Dates | Comments | PoName | PoImage | State |
| 好友说说的id名 | 发表说说的用户的id即用户好友的id | 发表的内容 | 发表的时间 | 该说说评论 | （用户好友）发表人名字 | （用户好友）发表人的头像 | 该说说的状态（若为1则是所有人可见，若为0则仅自己可见） |

1. 视图：有关好友日志动态的视图

视图名（BLogF）







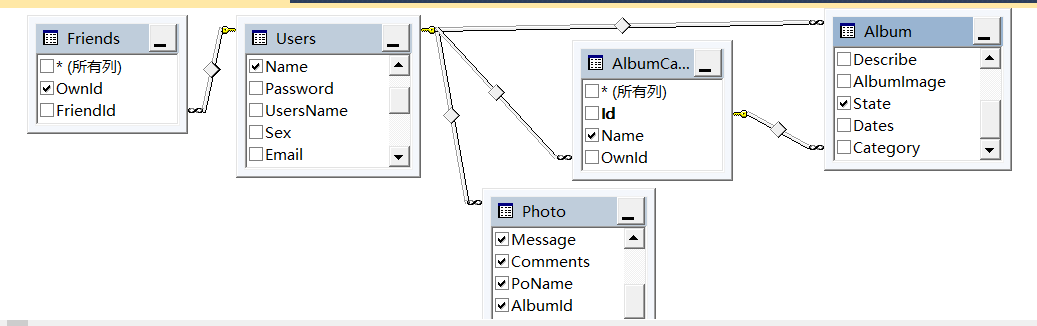
列名：

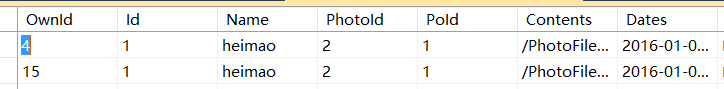
|  |  |  |
| --- | --- | --- |
| OwnId | Id | Name |
| 用户id | 用户好友的id | 用户好友名字 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BlogId | PoId | Title | Contents | Dates | PoName | Comments | Category | PoImage | State | Expr1 |
| 用户好友发表的日志的id | （用户好友）发表人的id | 日志标题 | 日志内容 | 发表日期 | （用户好友）发表人名字 | 该日志的评论数量 | 日志的分类 | （用户好友）发表人的头像 | 该日志的状态（若为1则是所有人可见，若为0则仅自己可见） | 日志的分类名 |

1. 视图 有关好友照片动态的视图

视图名（PhotoF）







列名

|  |  |  |
| --- | --- | --- |
| OwnId | Id | Name |
| 用户id | 用户好友的id | 用户好友名字 |

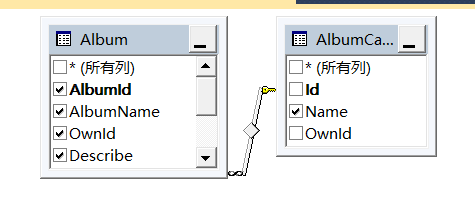
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PhotoId | PoId | Contents | Dates | Message |
| 用户好友发表的照片的id | （用户好友）发表者id | 存放该照片的路径 | 发表时间 | 该照片的简介 |

|  |  |  |  |
| --- | --- | --- | --- |
| Comments | PoName | AlbumId | AlbumName |
| 评论数量 | （用户好友）发表者的名字 | 该照片所属相册的id | 所属相册名 |

|  |  |  |
| --- | --- | --- |
| State | Category | PoImage |
| 该相册状态（若为1则是所有人可见，若为0则仅自己可见） | 所属相册分类 | （用户好友）发表者的头像 |

1. 视图 有关相册分类的视图

视图名（AlbumC）





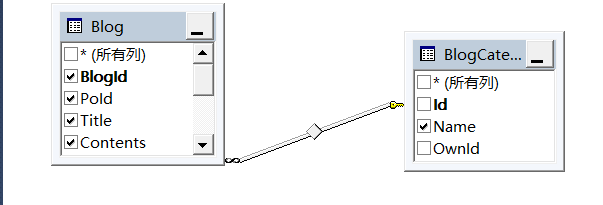
列名：

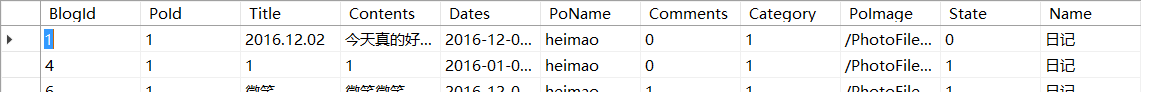
|  |  |  |  |
| --- | --- | --- | --- |
| AlbumId | AlbumName | OwnId | Describe |
| 主键 相册id 唯一标识 | 相册姓名 | 相册所有者的id | 该相册的简介 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AlbumImage | State | Dates | Category; | Name |
| 存放相册封面的图片的路径 | 该相册的状态（若为1则是所有人可见，若为0则仅自己可见） | 创建相册的时间 | 该相册的分类id | 该相册分类名 |

1. 视图 有关日志分类的视图

视图名（PhotoC）

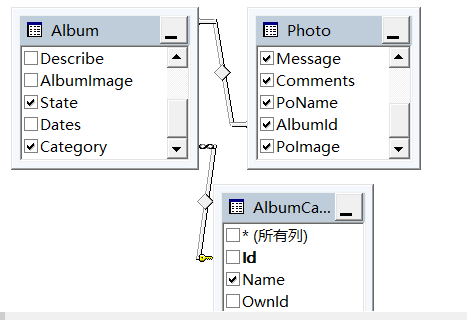


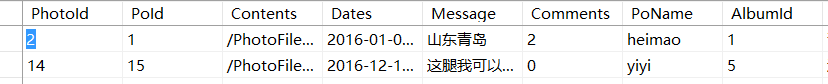


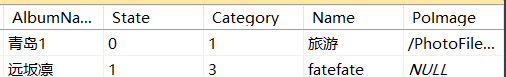
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BlogId | PoId | Title | Contents | Dates | PoName | Comments | Category | PoImage | State | Name |
| 主键  每条日志的id 唯一标识 | 发表人的id | 日志标题 | 日志内容 | 发表日期 | 发表人名字 | 该日志的评论数量 | 日志的分类id | 发表人的头像 | 该日志的状态（若为1则是所有人可见，若为0则仅自己可见） | 该日志分类名 |

1. 视图 有关照片和相册的视图

视图名（PhotoAlbum）





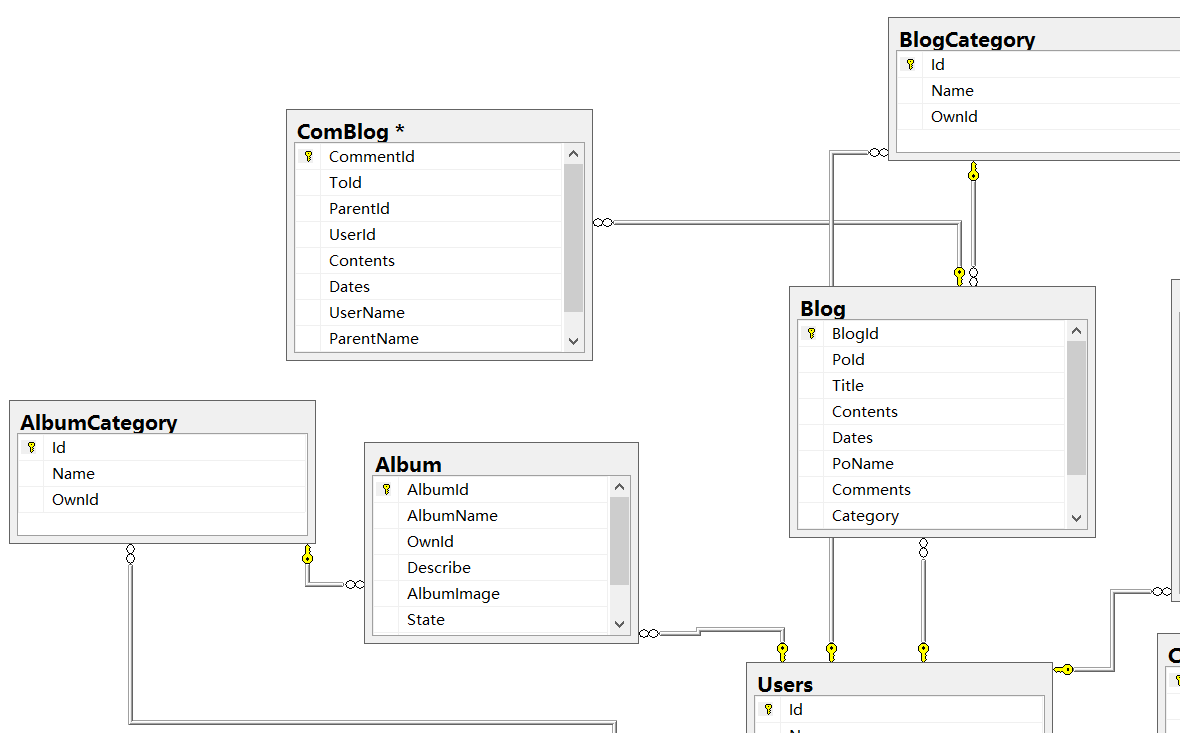


|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PhotoId | PoId | Contents | Dates | Message |
| 该照片的id 主键 唯一标识 | 发表者id | 存放该照片的路径 | 发表时间 | 该照片的简介 |

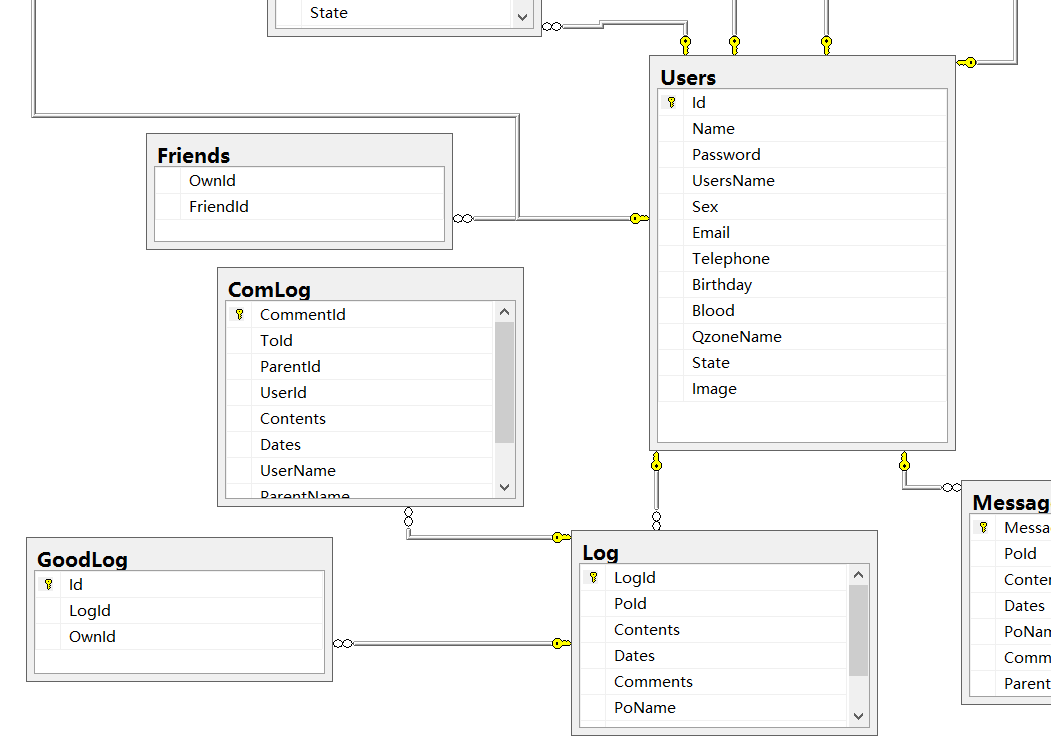
|  |  |  |  |
| --- | --- | --- | --- |
| Comments | PoName | AlbumId | AlbumName |
| 评论数量 | 发表者的名字 | 该照片所属相册的id | 所属相册名 |

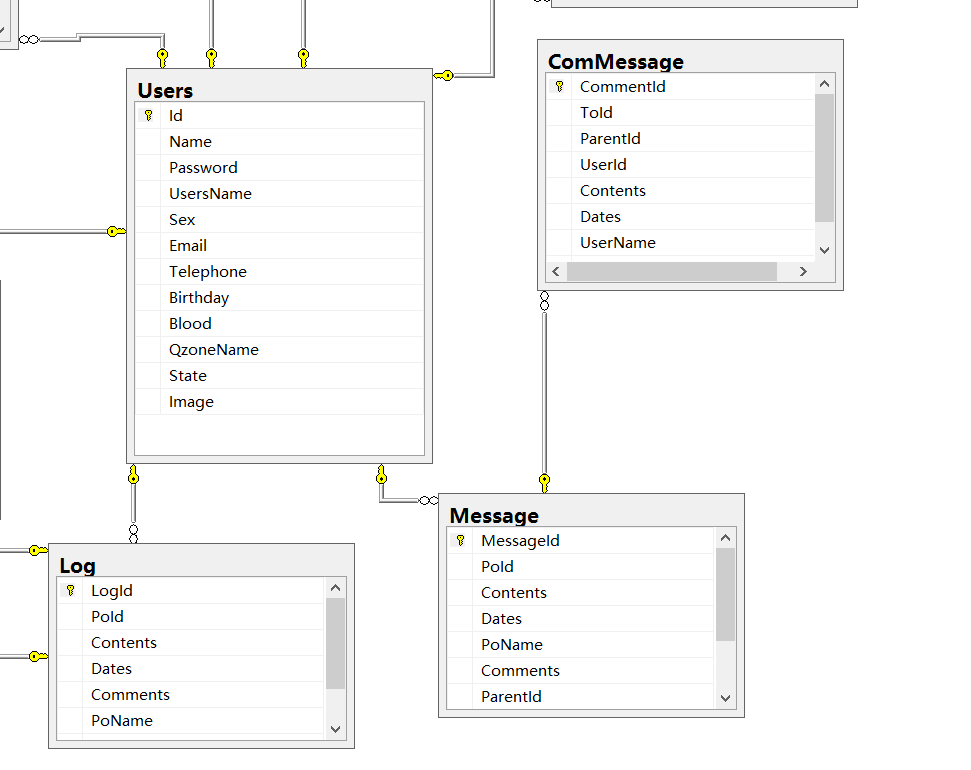
|  |  |  |
| --- | --- | --- |
| State | Category | PoImage |
| 该相册状态（若为1则是所有人可见，若为0则仅自己可见） | 所属相册分类 | 发表者的头像 |

1. 数据库关系









图可能比较乱 我会一个个列出关系来

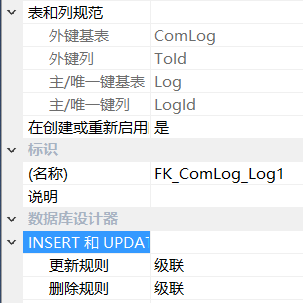
1. 主键：用户的id ；外键：说说表中发布说说者的id

当用户被删除时，该用户所有说说都会被删除



1. 主键：说说的id ；外键：说说的评论表中所对应说说的id

当该说说被删除时，该说说所有评论都会被删除



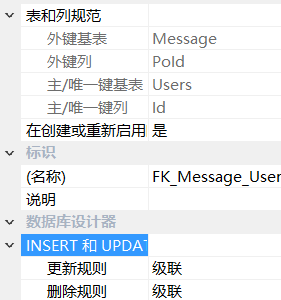
1. 主键：说说的id ；外键：说说的点赞表中所对应说说的id

当该说说被删除时，该说说所有赞都会被删除



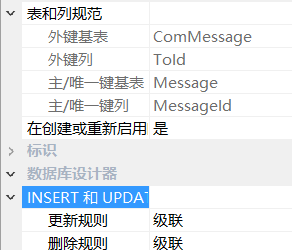
1. 主键：用户的id ；外键：留言表中发布留言者的id

当用户被删除时，该用户所有留言都会被删除



1. 主键：留言的id ；外键：留言的评论表中所对应留言的id

当该留言被删除时，该留言所有评论都会被删除



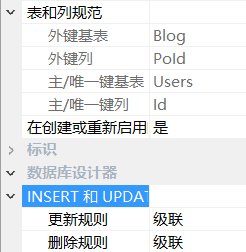
1. 主键：用户id 外键：用户关注好友的id

当该用户被删除时，该用户的被关注的关系会被删除



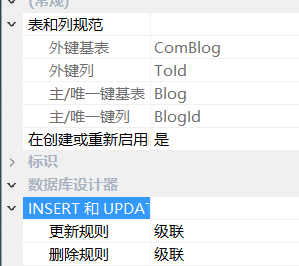
7、主键：用户的id ；外键：日志表中发布日志者的id

当用户被删除时，该用户所有日志都会被删除



1. 主键：日志的id ；外键：日志的评论表中所对应日志的id

当该日志被删除时，该日志所有评论都会被删除



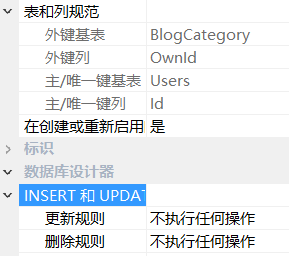
1. 主键：日志分类表中分类的id 外键：日志表中的日志分类id

当日志分类表中的某个分类被删除了 那么所属该分类的日志都将被删除



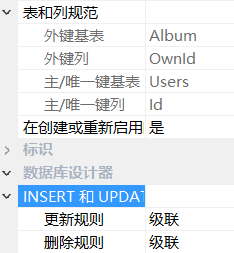
1. 主键：用户表中的用户id； 外键：日志分类表中该分类所属用户的id

当某个用户被删除了，该用户的所有分类被删除；（由于在sql server中这个关系的级联关系不能够被建立，于是我只是建立了关系并没有设置级联）



1. 主键：用户的id ；外键：相册表中发布相册者的id

当用户被删除时，该用户所有相册都会被删除



1. 主键：相册分类表中分类的id 外键：相册表中的相册分类id

当相册分类表中的某个分类被删除了 那么所属该分类的相册都将被删除



1. 主键：用户表中的用户id； 外键：相册分类表中该分类所属用户的id

当某个用户被删除了，该用户的所有分类被删除；（由于在sql server中这个关系的级联关系不能够被建立，于是我只是建立了关系并没有设置级联）



1. 主键：相册id 外键：照片表中该照片所属相册的id

当该相册被删除时，该相册下的所有照片都将被删除（由于在sql server中这个关系的级联关系不能够被建立，于是我只是建立了关系并没有设置级联，于是我选择了在代码里进行操作）



1. 主键：用户的id ；外键：照片表中发布照片者的id

当用户被删除时，该用户所有照片都会被删除



1. 主键：照片的id ；外键：照片的评论表中所对应照片的id

当该照片被删除时，该照片所有评论都会被删除



1. 所使用的一些方法类接口函数等等
2. 有关上传图片的函数