# 其他

## 模板：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **类型** | **注释** | **备注** |
|  |  |  |  |

# 我的

## 登录/注册

**User\_info**:用户信息表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 注释 | 备注 |
| id | int(4) | 身份ID | 主键 |
| phone | Varchar(12) | 电话号码 |  |
| nickname | varchar(22) | 昵称 | 必填 |
| gender | tinyint(1) | 性别 | 1男0女（默认为1） |
| birthday | Varchar(12) | 出生日期 | 格式:1997-11-20 |
| heart\_word | Varchar(40) | 个性签名 |  |
| Password | Varchar(16) | 密码 | 8-16位 |
| Head\_pic | Varchar(20) | 头像 | 只记录文件单号 |

**建表语句：**

create table user\_info(

id int unsigned auto\_increment,

phone varchar(12),

nickname varchar(12) not null,

gender tinyint(1) default 1 check(gender in(1,0)),

birthday varchar(12) default '1997-11-20',

heart\_word varchar(40) default '快来发布心情吧~',

password varchar(16) not null,

head\_pic varchar(20),

primary key (id)

)ENGINE= InnoDB default charset=utf8;

## feed\_back反馈

### 建表语句

create table feed\_back(

-> id int(4) unsigned auto\_increment primary key,

-> uid int(4) unsigned not null,

-> feed\_back tinytext not null,

-> time bigint(8) unsigned not null,

-> status int(2) unsigned not null default 0,

-> index(uid),

-> foreign key(uid) references user\_info(id) on delete cascade on update cascade

-> )ENGINE=InnoDB default charset=utf8;

# 广场

## 动态

### dynamic:动态表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 注释 | 备注 |
| id | Int(4) | 动态id | 主键 |
| Uid | Int(4) | 发布人id | 必填 |
| time | Bigint(8) | 发布时间，毫秒数 | 必填 |
| awesome | Int(2) | 点赞个数 | 默认0 |
| comment | Int(2) | 评论条数 | 默认0 |
| Content | tinytext | 动态内容 | 必填 |
|  |  |  |  |

**建表语句：**

Create table dynamic(

id int(4) unsigned auto\_increment primary key,

uid int(4) unsigned not null,

like int(2) unsigned default 0,

comment int(2) unsigned default 0,

content varchar(50) not null,

time bigint(8) unsigned not null

)ENGINE= InnoDB default charset=utf-8;

### comment:评论表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 注释 | 备注 |
| id | Int(4) | 评论id | 主键 |
| Uid | Int(4) | 发布人id | 必填 |
| time | Bigint(8) | 发布时间，毫秒数 | 必填 |
| awesome | Int(2) | 点赞个数 | 默认0 |
| comment | Int(2) | 子评论条数 | 默认0 |
| Content | tinytext | 评论内容 | 必填 |
| Parent\_id | Int(4) | 父(动态/评论)id | 必填 |

**建表语句：**

Create table comment(

id int(4) unsigned auto\_increment primary key,

uid int(4) unsigned not null,

time bigint(8) unsigned not null,

awesome int(2) unsigned default 0,

comment int(2) unsigned default 0,

content tinytext not null,

parent\_id int(4) unsigned not null,

index(parent\_id), //外键一定要加索引

foreign key(parent\_id) references dynamic(id) on delete cascade on update cascade

) ENGINE= InnoDB default charset=utf8;

select a.id,if((select uid from awesome b where a.id=b.dynamic\_id and b.uid=1001) is null,0,1) as ok from dynamic a;

### sub\_comment:子评论(回复)表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段 | 类型 | 注释 | 备注 |
| id | Int(4) | 评论id | 主键 |
| Uid | Int(4) | 发布人id | 必填 |
| time | Bigint(8) | 发布时间，毫秒数 | 必填 |
| awesome | Int(2) | 点赞个数 | 默认0 |
| Content | tinytext | 评论内容 | 必填 |
| Parent\_id | Int(4) | 父(动态/评论)id | 必填 |
| to\_uid | Int(4) | 被回复者id,(level2) |  |

**建表语句：**

Create table sub\_comment(

id int(4) unsigned auto\_increment primary key,

uid int(4) unsigned not null,

time bigint(8) unsigned not null,

awesome int(2) unsigned default 0,

content tinytext not null,

parent\_id int(4) unsigned not null,

to\_uid int(4) unsigned,

index(parent\_id), //外键一定要加索引

foreign key(parent\_id) references comment(id) on delete cascade on update cascade

) ENGINE= InnoDB default charset=utf8;

### awesome ：点赞

#### 建表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **类型** | **注释** | **备注** |
| id | Int(4) | id | 主键自增 |
| dynamic\_id | Int(4) | 动态id | Not null, |
| uid | Interesting(4) | 用户id | Not null |
|  |  |  |  |

Create table awesome(

id int(4) unsigned primary key auto\_increment,

dynamic\_id int(4) unsigned not null,

uid int(4) unsigned not null,

unique key(dynamic\_id,uid),

index(dynamic\_id),

foreign key(dynamic\_id) references dynamic(id) on delete cascade on update cascade

) ENGINE= InnoDB default charset=utf8;

#### 触发器：

添加：

delimiter ||

create trigger add\_awesome

after insert

on awesome for each row

begin

update dynamic set awesome=awesome+1 where NEW.dynamic\_id=dynamic.id;

end ||

delimiter ;

删除：

delimiter ||

create trigger del\_awesome

after delete

on awesome for each row

begin

update dynamic set awesome=awesome-1 where OLD.dynamic\_id=dynamic.id;

end ||

delimiter ;

### sub\_awesome 动态赞表

#### 建表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **类型** | **注释** | **备注** |
| id | Int(4) | id | 主键自增 |
| comment\_id | Int(4) | 评论id | Not null, |
| uid | int(4) | 用户id | Not null |
|  |  |  |  |

Create table sub\_awesome(

id int(4) unsigned primary key auto\_increment,

comment\_id int(4) unsigned not null,

uid int(4) unsigned not null,

unique key(comment\_id,uid),

index(comment\_id),

foreign key(comment\_id) references comment(id) on delete cascade on update cascade

) ENGINE= InnoDB default charset=utf8;

#### 触发器

添加：

delimiter ||

create trigger add\_sub\_awesome

after insert

on sub\_awesome for each row

begin

update comment set awesome=awesome+1 where NEW.comment\_id=comment.id;

end ||

delimiter ;

删除：

delimiter ||

create trigger del\_sub\_awesome

after delete

on sub\_awesome for each row

begin

update comment set awesome=awesome-1 where OLD.comment\_id=comment.id;

end ||

delimiter ;

### 触发器：

#### 动态删除，关联评论也删除(存在本表锁定问题，需要用时间触发器解决)

delimiter ||

create trigger del\_dynamic

after delete

on dynamic for each row

begin

delete from comment where parent\_id = OLD.id ;

end ||

delimiter ;

#### 评论删除，关联子评论删除(存在本表锁定问题，需要用时间触发器解决)

delimiter ||

create trigger del\_comment

after delete

on comment for each row

begin

update dynamic set comment=comment-1 where OLD.parent\_id=id;

delete from sub\_comment where parent\_id = OLD.id ; //暂时用级联删除

end ||

delimiter ;

#### 子评论删除(存在本表锁定问题，需要用时间触发器解决)

delimiter ||

create trigger del\_sub\_comment

after delete

on sub\_comment for each row

begin

update comment set comment=comment-1 where OLD.parent\_id=id;

end ||

delimiter ;

#### 添加评论

delimiter ||

create trigger add\_comment

after insert on comment for each row

begin

update dynamic set comment=comment+1 where NEW.parent\_id=id;

end ||

delimiter ;

#### 添加子评论

delimiter ||

create trigger add\_sub\_comment

after insert on sub\_comment for each row

begin

update comment set comment=comment+1 where NEW.parent\_id=id;

end ||

delimiter ;

# 服务/主页

## 校园帮帮

### School\_help表：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **类型** | **注释** | **备注** |
| id | Int(4) | 帮助id | 主键 |
| uid | Int(4) | 发布人id | 必填 |
| Time | Bitint(8) | 发布时间 | 必填 |
| Title | Varchar(20) | 标题 | 必填 |
| Content | Tinytext | 内容 | 必填 |
| browser | Int(4) | 浏览量 |  |
| Bounty | Int(4) | 赏金 |  |
| status | Int(2) | 状态 | 0 待提交 1待确认 2确认 3驳回 |
| Picture | Varchar(20) | 配图 | 只记录单号 |

建表语句：

Create table school\_help(

id int(4) unsigned auto\_increment primary key,

uid int(4) unsigned not null,

time bigint(8) unsigned not null,

title varchar(20) not null,

content tinytext not null,

browser int(4) unsigned default 0,

bounty int(4) unsigned default 0,

status int(2) unsigned default 0,

picture varchar(20)

) ENGINE= InnoDB default charset=utf8;

## 物品遗拾

### stuff

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **类型** | **注释** | **备注** |
| Id | Int(4) | 物品id | 主键 |
| Uid | Int(4) | 发布人 | 必填 |
| Time | Bigint(4) | 时间 | 必填 |
| Content | Tinytext | 内容 | 必填 |
| Picture1 | Varchar(20) | 配图1 |  |
| Picture2 | Varchar(20) | 配图2 |  |
| Hot | Int(4) | 热度 |  |
| Keyword | Varchar(20) | 关键字 |  |

建表语句

Create table stuff(

id int(4) unsigned auto\_increment primary key,

uid int(4) unsigned not null,

time bigint(8) unsigned not null,

content tinytext not null,

picture1 varchar(10),

picture2 varchar(10),

hot int(4) unsigned not null default 0,

keyword varchar(20) not null

) ENGINE= InnoDB default charset=utf8;

create table blog( id int(4) unsigned primary key, title varchar(100) not null, summary tinytext not null, descrip varchar(100) ) not null, url varchar(80) not null )ENGINE=InnoDB default charset=utf8;