DBH2-2016M8009073008-袁胜-SQL实验报告3

作业

基本信息	
姓名：	袁胜
专业：	计算机技术
学号：	2016M8009073008
年级：	2016
院系：	工程科学学院
E-mail:	pigmon@163.com

1. 实验目的

实践 MySQL 的事务处理功能。

2. 实验准备

(1) 下列组件是完成本实验所必须的

MySQL 5.7.17 for Linux;
PHP 5 含 MySQL 模块

(2) 实验用表

实验内容采用我工作的实际项目中，新用户注册和删除的过程的事物处理。这两个过程会对多个表产生 INSERT 或 UPDATE 操作，而过程需要多个表之间的逻辑同步，所以采用事务处理。

该项目数据库采用MySQL 5，程序部分采用 PHP 实现。

3. 实验内容和步骤

3.1 事务处理函数

```php
/*
@function: transaction
@desc: 事务处理函数
@param:
    $_arr_query: SQL 语句数组
@ret：错误代码
*/
public static function transaction($_arr_query)
{
    $mysqli = new mysqli(DB_DOMAIN, DB_USER, DB_PSWD, DYNAMIC_DB_NAME);

    if ($mysqli->connect_error) {
        die('Connect Error (' . $mysqli->connect_errno . ') ' .
            $mysqli->connect_error);
    }
    $mysqli->query("SET NAMES utf8");

    $mysqli->autocommit(FALSE);

    $ret = TRUE;

    $count = count($_arr_query);
    foreach ($_arr_query as $query) {
        $result = $mysqli->query($query);
        $ret = ($ret && $result);
    }

    $mysqli->commit();
    $mysqli->close();

    return $ret;
}
```

3.2 对事物处理函数的调用

3.2.1 新建用户

```php
public static function new_player($_type, &$_uid)
{
    $leader = "0";
    $box1 = "0";
    $team = "0";

    // fetch uid
    $uid = PlayerMngr::fetch_uid();

    PlayerMngr::get_init_data($uid, $_type, $leader, $box1, $team);


    // store db tables
    $db_ok = false;
    $arr_sql = array();
    $arr_sql[] = "INSERT INTO basic (uid,name,exp,coin,gold,stamina,chara_cnt,leader) VALUES ({$uid},'无名氏',10,100,5,20,4,'{_leader}')";
    $arr_sql[] = "INSERT INTO box1 (uid, g1) VALUES ({$uid}, '{$box1}')";
    $arr_sql[] = "INSERT INTO box2 (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO box3 (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO box4 (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO friend (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO apply (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO recept (uid) VALUES ({$uid})";
    $arr_sql[] = "INSERT INTO team (uid, team1) VALUES ({$uid}, '{$team}')";
    $arr_sql[] = "INSERT INTO progress (uid,story,sp) VALUES ({$uid},'0001000000','0')";
    $db_ok = MySQL_Delegate::transaction($arr_sql);

    // login and init mc
    if ($db_ok)
    {
        $_uid = $uid;
        $ret = PlayerLoader::load($uid);

        return $ret;
    }
    else
```

```
        {
            return ERR_DB;
        }
}
```

3.2.2 删除用户

```
public static function del_player($uid)
{
    $arr_sql[] = "DELETE FROM basic WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM box1 WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM box2 WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM box3 WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM box4 WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM friend WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM apply WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM recept WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM team WHERE uid={$uid}";
    $arr_sql[] = "DELETE FROM progress WHERE uid={$uid}";

    $db_ok = MySQL_Delegate::transaction($arr_sql);

    return ($db_ok) ? NO_ERR : ERR_DB;
}
```

4. 实验结论

> - MySQL 5 的 InnoDB 模式已经支持事务处理了。
> - 事务处理可以帮助开发人员管理原子操作的过程。