**实验6：SQL输入验证清理实验报告**

姓名：江继军 学号：21373222 班级：213921

实验日期：2024年5月15日 实验地点：一号馆406 成绩：

1. **实验目的**

1.熟悉SQL注入攻击的基本防范措施，掌握SQL输入验证清理的基本方法。

2.基于MySQL数据库、Python pymysql和re库，实现用正则表达式验证清理输入的SQL语句，防范SQL注入攻击。（本次报告页数限制在6页内）

**二、实验内容**

1. 环境搭建

1. 数据库环境：基于实验五安装的数据库和创建的表。
2. 安装Python pymysql库，用于在Python程序中连接MySQL数据库。

2. 用正则表达式验证清理输入的SQL语句

1. 编写验证清理函数，构建特殊字符字典并查找输入语句中的特殊字符，可清理或抛出异常。
2. 编写数据库连接和SQL语句执行函数，在终端输出执行结果。
3. 输入多条SQL语句，验证正常SQL语句的执行，以及恶意SQL语句的清理或恶意提示。

**三、实验环境**

分项列出主要实验环境（主机/操作系统、Python开发环境、Python版本、第三方库版本等）：

1. 操作系统：Win11
2. Python开发环境：pycharm-community
3. Python版本：3.9
4. pymysql版本：1.1.0
5. re版本：2.2.1

**四、实验步骤**

写出环境搭建、编写验证清理函数、数据库连接、SQL语句执行等步骤（采用1.2.3.4的序号方式列出详细步骤；写明所编写的代码；贴出效果示意图，例如正常SQL语句执行、SQL注入攻击失败结果截图等）。

1. 环境搭建：

数据表为“students”，验证可以在Python程序中成功连接数据库（写明所编写的代码，给出连接成功截图）。

创建database\_connect\_test函数，测试是否可以连接到数据库，如果连接数据库失败则不必进行下面SQL语句操作。



图 1 测试数据库连接函数

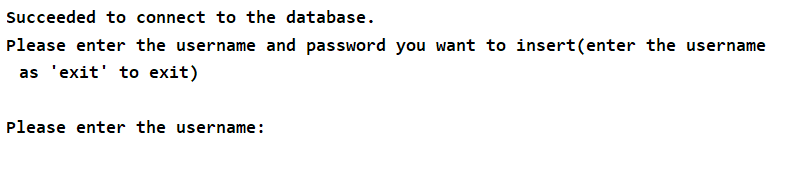


图 2 连接成功截图

1. 编写SQL语句验证清理函数：

构建特殊字符字典（过滤表），编写验证清理函数，通过正则匹配查找输入语句中的特殊字符，可清理或抛出异常（写明所编写的代码，给出清理验证结果截图）。

创建的过滤表、验证清理函数以及结果如下：



图 3 过滤表内容

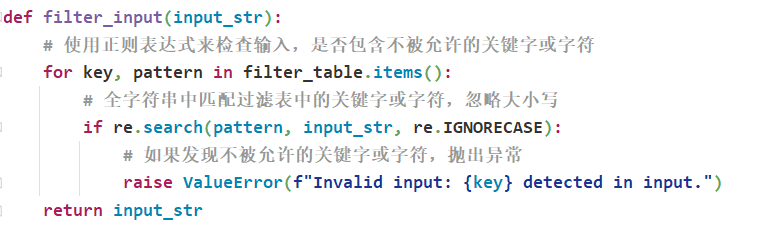


图 4 过滤函数

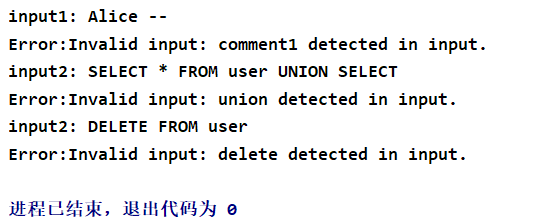


图 5 清理验证结果截图

1. 编写数据库连接和SQL语句执行函数：

连接数据库，构建SQL执行语句，实现将数据（例如：新的用户名和密码）插入到表中，将SQL语句发送到数据库服务器执行，将执行结果永久保存到数据库中（写明所编写的代码，给出SQL语句执行成功的结果截图）。



图 6 SQL语句执行函数



图 7 SQL语句执行函数输入

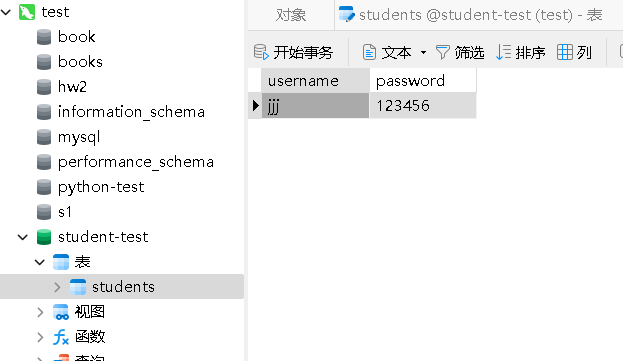


图 8 SQL插入成功验证

1. 用正则表达式验证清理输入的SQL语句：

基于前两步的代码，编写主函数，实现通过正则表达式清理用户输入的SQL语句，将没有异常字符的插入语句正常执行，将存在异常字符的SQL执行语句不予执行并进行提示，请自行输入多条SQL语句验证以上功能（写明所编写的代码，给出SQL语句成功执行或验证清理的结果截图）。[自行编写课上未教授的清理验证方法，+3~6分；其他改进项，酌情+1~4分。]

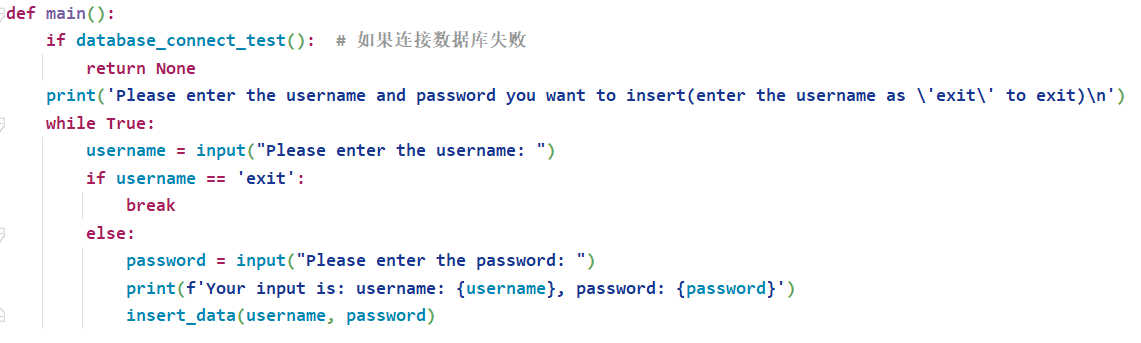


图 9 主函数

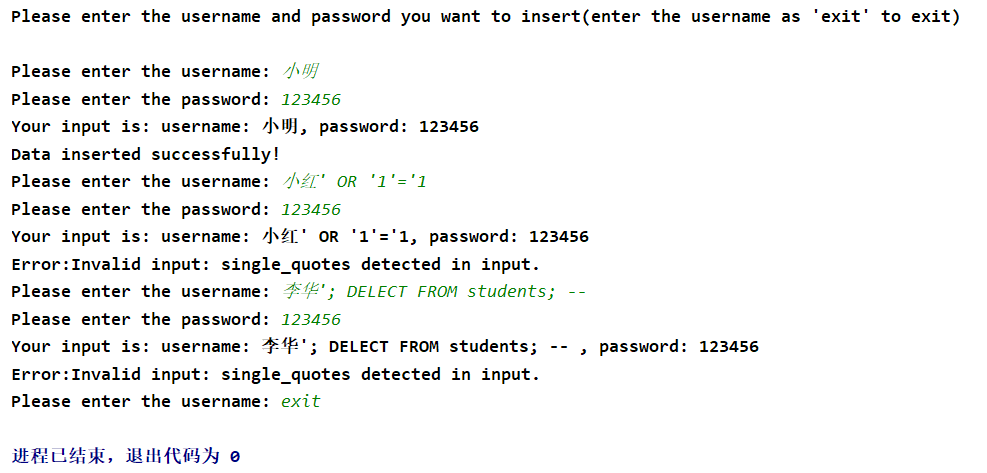


图 10 命令行结果截图

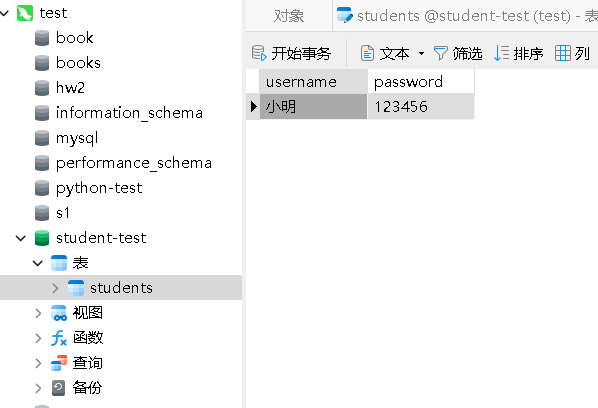
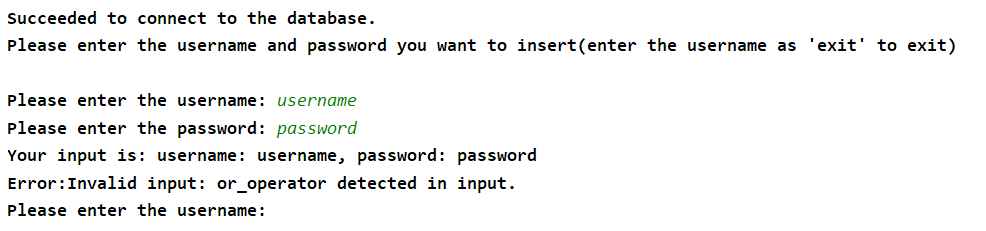


图 11 数据库插入结果截图

**五、问题和解决方法**

1. 当输入的合法内容恰好过滤表中的字符时，会将其当作非法输入阻止。

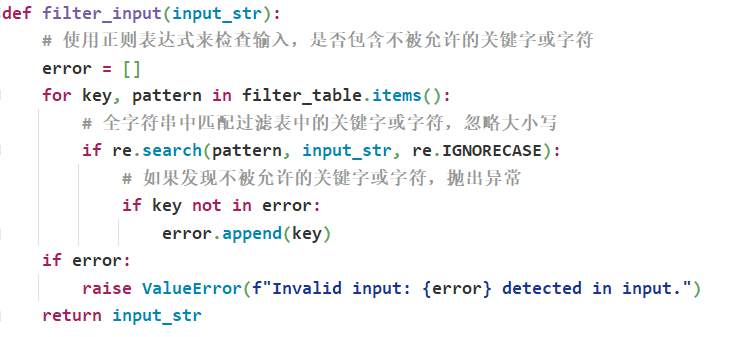
如下，口令输入为“password”，其中正好有过滤词“or”，在检查时会被阻止

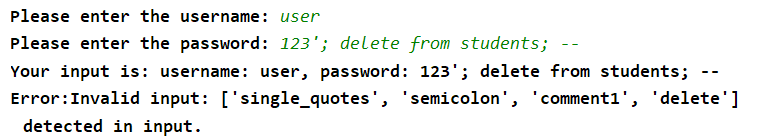


解决方法：使用参数化查询无需编写过滤表，能显著增强SQL安全性。

1. 如图 10 所示，一个SQL注入攻击可能有多种，但检查只给出检测到第一次的异常，基本上都是引号问题。

解决方法：在过滤时，遇到异常将其保存起来，并不直接抛出，等到整个句子检查完毕，再抛出，但这样可能会抛出如序号1所说的问题，会误报。改进代码与结构如下所示：





**六、实验总结**

总结：

通过本次实验，我深刻体会到了SQL注入攻击的危害以及防范此类攻击的重要性。SQL注入是一种常见的网络攻击手段，它通过在SQL语句中插入恶意代码，欺骗数据库执行非法操作，从而获取、篡改或删除数据库中的数据。

在本实验中，通过构建过滤表，编写程序实现了对用户的输入进行验证和清理，可以确保输入数据的格式和内容符合预期，并对恶意特殊字符发出预警，以防止它们被用于注入攻击，从而保障了数据库安全。

**七、附件**

提交可执行的原程序压缩包，随电子版实验报告提交，文件名称为学号+姓名+实验12。