# 测试平台数据库文档

#### 数据库选用：sqlite3

sqlite3为小型关系型数据库，免去了繁琐的安装配置环节，并且相比主流数据库mysql和oracle，它的处理速度都要高过它们；sqlite属于嵌入型数据库，相当于程序一部分，所以不能作为分布式数据库管理，但提供标准ODBC连接和API，支持多种语言代码内调用；

#### 创建的数据库名：webFrame.db

路径地址放在web项目下，/sql/webFrame.db

#### 表结构：

表名： USER

用户表，储存用户基本信息

字段： ID INT 主键，用户唯一标示

USERNAME TEXT 登录用户名，唯一且不为空

PASSWORD TEXT 登录密码，不为空

ORGANIZE TEXT 用户所属权限组，无约束

REALNAME TEXT 用户真实姓名，无约束

表名： ROLE

角色权限表，储存权限信息

字段： ORGANIZE TEXT 主键，权限组唯一标示

ORNAME TEXT 权限组项目名，唯一且不为空

表名： REPORT\_AUTO

自动化报告表，自动化测试报告的基本信息

字段： RPONAME TEXT 主键，报告的文件名

RPOPATH TEXT 报告的储存地址，唯一

RPOPASSRATE TEXT 该报告的通过率，无约束

PROCODE TEXT 报告所属项目，不为空

RUNNER TEXT 生成该报告的执行人，无约束

FINISHED TEXT 是否完成，0:正在执行(默认) 1:完成 2:失败;

SCHEDULE TEXT 报告执行进度，0:(默认)

通常为0到100;

表名： PROJECT

项目表，储存项目信息

字段： PROCODE TEXT 主键，项目的唯一标示

PRONAME TEXT 项目的名字，唯一且不为空

SCRIPTS\_SUM TEXT 该项目脚本总数，无约束

#### 数据库调用方法：

模块路径： 在项目目录下，/lib/sqlLib/sqlConnect.py

讲解：

类： SqlConnect( database )

--database:数据库地址（测试平台项目中固定为'sql/webFrame.db'）

方法1： excute( sql )

--sql:需执行的sql语句

--查询的操作（不含提交操作）调用此方法

--返回sql语句查询出的所有数据

用法举例：

--result = SqlConnect("sql/webFrame.db").excute("select \* from user")

--print result

方法2： commit\_excute( sql )

--sql:需执行的sql语句

--修改、新增的操作（需要提交的操作）调用此方法

--没有返回

用法举例：

-- SqlConnect("sql/webFrame.db").commit\_excute("insert into user values ('1','songruijie','123456','admin')")