|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | 完成后的会员级别 |
| 新会员注册 | 向推荐人发红包 | 向上5层5级会员发红包【由谁发?注册人?】 | **0** |
| 发展人员>=2  由0级申请到1级 | 向上面第2层的2级会员发红包 |  | **1** |
| 发展人员>=4  由1级申请到2级 | 向上面第3层的3级会员发红包 |  | **2** |
| 发展人员>=8  由2级申请到3级 | 向上面第4层的4级会员发红包 |  | **3** |
| 发展人员>=16  由3级申请到4级 | 向上面第5层的5级会员发红包 |  | **4** |
| ……… |  |  |  |
| 发展人员>=256  由8级申请到9级 | 向上面第10层的9级会员发红包 |  | **9** |

会员升到级别9之后则无需继续升级。

在申请升级的时候,如果发现上面第N层的级别不够N,则继续往上查找级别>=N的第一个会员发红包

**Member**: 会员状态

|  |  |
| --- | --- |
| uid | 用户ID 自增 |
| ruid | 推荐人ID |
| puid | 接点人ID 就是parentUid |
| reaches | 层级 顶层会员是0 依次往下增加 |
| level | 会员等级 |
| bdirection | 方向 0 左 1 中 2 右 |
| b\_left\_uid | 左下级Uid |
| b\_middle\_uid | 中下级Uid |
| b\_right\_uid | 右下级Uid |
| check\_status | 会员确认状态 0 未处理 1 推荐人已审核 2 5级会员已审核 -1 推荐人未审核通过 -2 5级会员未审核通过 |
| statu | 会员状态 0 不可用 1 可用 |
| point | 积分 |

**Member\_superior**: 上层会员

**Member\_relation**:

|  |  |
| --- | --- |
| Puid |  |
| preaches |  |
| Cuid |  |
| Creaches |  |
| check\_status | 会员申请审核状态 |

发展下线用户的时候，要生成一个member\_relation对于关系，同时查找所有Puid的Puid同时生成一份relation关系记录。

**member\_apply**

|  |  |
| --- | --- |
| id |  |
| uid | 申请人id |
| examine\_uid | 审核人id |
| cuid | 申请目标人id，type=1->uid=cuid;type=2->uid是cuid的推荐人，exmaine\_uid是cuid的5级上层会员；type=3->uid=cuid |
| apply\_type | 申请类型 1新会员向推荐人红包通过审核 2 新会用向5级会员红包通过审核 3 升级申请 |
| status | 0 待处理 1 审核通过 -1 审核不通过 |
| creation\_time |  |
| update\_time |  |

**member\_point\_earn** 用户获取积分记录

uid

point 获得积分数量

type 1 称为正式会员

create\_time

update\_time

**member\_point\_consum** 用户消费积分记录

uid

point 消费积分数量

type 1 订单

consum\_id

create\_time

update\_time

网站安装之后 需要把超级管理员的id修改为2，puid设置为-1；同时修改权限组为1，其他的会员在创建的时候需要把权限组设置为2；

后台管理员创建的会员都是合格会员，不需要发红包；

**网站管理员生成1个顶层用户：**

**friend\_member：**

recommended=0；contacted=0；reaches=0；level=9；bdirection=0；check\_status=2；statu=1

**friend\_member\_superior:**

first\_superior -> ninth\_superior = 顶层用户;

需要更新用户所属组

推荐会员：

**friend\_member:**

recommended=推荐会员ID；contacted=接点会员ID；reaches=接点会员+1；level=0；bdirection=选择方向；check\_status=0；statu=1；

同时更新结点人的 x\_left\_uid

**friend\_member\_superior:**

根据自己的reaches递归生成，

**friend\_member\_relation**：

Puid=接点人ID；preaches=接点人reaches；Cuid=新会员ID；Creaches=新会员reaches；status=新会员check\_status;

同时递归自己的Puid，生成同样的记录

**新会员登陆系统之后：**

查找用户的基本信息 -> user\_info 包括会员基本信息；团队成员统计

select \* from friend\_member 如果发现

check\_status = 0 -> 我要开通正式会员

select \* from member\_apply where uid = xx and type = 1

-> if null 发起申请

-> if not null and status = 0 -> 等待审核

-> if not null and status = -1 -> 审核未通过

check\_status = 1 -> 我要开通正式会员

select \* from member\_apply where uid = xx and type = 2

-> if null 发起申请

-> if not null and status = 0 -> 等待审核

-> if not null and status = -1 -> 审核未通过

只有check\_status = 2 的用户才可以进行后续的操作：

查找用户的待处理开通会员申请 [提醒]

select \* from member\_apply where type in (1,2) and status = 0

查找用户的待处理会员升级申请 [提醒]

select \* from member\_apply where type in (3) and status = 0

查找用户是否需要进行升级

select count(\*) from **friend\_member\_relation where puid =**

处理会员开通申请：

list:select \* from member\_apply where type in (1,2)

处理 status = 0 的 ，

审核通过

type = 1 -> member:check\_status = 1

member\_relation:check\_status = 1

type = 2 -> member:check\_status = 2

member\_relation:check\_status = 2

生成用户积分 更新member:point

审核不通过

type = 1 -> member:check\_status = -1

member\_relation:check\_status = -1

type = 2 -> member:check\_status = -2

member\_relation:check\_status = -2

处理会员升级申请：

list:select \* from member\_apply where type in (3)

处理 status = 0 未审核升级申请

审核通过 ： member：level +1

审核不通过 ： member:level