**School club数据库设计**

**项目成员：赖敏**

**纪眉妹**

**李佳琪**

**2016年11月22日**

**目录**

[第一章 3](#_Toc4015)

[1.1数据库 3](#_Toc1215)

[1.2目的和作用 3](#_Toc6818)

[1.3信息需求 3](#_Toc20207)

[第二章 概念结构设计 5](#_Toc4249)

[2.1 实体及其属性图的设计 5](#_Toc11931)

[2.2 局部E-R图的设计 7](#_Toc28547)

[2.3 全局E-R图的设计 9](#_Toc20526)

[第三章 逻辑结构设计 1](#_Toc10104)0

[第四章 数据库](#_Toc10104)实施...................................................................................................11

4.1创建关系表...................................................................................................11

4.2数据库实施代码...........................................................................................15

4.2.1创建数据库..........................................................................................15

4.2.1数据初始化...................................................................................................16

# 

# 第一章

## 1.1数据库

数据库管理系统 Microsoft SQL Server 2012

## 1.2目的和作用

将数据分析的结果进一步整理，形成最终的计算机模型，以便数据库开发人员建立物理数据库，方便其他项目小组人员查阅参考。

## 1.3信息需求

* 用户—社团：一个用户可以所属多个社团，一个社团也可拥有多个用户，所以用户与社团之间的关系为多对多。
* 用户—高校：一个高校可以有多个用户，但一个用户只能属于一个高校，所以用户与高校之间的关系为一对多。
* 用户—活动：一个用户可以参加多个活动，一个活动可以允许多个用户参加，所以用户与活动之间的关系为多对多。
* 用户—留言：一个用户可以发布多个留言，一个留言只对应一个用户，所以用户与留言之间的关系为多对一。
* 用户—帖子：一个用户可以发布多个帖子，一个帖子只对应一个用户，所以用户与帖子之间的关系为多对一。
* 用户—创业：一个用户可以发布多个创业信息，一个创业信息只对应一个用户，所以用户与创业之间的关系对多对一。
* 用户—关注：一个用户可以关注多个社团或活动，一个社团或活动也可被多个用户关注，所以用户与关注之间的关系为多对多。
* 社团—社团理事长：一个社团只有一个社团理事长，一个社团理事长对应一个社团，所以社团与社团理事长的关系为一对一。
* 社团—高校：一个社团只属于一个高校，一个高校有多个社团，所以社团与高校之间的关系为一对多。
* 社团—社团活动：一个社团可以发布多个活动，一个活动只对应一个社团，所以社团与活动之间的关系为多对一。
* 社团—赞助：一个社团可以发布多个赞助信息，一个赞助信息只对应一个社团，所以社团与赞助的关系为多对一。
* 高校—高校活动：一个高校可以发布多个活动，一个活动只对应一个高校，所以高校与活动之间的关系为多对一。
* 帖子—帖子回复：一个帖子包含多个帖子回复，一个回复只对应一个帖子，因此帖子和帖子回复之间的关系是一对多的关系。
* 赞助—赞助者：一个赞助对应多个赞助者，一个赞助者也可赞助多个项目，因此赞助和赞助者的关系为多对多。

# 

# 第二章 概念结构设计

## 2.1 实体及其属性图的设计

1. **用户属性图**

表2.1.1 用户属性

1. **社团属性图**

****

表2.1.2 社团属性图

1. **高校属性图**

****

表2.1.3 高校属性图

1. **帖子属性图**

****

表2.1.4 帖子属性图

1. **留言属性图**

****

表2.1.5 留言属性图

1. **资讯属性图**

****

表2.1.6 资讯属性图

1. **活动属性图**

****

表2.1.7 活动属性图

**8)管理员属性图**

****

表2.1.10 管理员属性图

## 2.2 局部E-R图的设计

1)**用户-社团-高校E-R图**



2)**用户-留言-帖子E-R图**

****

1. **高校-资讯E-R图**

****

1. **用户/社团/赞助方-活动E-R图**

****

## 2.3 全局E-R图的设计

****

# 第三章 逻辑结构设计

根据E-R图中的实体及实体之间的联系，关系模式如下（下划线为主键；#为外键）：

用户（用户ID，用户名，用户账号，登录密码，高校ID#，社团ID#，邮箱，手机号，头像）

高校（高校ID，高校名，登录密码）

社团（社团ID，社团名，社团账号，登录密码，高校ID#，社团简介，社团封面）

社团留言（社团留言ID，社团留言时间，社团留言内容，社团ID#）

帖子（帖子ID，标题，内容，发布时间，用户ID#）

帖子留言（帖子留言ID，帖子留言时间，帖子留言内容，帖子ID#）

回复（回复ID，回复时间，回复内容，用户ID#）

资讯（资讯ID，发布时间，内容，高校ID#）

活动（活动ID，活动名，活动内容，活动开始时间，活动结束时间，活动图片）

活动留言（活动留言ID，活动留言时间，活动留言内容，活动ID#）

关注（用户ID#，活动ID#，关注时间）

？？报名参加（用户ID#，活动ID#，报名时间，是否参加，参与ID，参与时间）

发布（社团ID#，活动ID#，发布时间）

管理员（管理员ID，管理员名称，管理员密码）

# 数据库实施

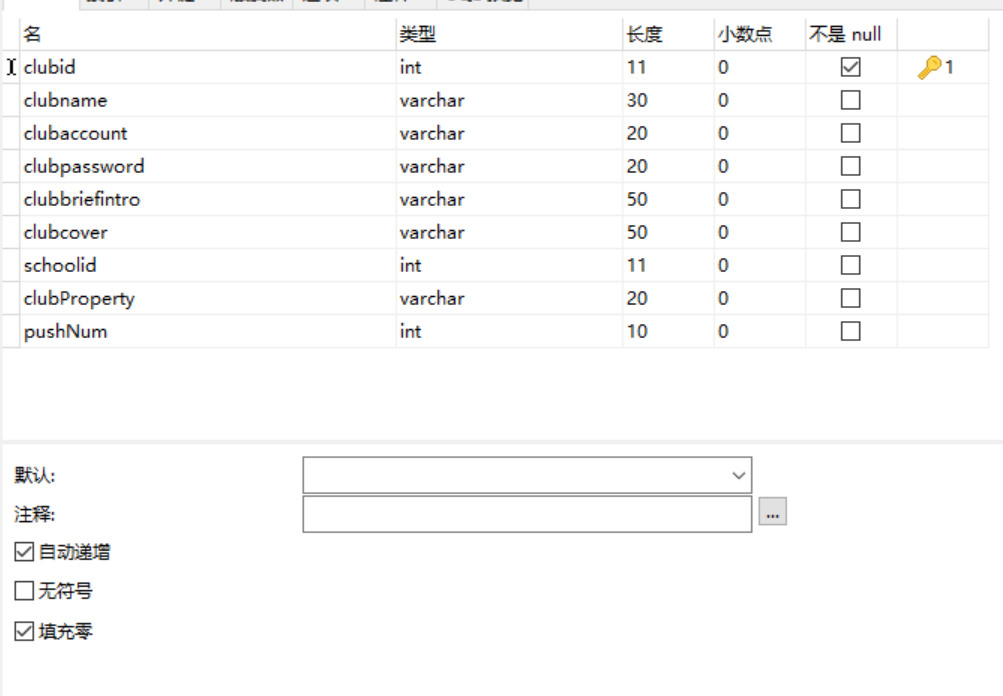
## 4.1创建关系表

将前面得到的关系模式转换为MySQL支持的具体关系表如下：

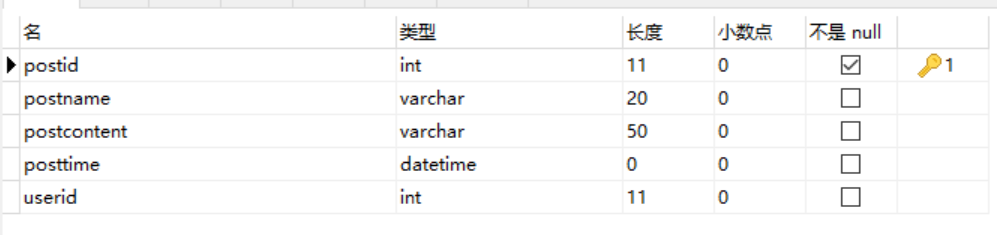
1. t\_user:用户表



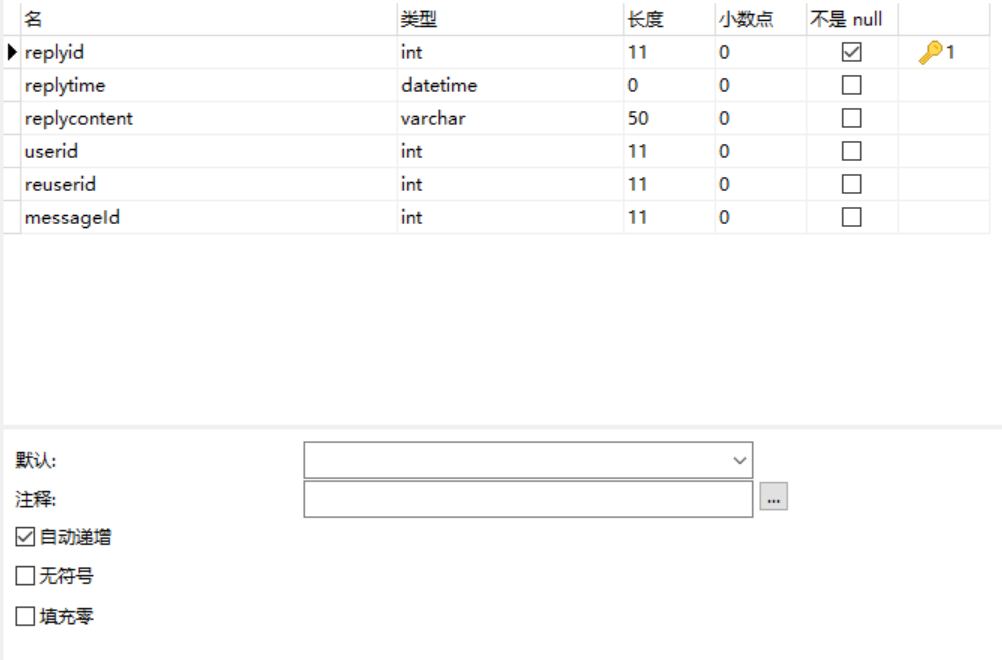
1. t\_club：社团表



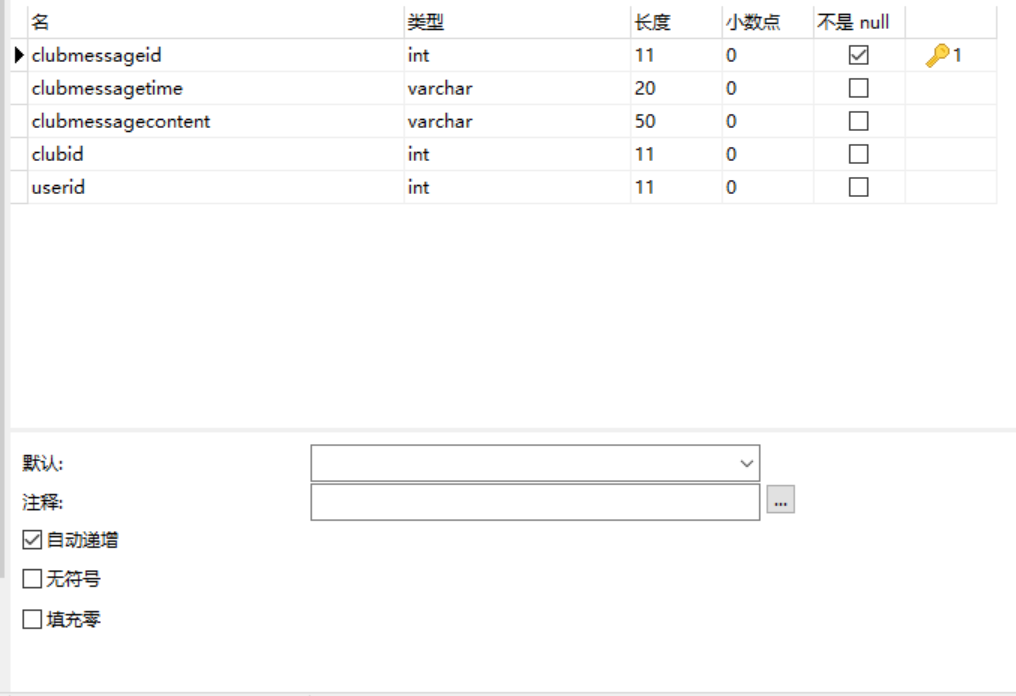
1. t\_school:高校表
2. t\_post:帖子表



5)t\_reply:评论/回复表



1. clubmessage:社团留言表



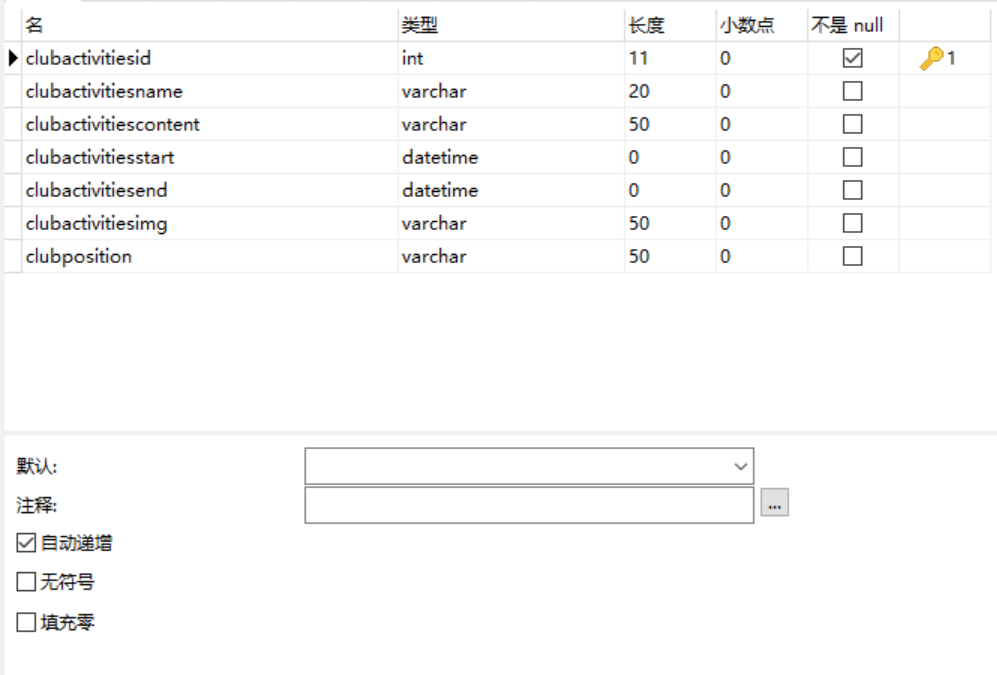
1. postmessage:帖子留言表



1. schoolnews:校园资讯表



1. clubactivities:社团活动表



1. Clubactivitiesmessage:社团留言表



## 4.2 数据库实施代码

## 4.2.1 创建数据库

USE schoolclub

GO

CREATE DATABASE schoolclub

ON

( NAME = schoolclub

FILENAME = 'd:\schoolclub .mdf',

SIZE = 10,

MAXSIZE = 50,

FILEGROWTH = 5 )

LOG ON

( NAME = SMMS ,

FILENAME = 'd:\schoolclub .ldf',

SIZE = 5MB,

MAXSIZE = 25MB,

FILEGROWTH = 5MB )

GO

## 4.2.2 数据初始化

1)社团表

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

-- Table structure for `t\_club`

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

DROP TABLE IF EXISTS `t\_club`;

CREATE TABLE `t\_club` (

`clubid` int(11) NOT NULL AUTO\_INCREMENT,

`clubname` varchar(30) DEFAULT NULL,

`clubaccount` varchar(20) DEFAULT NULL,

`clubpassword` varchar(20) DEFAULT NULL,

`clubbriefintro` varchar(50) DEFAULT NULL,

`clubcover` varchar(50) DEFAULT NULL,

`schoolid` int(11) DEFAULT NULL,

`clubProperty` varchar(20) DEFAULT NULL,

`pushNum` int(10) DEFAULT NULL,

PRIMARY KEY (`clubid`),

KEY `FK\_Reference\_4` (`schoolid`),

CONSTRAINT `FK\_Reference\_4` FOREIGN KEY (`schoolid`) REFERENCES `t\_school` (`schoolid`)

) ENGINE=InnoDB AUTO\_INCREMENT=29 DEFAULT CHARSET=utf8;

2)社团活动表

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

-- Table structure for `t\_clubactivities`

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

DROP TABLE IF EXISTS `t\_clubactivities`;

CREATE TABLE `t\_clubactivities` (

`clubactivitiesid` int(11) NOT NULL AUTO\_INCREMENT,

`clubactivitiesname` varchar(20) DEFAULT NULL,

`clubactivitiescontent` varchar(50) DEFAULT NULL,

`clubactivitiesstart` datetime DEFAULT NULL,

`clubactivitiesend` datetime DEFAULT NULL,

`clubactivitiesimg` varchar(50) DEFAULT NULL,

`clubposition` varchar(50) DEFAULT NULL,

PRIMARY KEY (`clubactivitiesid`)

) ENGINE=InnoDB AUTO\_INCREMENT=1014 DEFAULT CHARSET=utf8;

3)报名参加名单表

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

-- Table structure for `t\_entryform`

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

DROP TABLE IF EXISTS `t\_entryform`;

CREATE TABLE `t\_entryform` (

`entryformid` int(11) NOT NULL AUTO\_INCREMENT,

`userid` int(11) DEFAULT NULL,

`clubactivitiesid` int(11) DEFAULT NULL,

`entrytime` datetime DEFAULT NULL,

`jointime` datetime DEFAULT NULL,

PRIMARY KEY (`entryformid`),

KEY `FK\_Reference\_16` (`clubactivitiesid`),

KEY `FK\_Reference\_17` (`userid`),

CONSTRAINT `FK\_Reference\_16` FOREIGN KEY (`clubactivitiesid`) REFERENCES `t\_clubactivities` (`clubactivitiesid`),

CONSTRAINT `FK\_Reference\_17` FOREIGN KEY (`userid`) REFERENCES `t\_user` (`userid`)

) ENGINE=InnoDB AUTO\_INCREMENT=10014 DEFAULT CHARSET=utf8;

4)帖子表

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

-- Table structure for `t\_post`

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

DROP TABLE IF EXISTS `t\_post`;

CREATE TABLE `t\_post` (

`postid` int(11) NOT NULL,

`postname` varchar(20) DEFAULT NULL,

`postcontent` varchar(50) DEFAULT NULL,

`posttime` datetime DEFAULT NULL,

`userid` int(11) DEFAULT NULL,

PRIMARY KEY (`postid`),

KEY `FK\_Reference\_3` (`userid`),

CONSTRAINT `FK\_Reference\_3` FOREIGN KEY (`userid`) REFERENCES `t\_user` (`userid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

1. 帖子回复表

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

-- Table structure for `t\_reply`

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

DROP TABLE IF EXISTS `t\_reply`;

CREATE TABLE `t\_reply` (

`replyid` int(11) NOT NULL AUTO\_INCREMENT,

`replytime` datetime DEFAULT NULL,

`replycontent` varchar(50) DEFAULT NULL,

`userid` int(11) DEFAULT NULL,

`reuserid` int(11) DEFAULT NULL,

`messageId` int(11) DEFAULT NULL,

PRIMARY KEY (`replyid`),

KEY `FK\_Reference\_7` (`userid`),

KEY `reuserid` (`reuserid`),

KEY `messageId` (`messageId`),

CONSTRAINT `FK\_Reference\_7` FOREIGN KEY (`userid`) REFERENCES `t\_user` (`userid`),

CONSTRAINT `t\_reply\_ibfk\_1` FOREIGN KEY (`reuserid`) REFERENCES `t\_user` (`userid`),

CONSTRAINT `t\_reply\_ibfk\_2` FOREIGN KEY (`messageId`) REFERENCES `clubmessage` (`clubmessageid`)

) ENGINE=InnoDB AUTO\_INCREMENT=10000050 DEFAULT CHARSET=utf8;

6)校园资讯表

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

-- Table structure for `t\_schoolnews`

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

DROP TABLE IF EXISTS `t\_schoolnews`;

CREATE TABLE `t\_schoolnews` (

`schoolnewsid` int(11) NOT NULL,

`schoolnewstime` varchar(20) DEFAULT NULL,

`schoolnewscontent` varchar(50) DEFAULT NULL,

`schoolid` int(11) DEFAULT NULL,

PRIMARY KEY (`schoolnewsid`),

KEY `FK\_Reference\_8` (`schoolid`),

CONSTRAINT `FK\_Reference\_8` FOREIGN KEY (`schoolid`) REFERENCES `t\_school` (`schoolid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

7) 用户表

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

-- Table structure for `t\_user`

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

DROP TABLE IF EXISTS `t\_user`;

CREATE TABLE `t\_user` (

`userid` int(11) NOT NULL AUTO\_INCREMENT,

`schoolid` int(11) DEFAULT NULL,

`nickname` varchar(100) DEFAULT NULL,

`username` varchar(20) DEFAULT NULL,

`password` varchar(32) NOT NULL,

`clubid` int(11) DEFAULT NULL,

`email` varchar(20) NOT NULL,

`telnumber` varchar(20) DEFAULT NULL,

`headImg` varchar(50) DEFAULT NULL,

`role` int(11) NOT NULL,

PRIMARY KEY (`userid`),

KEY `FK\_Reference\_1` (`schoolid`),

KEY `FK\_Reference\_13` (`clubid`),

CONSTRAINT `FK\_Reference\_1` FOREIGN KEY (`schoolid`) REFERENCES `t\_school` (`schoolid`),

CONSTRAINT `FK\_Reference\_13` FOREIGN KEY (`clubid`) REFERENCES `t\_club` (`clubid`)

) ENGINE=InnoDB AUTO\_INCREMENT=21 DEFAULT CHARSET=utf8;

1. 用户参加活动表

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

-- Table structure for `t\_userandactivity`

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

DROP TABLE IF EXISTS `t\_userandactivity`;

CREATE TABLE `t\_userandactivity` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`userId` int(11) NOT NULL,

`activityId` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=12 DEFAULT CHARSET=utf8;