

接口文档说明:

蓝色字的接口是最近新添加的, 黑色字的接口是上次就有的, 没有修改。另外, 小组名目前是不可重复的。

注意插入数据时, 要传输完整的数据, 如果某些内容为空, 则将其设为"", 因数据库中  
没有判断变量是否存在, 所以不设的话会数据库操作报错。(这个我看看要不要再修改。。。)

如有疏漏, 请尽快告知

By 邝婉静 电话: 15920386181

用户操作:

**insert\_new\_user(\$para):**

\$para 为关联数组, 需要插入的信息包括: 名字(name), 昵称(nickname),  
密码(password), 性别(gender), 真实姓名(realname),  
电子邮件(email), 个人简介(description), 注册日期(setuptime)  
(注意: 若某一项为空, 则设为空传给数据库)

```
$para['name'];  
$para['password']  
$para['gender']  
$para['realname']  
$para['email']  
$para['description']  
$para['setuptime']
```

注意在添加用户前先检查用户名是否已经存在。

**has\_user\_name(\$username):**

输入一个用户名, 检测该用户是否已经存在, 是的话, 返回 true, 否则返回 false。

**delete\_user\_by\_username(\$name):**

根据用户名删除一个用户。

**get\_user\_by\_name(\$name):**

该函数接收一个用户名, 返回整个用户信息。

返回一个关联数组:

```
'uid' -- 用户 id  
'uname' -- 用户名  
'npwd' -- 用户密码  
'gender' -- 用户性别  
'urealname' -- 用户真实姓名  
'email' -- 用户电子邮件  
'description' -- 用户个人说明  
'setuptime' -- 用户注册时间
```

**get\_user\_by\_id(\$id):**

根据 id 返回一个用户，返回的是一个关联数组，返回格式同上。

`update_user_password_by_name($name, $pwd):`

该函数接收一个用户名， 一个新的密码，执行后更改用户的密码为新的密码 `pwd`。

`update_user_info_by_name($name, $infolist, $newinfo):`

该函数接收一个用户名， 一个修改项列表， 以及对应修改项的新的值。

修改项数组的值可以为:

'uid' --用户 id

'uname' --用户名

'npwd' -- 用户密码

'gender' -- 用户性别

'urealname' -- 用户真实姓名

'email' -- 用户电子邮件

'description' -- 用户个人说明

'setuptime' -- 用户注册时间

对应修改项上写更改后的值。比如需要更改用户的 email， 则 `$infolist[] = "email";`

`$newinfo[$infolist[]] = "new email @ xx.com";`

好友操作:

`insert_friend($username1, $username2):`

该函数接受两个用户名， 返回 `true` 则表示插入好友关系成功， 否则， 插入不成功， 两人已经是好友。

`check_friends($username1, $username2):`

该函数接收两个用户名， 返回 `false` 表示不是好友， 返回 `true` 表示是好友。

`get_friends_by_user_name($username):`

该函数接收一个用户名， 返回一个 `uid` 数组表示该用户的所有好友 id。

`delete_friend_by_names($username1, $username2):`

该函数接收两个用户名， 如果两个用户是好友的话， 就删除好友关系。

小组操作: (暂时小组名， 不能重复)

`create_group($para, $members):`

该函数接收 `$para` 和 `$members`， 前者是一个组的信息， 包括 `groupname`， 组的名字， `groupadmin`， 小组组长。

`$members` 是小组其他成员用户名的列表。

`get_group_info_by_name($groupname):`

该函数接收一个组名， 返回一个组的组名、 组长 id。 返回空表示没有这个小组。(还没判断小组名字是否一样的行为)。

`get_group_members_by_name($groupname):`

该函数接收一个组名，返回组的所有成员的用户 id。

`add_group_member($gid, $username):`

输入小组 id 及要添加的用户，将该用户添加到小组里面。

`has_group_name($groupname):`

判断是否已经存在该小组。

`dismiss_group_by_group_name($groupname):`

给一个小组名，解散该小组。

`get_all_groups_by_name($username):`

给出一个用户名，查找该用户所属的小组。返回小组 id 号数组。不是关联数组，直接用数字索引就能访问到每一个小组 id。

`kick_one_out_of_group($groupname, $username):`

给出小组的名字，已经小组内某一个成员的名字，将该成员从小组中剔除。

资源操作：

`insert_resource($para):`

\$para 为关联数组  
'rtitle' --资源标题,  
'rgeneral' --资源概述  
'setupername', --资源创建者  
'setupdate' --资源创建时间

`get_resources_by_title($id):`

该函数作废！

`get_resource_by_id($id):`

该函数接收一个资源 id，返回资源全部内容，返回的至多是一个资源。

rid --id,  
rtitle --标题,  
rgeneral --概述,  
setuper --创建者 id,  
setuptime --创建日期

`get_top_n_resources($n):`

该函数接收一个整数，输出数据库前 n 条资源信息。返回的也是一个关联数组，格式同

上。

`get_resources_by_setuper($username):`

该函数接收一个创建者的用户名，然后返回该用户创建的所有资源。  
格式同上。

评论操作：

`insert_commentary($para):`

给某一资源插入一条评论，\$para 为一关联数组，格式如下：

\$para['rid'] --资源 id  
\$para['title'] --资源题目  
\$para['content'] --评论内容  
\$para['setupername'] --创建的用户名  
\$para['setupdate'] --创建日期

`get_commentary_by_id($cid):`

接收一条评论的 id，将该评论的内容提出到一个关联数组中。

关联数组的格式如下：

'cid' --评论 id  
'rid' --评论的资源对象  
'ctitle' --评论标题  
'content' --评论内容  
'setuper' --评论创建者  
'setuptime' --评论创建时间

`get_resource_commentaries_by_id($rid):`

接收一个资源的 id，将关于该资源的评论返回到一个数组。该数组的每一个元素都是一条评论关联数组，格式同上。

`update_resource_commentary_by_id($cid, $content):`

接收一条评论的 id 以及要更新的内容，更新该条评论的内容。

`delete_resource_commentary_by_id($cid):`

删除一条评论，接收一个评论的 id，操作者是评论的创建者。

笔记操作：

`insert_note($para):`

该函数接收一个关联数组，将关联数组的信息作为一个 note 插入表中，格式如下：

\$para['resoucesid']

\$para['privacy'] --note 的权限。 1 代表只有自己才能看到。2 代表朋友公开，4 代表小

组内公开，6 代表公开给朋友和小组!!!!!!!

```
$para['title'] --note 标题  
$para['content'] -- note 内容  
$para['setuper'] --创建人用户名  
$para['setuptime'] --创建时间
```

**get\_notes\_by\_username(\$name):**

该函数查找用户名为\$name 的用户做过的所有笔记。返回一个关联数组，格式如下：

```
$para['nid'] --noteid  
$para['rid'] --对应资源的 id  
$para['privacy'] --对应的公开度  
$para['ntitle'] --note 标题  
$para['content'] -- note 内容  
$para['setuper'] --创建人用户名  
$para['setuptime'] --创建时间
```

**delete\_note\_by\_id(\$id):**

该函数删除用户特定 note id 的笔记。

**get\_private\_notes\_by\_username(\$username):**

提取该用户的隐私笔记。返回为一个关联数组。格式如下：

```
$para['nid'] --noteid  
$para['rid'] --对应资源的 id  
$para['privacy'] --对应的公开度  
$para['ntitle'] --note 标题  
$para['content'] -- note 内容  
$para['setuper'] --创建人用户名  
$para['setuptime'] --创建时间
```

**get\_friends\_public\_notes\_by\_username(\$username):**

提取该用户对好友公开的笔记。格式如上。

**get\_group\_public\_notes\_by\_username(\$username):**

提取该用户对小组公开的笔记。格式如上。

消息处理：

我将三种消息都放在同一个叫 MESSAGES 的表格中，表的属性有如下：

mid CHAR(10), --id

mreceiver CHAR(10), --发出者，如果是小组发出的消息，此处就是小组管理员。

msender CHAR(10), --接收者，如果是某用户请求加入某小组的消息，此处就是小组管理员。

mcontent INT, --0 表示添加好友消息，1 表示小组邀请消息，2 表示请求加入小组信息。

option CHAR(10) --如果是关于小组的消息，这里放的是小组的 gid。

输入的都是名字，包括小组名和用户名。

#### `insert_making_friends_message($para):`

该函数在某用户想加另一用户为好友的时候调用，格式如下：

`$para['receiver']` 请求接收者。

`$para['sender']` 请求发出者。

#### `insert_inviting_group_message($para):`

该函数发出的是一条邀请用户加入某一个小组的消息，输入一个关联数组，格式如下：

`$para['receiver']` 请求接收者。

`$para['sender']` 请求发出者。 这里是小组的名字。

#### `insert_applying_group_message($para):`

该函数发出一条某用户请求加入某小组的消息，接收方是小组管理员。

`$para['receiver']` 请求接收者。这里是小组的名字。

`$para['sender']` 请求发出者。

#### `get_messages_by_username($username):`

取出一个用户名下所有的消息。

#### `delete_message($mid):`

删除某条信息。

话题操作：

#### `insert_topic($para):`

给某一个小组添加话题：

输入关联数组如下：

`$para['groupid']` 小组 id

`$para['setuper']` 创建者用户名

`$para['setuptime']` 创建日期

`$para['title']` 话题标题

`$para['general']` 话题概述。

#### `get_group_topics_groupname($groupname):`

返回一个小组的所有话题，输入一个小组的名字。

返回一个关联数组。格式如下：

`$para['tid']` 话题 id

`$para['gid']` 小组 id

`$para['setuper']` 创建者 id

`$para['setuptime']` 创建日期

`$para['title']` 话题标题

`$para['general']` 话题概述。

`delete_group_topic_groupname($topicid):`

输入一个话题 id，删除该话题。

话题回复：

`insert_topic_response($para):`

\$para['tid'] 对应话题 id

\$para['setuper'] 创建者名字。

\$para['setupdate'] 创建日期。

\$para['title'] 回复标题

\$para['general'] 回复内容

`get_topics_responses($tpid):`

拿到某一个话题 id 下所有回复。返回一个关联数组，格式如下：

trid 回复 id

tid 话题 id

setuper 创建者 id

setuptime 日期

title 标题

general 主题内容