

（深圳）

实验报告

开课学期： 2021春季

课程名称： 数据库系统

实验名称： 微博 系统设计与实现

实验性质： 设计型

实验学时： 2 地点：

学生班级：

学生学号：

学生姓名：

评阅教师：

报告成绩：

实验与创新实践教育中心制

2021年1月

# 实验环境

*请填写用到的操作系统和主要软件版本。*

操作系统：Window系统

实验环境：Pycharm

主要软件版本：Flask==0.10.1

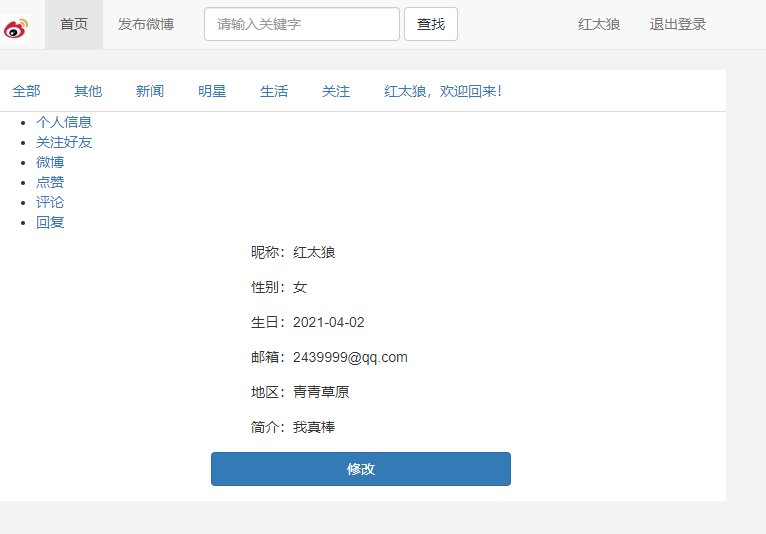
# 实验过程

## 系统功能

*请结合文字、表格、截图等方式，清晰描述系统的功能。亮点功能请用\*标志。*

系统实现微博的基本信息管理，包括用户管理，关注管理，文章管理（发布文章，浏览文章，搜索文章，点赞，评论，\*文章类别，\*关注用户的文章，\*回复评论）

除注册登录外，其余所有操作必须先登录

1. 用户管理：
   1. 用户注册：用户名，密码和确认密码
   2. 用户登录：用户名，密码登录
   3. 用户信息：
      1. 用户基本信息：
      2. 用户关注的好友及其分组：
      3. 用户的微博、点赞的微博、评论的微博、回复的微博：



1. 关注管理：
   1. 创建关注分组：默认分组自动创建
   2. 关注用户：



1. 文章管理：
   1. 浏览文章：
   2. 发布文章：
   3. 搜索文章：
   4. 点赞：
   5. 评论：
   6. 文章类别及关注用户微博：

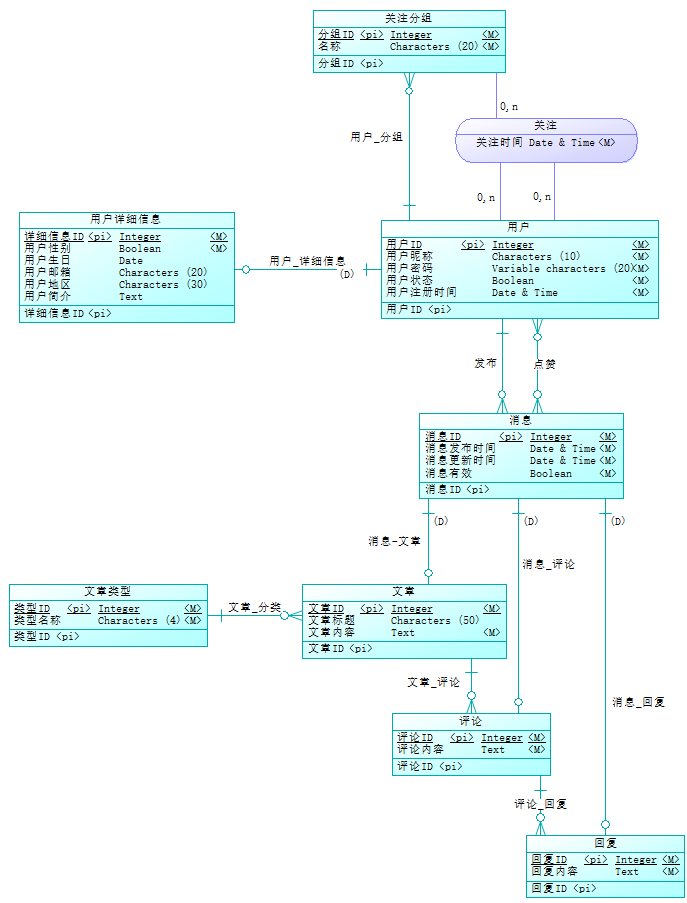


* 1. 微博明细：

## 数据库设计

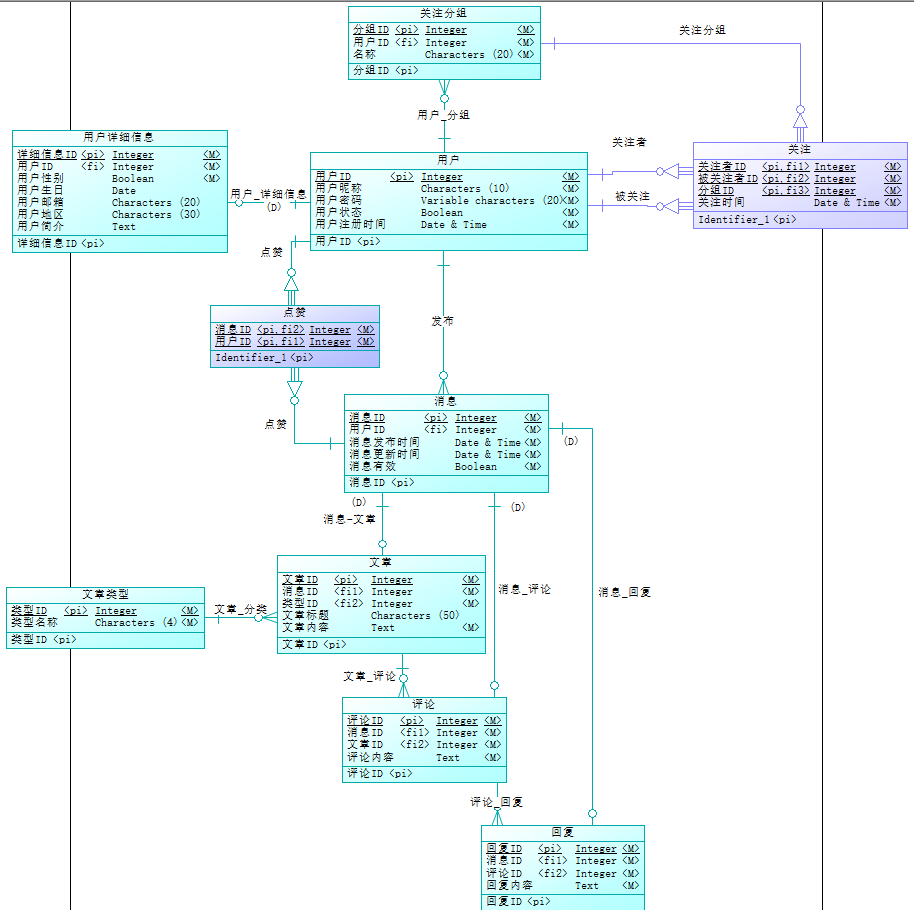
### ER图

*要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。*



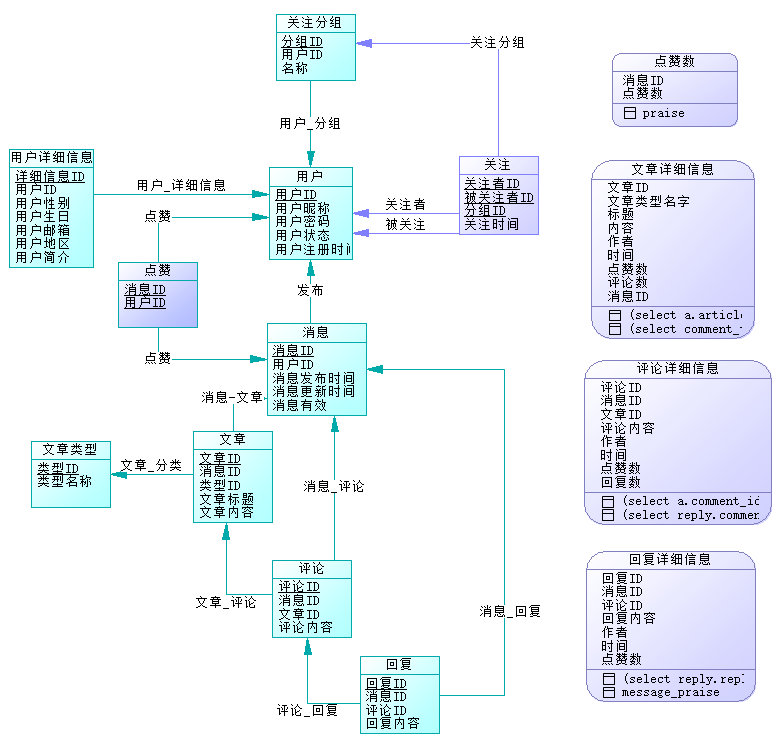
### LDM图

*要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。*



### PDM图

*要求：截图务必清晰，如果图太大可截图一个总图，然后截几个部分图展示。如果看不清截图会影响成绩。*

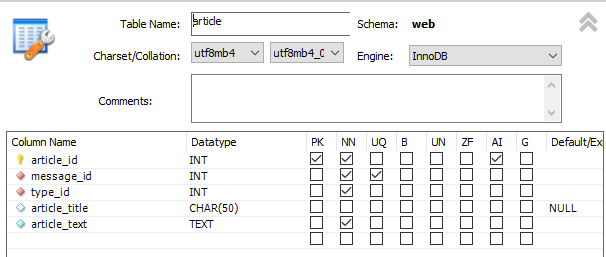
**

### 数据库表结构

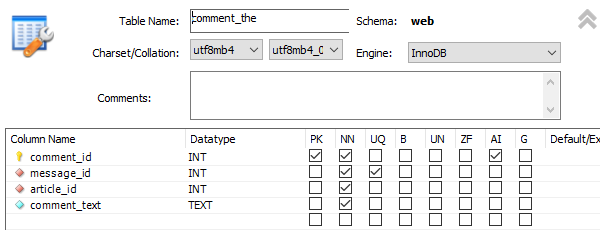
1. **表结构**

*选取几个比较有代表性的表结构截图，体现主键约束、外键约束、空值约束。*

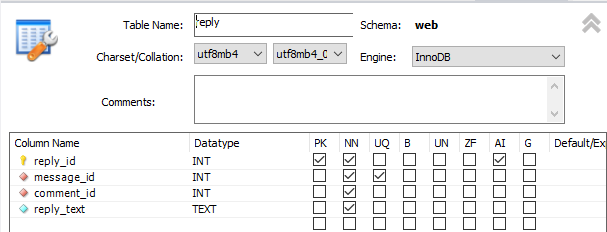
1. article文章表信息：
   1. Article\_id：文章id，主键，不能为空，插入自增
   2. Message\_id：消息id，该文章对应的消息id，外键，对应out\_message消息表，一个消息id唯一对应一篇文章。
   3. Type\_id：类型id，文章对应的类型id，外键，对应article\_type文章类型表，一个文章类型可以对应零个或多个文章。
   4. Article\_title：文章标题，可空
   5. Article\_text：文章内容，不可空



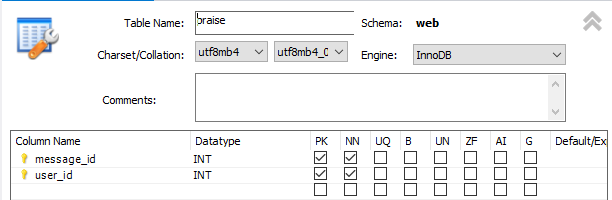
1. Comment\_the评论表信息：
   1. comment\_id：评论id，主键，不能为空，插入自增
   2. Message\_id：消息id，该评论对应的消息id，外键，对应out\_message消息表，一个消息id唯一对应一个评论。
   3. article\_id：文章id，评论对应的文章id，外键，对应article文章表，一个文章可以对应零个或多个评论。
   4. Article\_text：评论内容，不可空



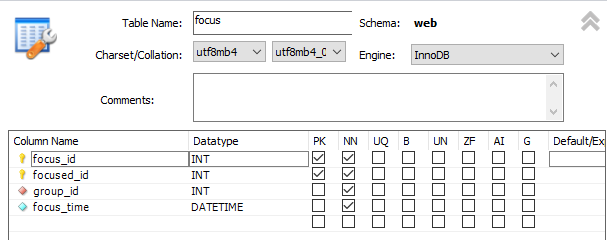
1. Reply回复表信息：
   1. reply\_id：回复id，主键，不能为空，插入自增
   2. Message\_id：消息id，该回复对应的消息id，外键，对应out\_message消息表，一个消息id唯一对应一个评论。
   3. Comment\_id：评论id，回复对应的评论id，外键，对应comment评论表，一个评论可以对应零个或多个回复。
   4. reply\_text：评论内容，不可空



1. praise点赞表：
   1. message\_id：消息id，外键做组合主键之一，对应out\_message消息表
   2. user\_id：用户id，外键做组合主键之一，对应userinfo用户表

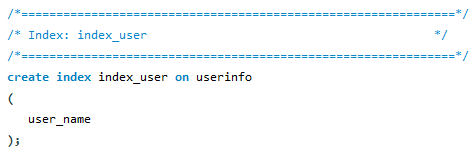


1. focus关注表：
   1. focus\_id：关注用户id，外键做组合主键之一，对应userinfo用户表
   2. focused\_id：被关注用户id，外键做组合主键之一，对应userinfo用户表
   3. group\_id：关注分组id，外键，对应group\_user关注分组表
   4. focus\_name：关注时间



1. **索引**

*讲解你建的非主索引，说明为什么要建这个索引。*



索引表如图：由于经常使用根据用户名搜索用户id，而用户表只有注册的时候会有增加，平时变化不大。索引有利于搜索速度的增加。

1. **视图**

*讲解你建的视图，说明为什么要建这个视图。*

1. Message\_praise消息点赞数表：根据消息id，快速得到点赞数。在微博的功能中，希望有文章，评论和回复的点赞数目。
   1. Message\_id：消息id
   2. Praise\_num：点赞数
2. Article\_detail 文章详细信息表：根据文章id，获得文章详细信息，有利于文章具体内容的显示。常用，且语句复杂，链接多个表。
   1. Article\_id：文章id
   2. Type\_name：文章类型
   3. Article\_title：文章标题
   4. Article\_text：文章内容
   5. User\_name：作者名称
   6. Message\_time：发布时间
   7. Praise\_num：点赞数
   8. Comment\_num：评论数
   9. Message\_id：文章对应消息id
3. Comment\_detail 评论详细信息表：根据评论id，获得评论详细信息，有利于评论具体内容的显示。常用，且语句复杂，链接多个表。
   1. comment\_id：评论id
   2. Message\_id：评论对应消息id
   3. Article\_id：评论的文章id
   4. comment\_text：评论内容
   5. User\_name：作者名称
   6. Message\_time：发布时间
   7. Praise\_num：点赞数
   8. reply\_num：回复数
4. reply\_detail 回复详细信息表：根据回复id，获得回复详细信息，有利于回复具体内容的显示。常用，且语句复杂，链接多个表。
   1. reply\_id：回复id
   2. Message\_id：回复对应消息id
   3. comment\_id：回复的评论id
   4. reply\_text：回复内容
   5. User\_name：作者名称
   6. Message\_time：发布时间
   7. Praise\_num：点赞数
5. **触发器（选做）**

*讲解你为什么建触发器和如何实现触发器。*

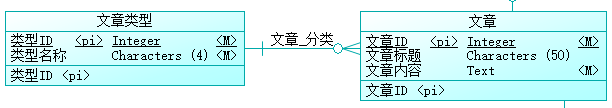
1. **事务（选做）**

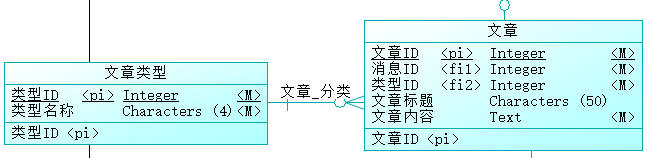
*讲解你为什么建事务和如何实现事务。*

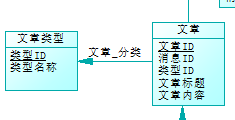
### 分析

*选择较为有代表性的实体和联系进行分析，讲解如何从ER图到LDM图、再到PDM图，最后变成数据库表结构的转换过程（参考实验二PPT里“补充知识 三、ER图🡪LDM图🡪PDM图”中老师讲的例子）。*

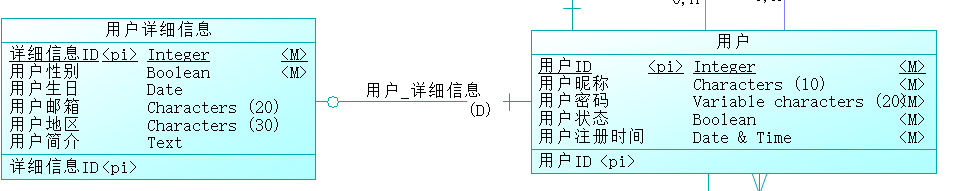
1. 文章分类：一篇文章有且仅有一个文章类型，一个文章类型可以有多个文章，一对多，文章类型id加入文章中。

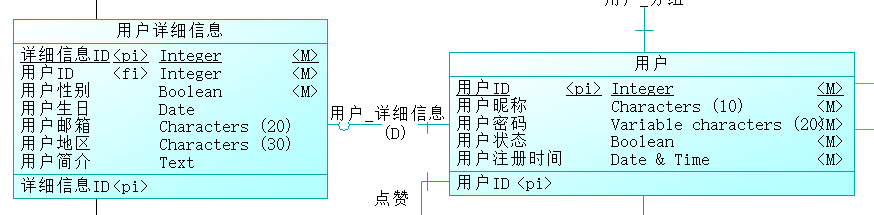
ER：

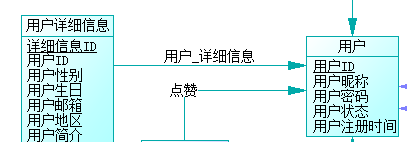
LDM：

PDM：

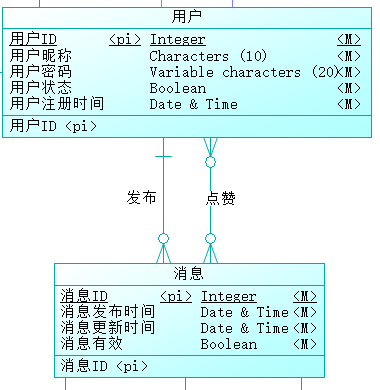
1. 用户及其详细信息：用户最多只有一个用户详细信息，每个用户详细信息有且对应一个用户，一对一，用户详细信息依赖于用户，用户id加入用户详细信息中。

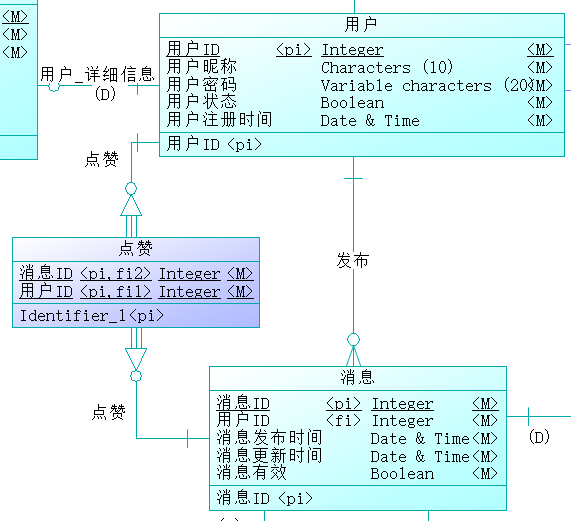
ER：

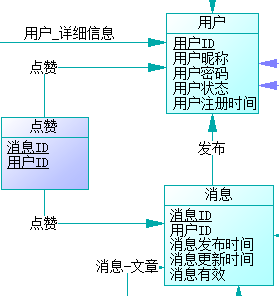
LDM：

PDM：

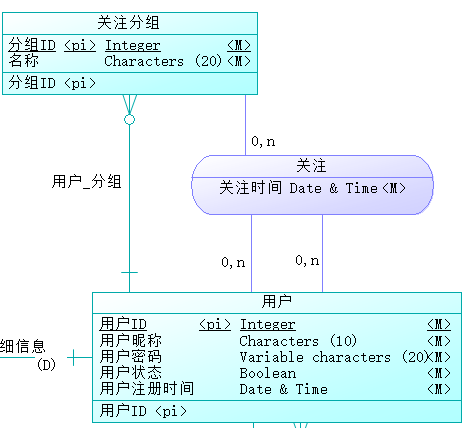
1. 点赞：每个用户可以给多条消息点赞，每条消息可以被多个用户点赞，多对多。创建新的表结构，以用户和消息主键作为联合主键

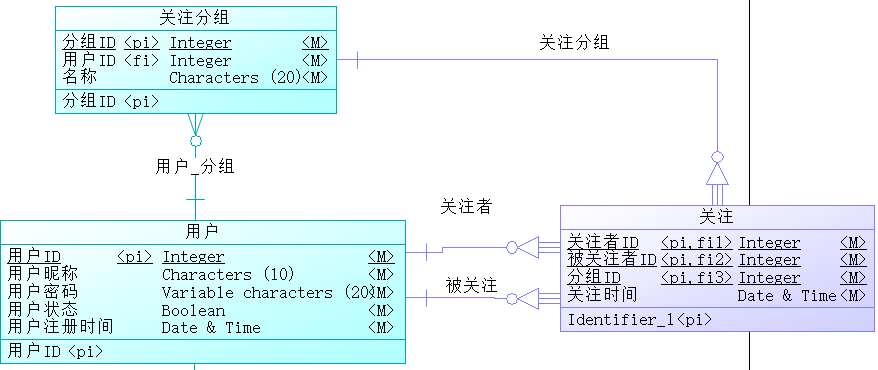
ER：

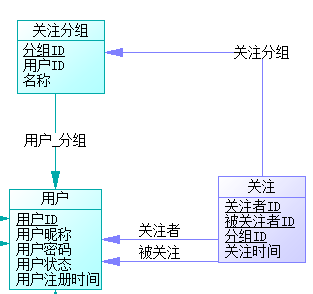
LDM：

PDM：

1. 关注：实体和联系多对多，实体主键加入联系中作为联合主键。

ER：

LDM：

PDM：

# 收获和反思

*请填写本次实验的收获，记录实验过程中出现的值得反思的问题及你的思考。*

收获满满。首先是数据库的设计以及各种实体之间的关系有了更深的理解，对于视图的用处和理解有所加深。其次是网页的开发有所进步。

反思：明明是数据库的实验，最后沉迷于界面，有点本末倒置。应该先完成内容，最后设计界面等等。

反思：在设计微博的时候，需要多次的返工数据库的设计，可以看出，光想的还是有点不切实际。

反思：最开始耗时很久，但是成果很少，自我感觉是把所有东西混到一起，没有明确的模块划分。