**MySQLEncapsulation类说明文档**

1. 类函数
2. void init(HashMap<String,String> config)

config字典应该有四个key

key value

dbhost 服务器名

dbname 数据库名

username 用户名

password 密码

1. MySQLEncapsulation getInstance()

返回Singleton的对象，如果没有init()，则按照默认参数初始化（当咱们确定好数据库名表名用户名密码都一样，就可以不用init了）

返回值是一个这个类的对象。

1. 对象函数
2. boolean insert(String table,HashMap<String,String> data)

插入函数，传入字典

Key是字段名如name

Value是内容如GongYanSong

返回值是成功与否。

1. boolean update(String table , String[] where , HashMap<String,String> data)

更新函数 ，第一个参数是表名，第二个参数是where语句，第三个参数是更新内容

Where参数大概是这样的

String[] where = {

“name = ‘Gong’”,

“ID” > ‘11000’,

“dept\_name like ‘%Soft%’”

};

第三个参数是set语句中的那些key,value也是表示字段名和更新内容。

返回值是成功与否。

1. boolean update\_safe(String table , String[] where , HashMap<String,String> data)

安全更新，与上面唯一不同是，当确定只更新一条特定内容时，用这个安全一点，因为如果没有传入where参数，一不小心传入空值，不会全部都更新。

1. boolean delete(String table ,String[] where)

删除内容，第一个参数是表名，第二个参数同update

返回值是成功与否。

1. boolean delete\_safe(String table ,String[] where)

安全删除，同安全更新

1. ResultSet select(String[] fields, String[] tables, HashMap<String,String[]> param)

哇这个就麻烦了

第一个参数是select 后面的那个 一般是\*的那个，可以传入很多字段名，但是必须都是以String数组

\*也必须放在数组里如String[] fields={“\*”}；

第二个参数是表名，也是放在数组里

第三个参数是很麻烦的东西，是一个字典，里面可以放where,order by,group by,limit四种东西，分别都是用字符串数组来盛放。

举一个例子，如果按照两个东西排序

String[] order ={

“dept\_name”,”name”

} ;

HashMap<String,String[]> param.put(“order”,order);

这个字典就可以当作参数传进去了。

字典Key可以有四种字符串“where”,”order”,”group”,”limit” Value值就是对应的String[], 注意，Limit中的数字也按字符串传入进去。另外group的Value值数组，第一个应该放gourp by的东西，第二个应该放having后面的条件。

返回值是ResultSet

1. int count(String field, String table, String[] where)

查找某个关系的数量，第一个参数是字段名（默认为Distinct），第二个参数是表名，第三个参数同update和delete中的String[]

返回值是int

1. 事务处理：

当需要事务处理时

先使用void beginTransection()

然后可以使用增删查改的函数

根据条件使用void commit()

提交

或者void rollback()

撤销

1. boolean excute(String sql)

执行一句SQL语句，返回成功与否。