类名：DataBase

初始化参数：无

常用函数：

function conn(); //链接$IVPConfig配置的数据库

function connect(&$CONFIG); //链接$CONFIG配置的数据库

function selectDb($dbname); //选择数据库

function close(); //关闭链接

function logDef(&$log); //定义Loghelper的实例

function sql($query); //执行$sql语句

static function rows($result); //得到查询结果的总数

static function getRow($result); //得到查询结果的一条记录

static function getAffectedRow(); //得到前一次操作影响的记录行数

static function getAutoIncrement($from); //查询表的自动递增的值是多少

function getLastInsertId(); //返回上一次insert操作产生的ID

function countRow($from,$where); //查询$sql语句查询总数

static function safe($str); //sql语句防注入

function cols($result); //查询结果的字段数

function getCol($result,$i); //取出查询结果的指定字段的名称

function getLastError(); //得到上一次的错误原因

类成员函数:

public:

static function protect(str); //sql语句防注入

function error($errdef); //记录错误

function conn(); //链接$IVPConfig配置的数据库

function connect(&$CONFIG); //链接$CONFIG配置的数据库

function selectDb($dbname); //选择数据库

function close(); //关闭链接

function logDef(&$log); //定义Loghelper的实例

function sql($query); //执行$sql语句

static function rows($result); //得到查询结果的总数

static function getRow($result); //得到查询结果的一条记录

static function getAffectedRow(); //得到前一次操作影响的记录行数

static function getAutoIncrement($from); //查询表的自动递增的值是多少

function getLastInsertId(); //返回上一次insert操作产生的ID

function countRow($from,$where); //查询$sql语句查询总数

static function safe($str); //sql语句放注入

function cols($result); //查询结果的字段数

function getCol($result,$i); //取出查询结果的指定字段的名称

function getLastError(); //得到上一次的错误原因

private:

none

类成员函数实例：

这是一个DataBase的类，封装了一些基本的数据库的操作。

函数：function conn();

参数：none

返回值：成功返回true，失败返回false

作用：链接$IVPConfig配置的数据库

注意：$IVPConfig [‘database’]数组里应该包含$IVPConfig[‘database’][‘host’],$ $IVPConfig [‘database’][‘dbuser’], $ $IVPConfig [‘database’][‘dbpassword’],

$ $IVPConfig [‘database’][‘dbname’]分别代表数据库所在主机的IP地址，用户名，密码，所要操作的数据库的名称. 如果有错误，错误会记录到error中.

样例代码：

<?php

$db=new DataBase();

$db->conn();

$db->close();

?>

结果：

无

函数：function connect(&$CONFIG);

参数：$CONFIG(一些配置参数)

返回值：成功返回true，失败返回false

作用：链接$CONFIG配置的数据库

注意：$CONFIG[‘database’]数组里应该包含$CONFIG[‘database’][‘host’],$CONFIG[‘database’][‘dbuser’], $CONFIG[‘database’][‘dbpassword’],

$CONFIG[‘database’][‘dbname’]分别代表数据库所在主机的IP地址，用户名，密码，所要操作的数据库的名称.如果有错误，错误会记录到error中.

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFiG);

db->close();

?>

结果：

无

函数：function selectDb($dbname);

参数：$dbname(数据库名称)

返回值：成功为true，失败为false

作用：更换要进行sql语句的数据库

注意：如果sql语句只是临时使用的话，请使用完后，再次调用selectDb($olddbname/\*这个由你配置的数据库的名称\*/)

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’)

$db->connect($CONFIG);

$db->selectDb(‘world’);

$db->sql(“SELECT \* FROM `t\_message`”);

$db->selectDb($CONFIG[‘database’][‘dbname’]);

db->close();

?>

结果：

无

函数：function close();

参数：none

返回值：none

作用：关闭数据库的链接

注意：none

样例代码：

<?php

$db=new DataBase();

$db->conn();

$db->close();

?>

结果：

无

函数：function logDef(&$log);

参数：$log(Loghelper的实例)

返回值：none

作用：定义Loghelper的实例

注意：有关Loghelper的类的详细信息，请查看Loghelper的类说明

样例代码：

<?

$log=new Loghelper();

$db=new DataBase();

$db->conn();

$db->logDef($log);

$db->close();

?>

结果：

无

函数：function sql($query);

参数：$query(sql的语句)

返回值：返回mysql\_query($query)的返回值

作用：执行$query语句的请求

注意：执行sql语句后，如果之前有调用logDef($log)的函数.那么所执行的操作将会被记录下来。

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$db->sql(“SELECT \* FROM `t\_message`”);

db->close();

?>

结果：

无

函数：static function rows($result);

参数：$result(调用类sql函数所产生的返回值)

返回值：资源的条目数

作用：查询资源的条目数

注意：无

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“SELECT \* FROM `t\_message`”);

$db->rows($result);

db->close();

?>

结果：

无

函数：static function getRow($result);

参数：$result(调用类sql函数所产生的返回值)

返回值：返回查询的资源的结果集中取一行作为关系数组，如果没有更多行，则返回false

作用：提取查询结果

注意：利用结果集中没有更多行返回false，可以将所有的结果取出。

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“SELECT \* FROM `t\_message`”);

While($re=$db->getRow($result)){

print\_r($re);

}

db->close();

?>

结果：

无

函数：static function getAffectedRow();

参数：none

返回值：执行成功，则返回受影响的行的数目，如果最近一次查询失败的话，函数返回 -1。

作用：返回前一次sql语句操作所影响的记录行数

注意：none

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“SELECT \* FROM `t\_message`”);

$nums=$db->getAffectedRow();

db->close();

?>

结果：

无

函数：static function getAutoIncrement($from);

参数：$from(所要查询的表名)

返回值：当前表名的自动递增的值的数值，失败返回0

作用：得到表名的自动递增的值的数值

注意：none

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$nums=$db-> getAutoIncrement(‘t\_message’);

db->close();

?>

结果：

无

函数：function getLastInsertId();

参数：无

返回值：成功返回id值，失败返回0

作用：返回上一次Insert操作产生的Id

注意：如果需要保存该值以后使用，要确保在insert的操作之后立即调用 getLastInsertId()。

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“INSERT INTO `mesage` VALUES(‘’,’hello’,time())”);

$nums=$db->getLastInsertId ();

db->close();

?>

结果：

无

函数：function countRow($from,$where);

参数：$from(数据库的名称),$where(查询条件)

返回值：查询失败返回-1，成功返回查询的条目数

作用：得到数据库中满足$where条件的数据的数目

注意：$from是数据库的名称,$where是WHERE+查询的条件

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$from=”`message`,`board`”;

$where=”WHERE message.id=board.id AND borad.id=1”;

$nums=$db->countRow($from,$where);

db->close();

?>

结果：

无

函数：static function safe($str);

参数：$str(字符串)

返回值：防注入处理过的字符串

作用：防sql注入的函数

注意：最好在对组成sql语句的变量使用这个函数。对于变量最好只使用一次该函数。

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$from=”`message`,`board`”);

$id=1;

$where=”WHERE message.id=board.id AND borad.id=’”.$this->safe($id).”’”;

$nums=$db->countRow($from,$where);

db->close();

?>

结果：

无

函数：function cols($result);

参数：$result(sql函数返回的结果集)

返回值：返回查询结果的字段数

作用：效果形同mysql\_num\_fields()

注意：无

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“SELECT \* FROM `message` WHERE `id`=1”);

$nums=$db->cols ($result);

db->close();

?>

结果：

无

函数：function getCol($result,$i);

参数：$result(sql函数返回的结果集),$i(整数数值)

返回值：指定字段的字段名

作用：得到查询结果字段中指定字段的字段名

注意：无

样例代码：

<?php

$db=new DataBase();

$CONFIG[‘database’]=array(‘host’=>’127.0.0.1’,’dbuser’=>’host’,’dbpassword’=>’123’,’dbname’=>’hello’);

$db->connect($CONFIG);

$result=$db->sql(“SELECT \* FROM `message` WHERE `id`=1”);

$nums=$db->getCol ($result,1);

db->close();

?>

结果：

无

函数：function getLastError();

参数：none

返回值：没有错误为no error,否则返回错误的具体值

作用：得到上一次错误的具体值

注意：无

样例代码：

无

结果：

无