作品编号： **略**

作品类别： ☑软件设计 □硬件制作 □工程实践 □密码应用技术

**2023年第八届全国密码技术竞赛作品设计报告**

作品题目： 基于AES算法的密码管理加密工具

2024年 6月 9日

中国密码学会

|  |
| --- |
| 基本信息表 |
| 编号：略 |
| 作品题目：基于AES算法的密码管理加密工具 |
| 作品类别：☑软件设计 □硬件制作 □工程实践 □密码技术应用 |
| 作品内容摘要：  在日常生活中，我们在许多网址或程序有自己的账户，不少人习惯在多个账户使用相同的，自己常用的用户名和密码，也许曾尝试过使用更加复杂的密码，但是容易忘记密码误事；也想过写个文档来存储自己的密码，又似乎缺乏安全性。  本作品正是通过Python语言编写程序，通过图形界面实现对不同账户的用户名，密码的储存和查看功能。可以通过AES算法实现对信息的加密保存和读取。同时有生成给定长度随机密码的功能，可以使密码更具有安全性。 |
| 关键词（五个）：AES算法；账户管理；安全密码生成；加密保存；分组加密。 |

# 1.作品功能与性能说明

本作品通过Python语言编写程序，通过图形界面实现了以下功能：

1. 新建账户，存储用户名和密码。通过add account按钮，输入用户名和密码，即可存储新的账户，账户信息以字典类型保存，最终通过AES算法加密，保存在password.dat中，若当前位置没有key，则新生成随机的key，存储在当前目录。
2. 更新账户信息。Update account实现的是对已存储的账户信息进行修改。用户先通过下拉菜单选择（而非输入）要修改的账户，再重新输入用户名和密码。
3. 删除账户。Delete account实现的是对已存储的账户信息的删除。用户选择要删除的账户即可。
4. 查看账户。View account实现的是对已存储的账户信息的查阅。用户选择想要查看的账户，即可收到带有用户名密码信息的弹窗。
5. 查看账户列表。List account实现的是列出所有已存储的账户名信息。并不会显示各个账户的密码
6. 生成强密码。Generate password实现的是生成用户给定长度的强密码，并选择一个账户替换原有密码。
7. Quit实现程序的安全退出。

# 2.设计与实现方案

## 2.1 实现原理

首先我们用了许多优秀的库：包括用于创建图形界面的 tkinter，用于加密和解密的 pycryptodome 库以及用于处理字符串和文件操作的标准库。

我们定义了类“passwordmanager”便于后续代码进行。

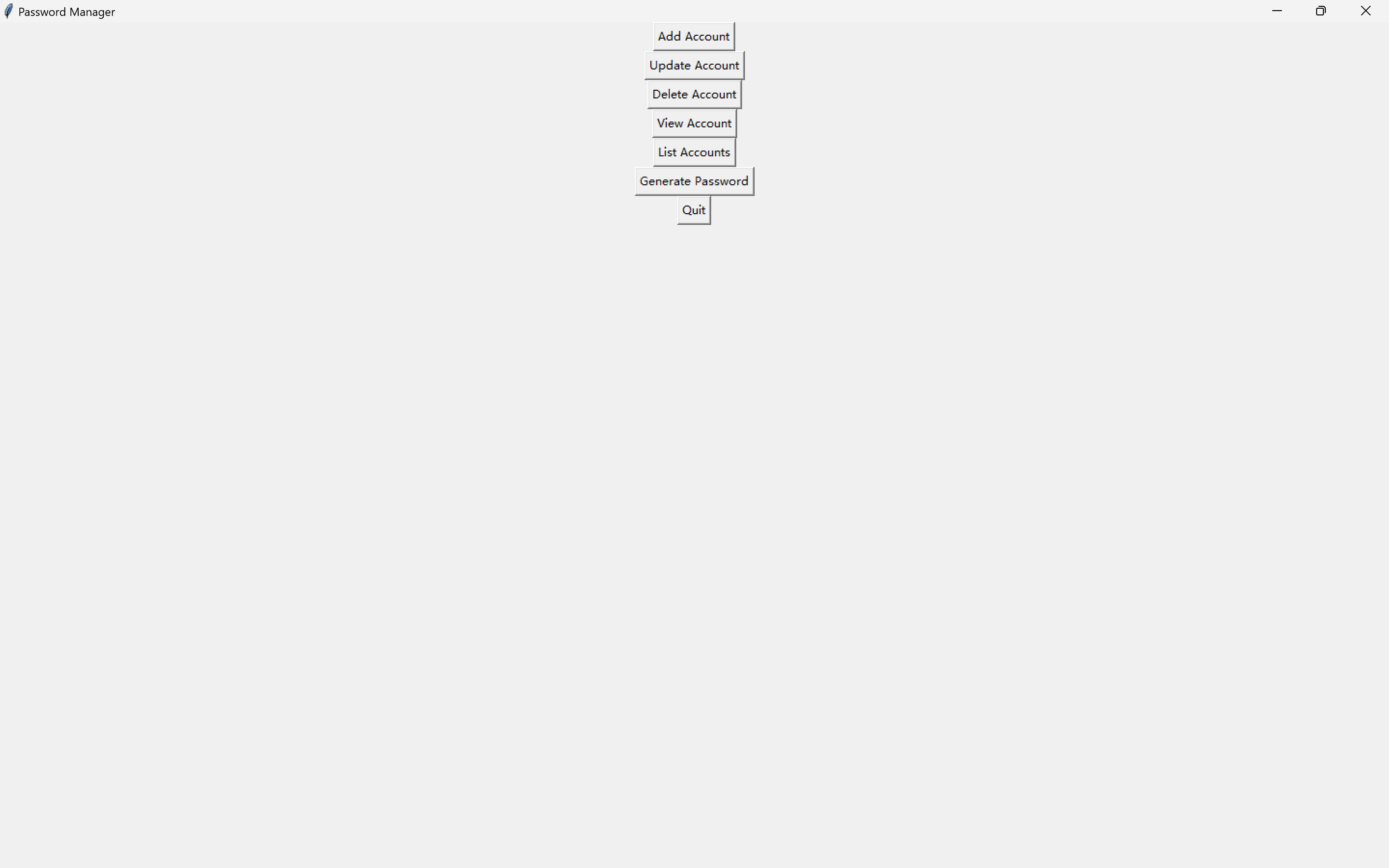
1. 初始化方法（\_\_init\_\_）：设置密钥文件和存储文件的路径，并调用方法加载或创建密钥和存储。
2. 加载或创建密钥（load\_or\_create\_key）：如果密钥文件存在，则读取密钥，否则生成一个新的密钥（32字节的随机密钥序列）并保存。
3. 加载或创建存储（load\_or\_create\_storage）：如果存储文件存在，读取并解密数据，否则初始化一个空字典。同时定义加密解密函数（encrypt,decrypt,save\_storage）
4. 主要功能函数：即

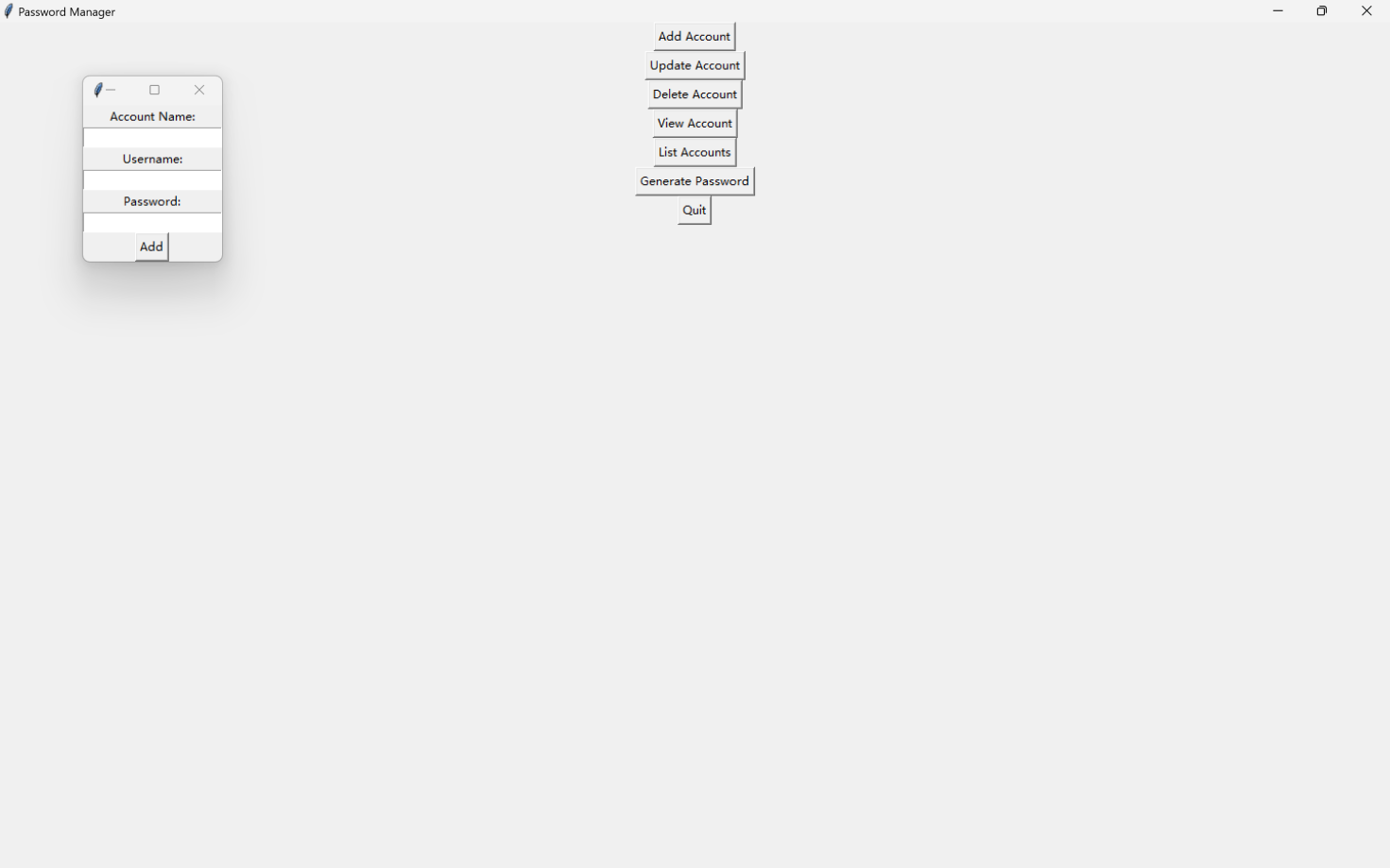
add\_account,get\_account,update\_account,delete\_account,list\_account,generate\_password,quit等函数

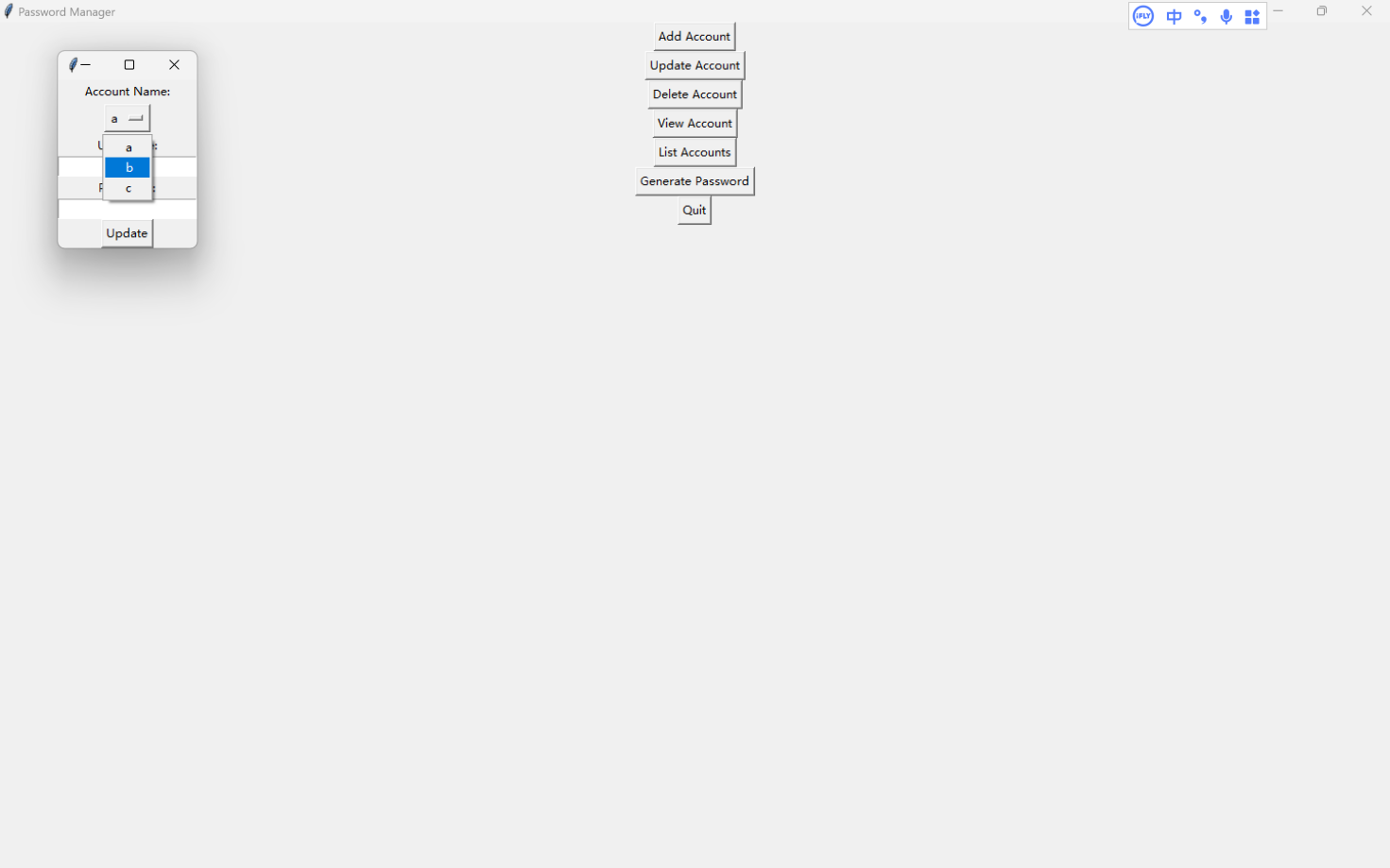
之后定义了passwordmanagerGUI类,用于编写图形界面。分别编写各个按钮的窗口。

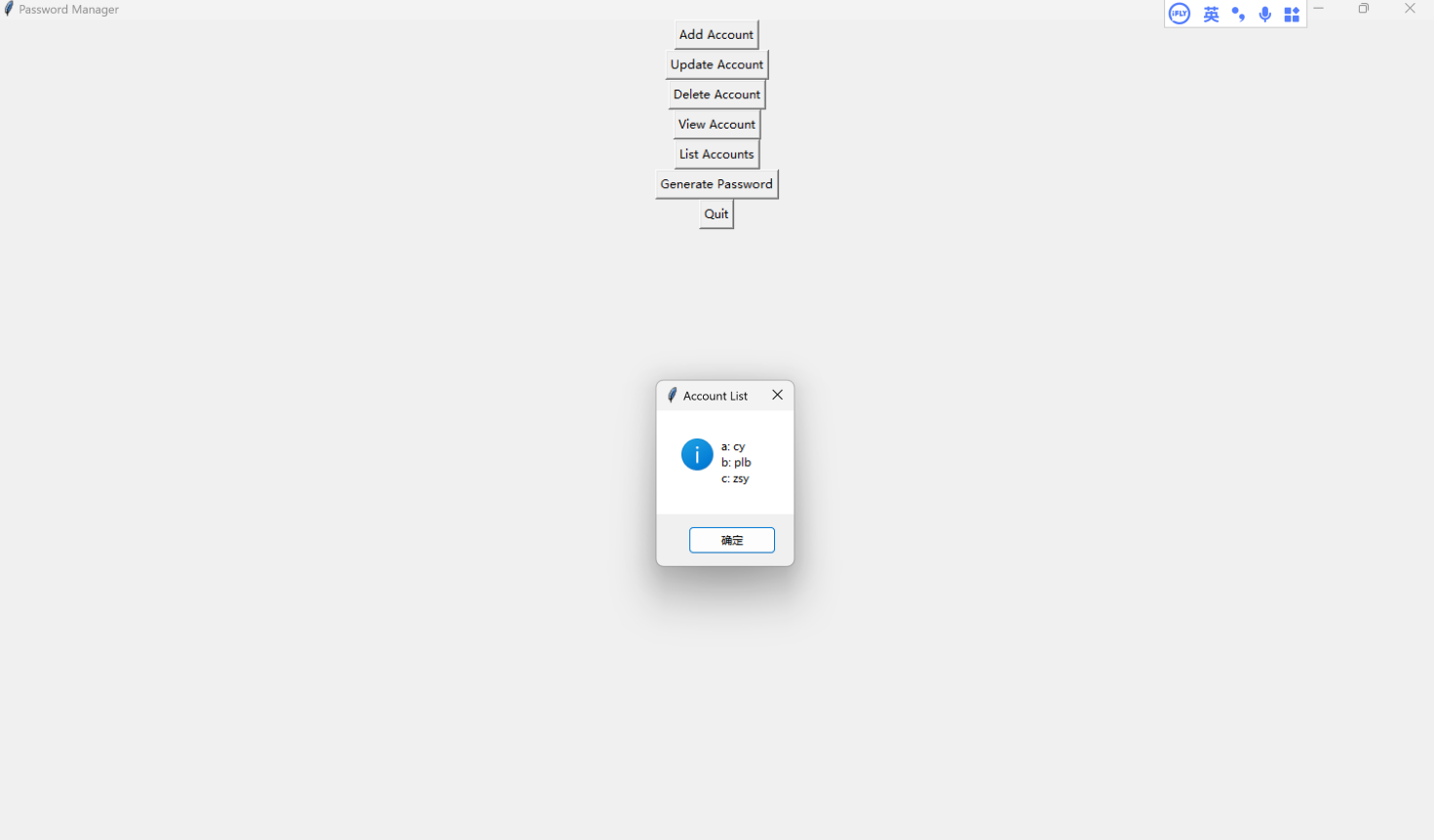
最后为主程序入口，进入主程序启动“PasswordManagerGUI”实例，进入while循环等待用户操作。

## 2.2运行结果









# 3.系统测试与结果

## 3.1测试方案

在一台设备上进行部分账户的存储，以此测试添加，更新，删除，查看，查看列表，生成强密码等主要功能。

测试完毕后，将保存账户信息的password.dat以及密钥key.bin传输到另一个设备。另一个设备测试是否能直接查看第一台设备存储的信息，再通过该程序查看，测试加密程序是否正常运行。

## 3.2 功能测试

1，添加a账户,用户名cy，密码123

2，添加b账户，用户名plb，密码456

3，添加c账户，用户名crypt，密码666

4，查看a账户，更新a账户，用户名改为chen，密码改为789，查看a账户

5，查看列表，删除b账户，查看列表

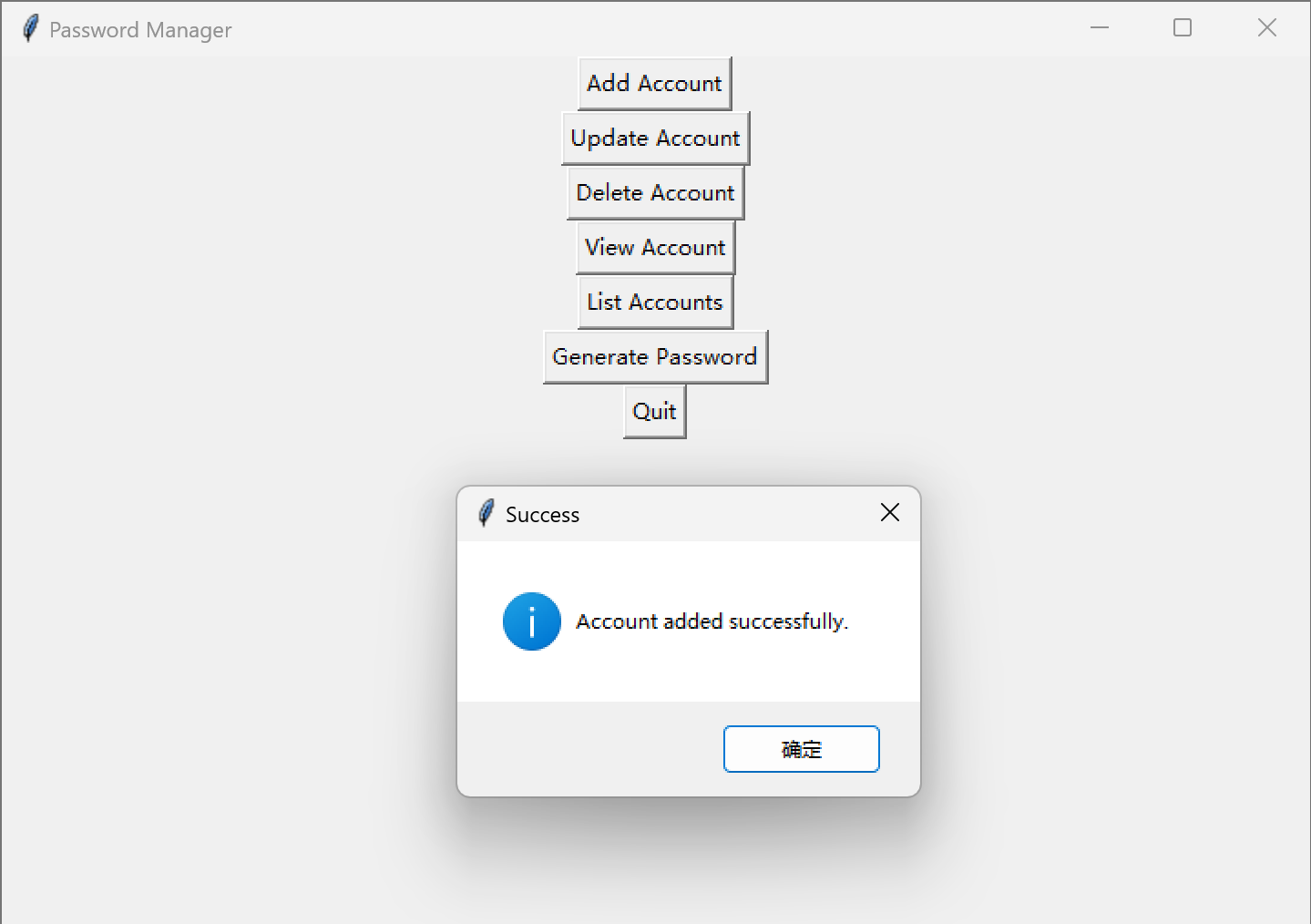
6，查看c账户，生成长度为6的密码到c账户，查看c账户。

7，退出程序。查看文件key.bin及password.dat

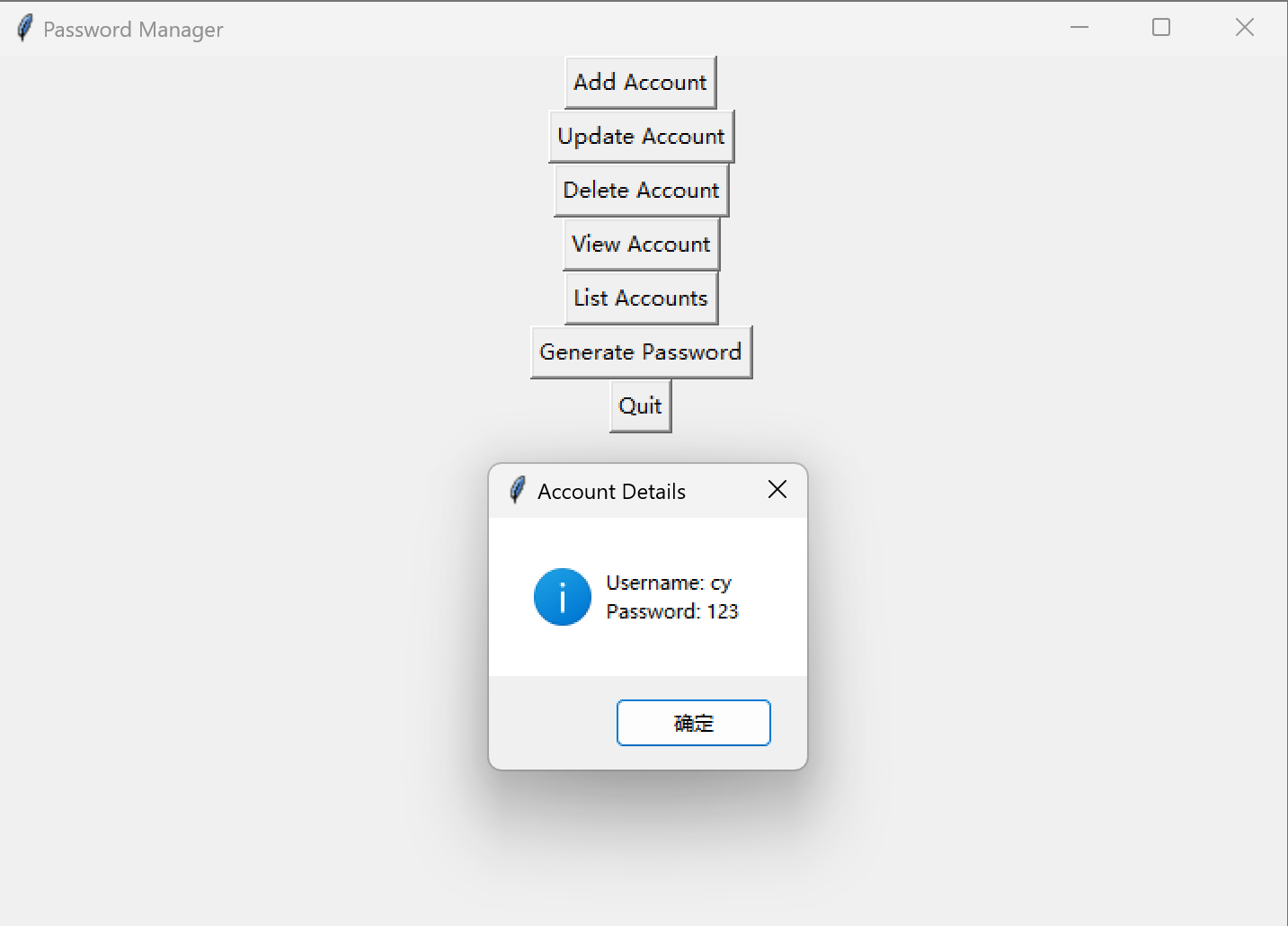
8，另一个设备用相同key.bin及password.dat，查看列表。

## 3.3 测试结果

每一步对应结果：1，2，3



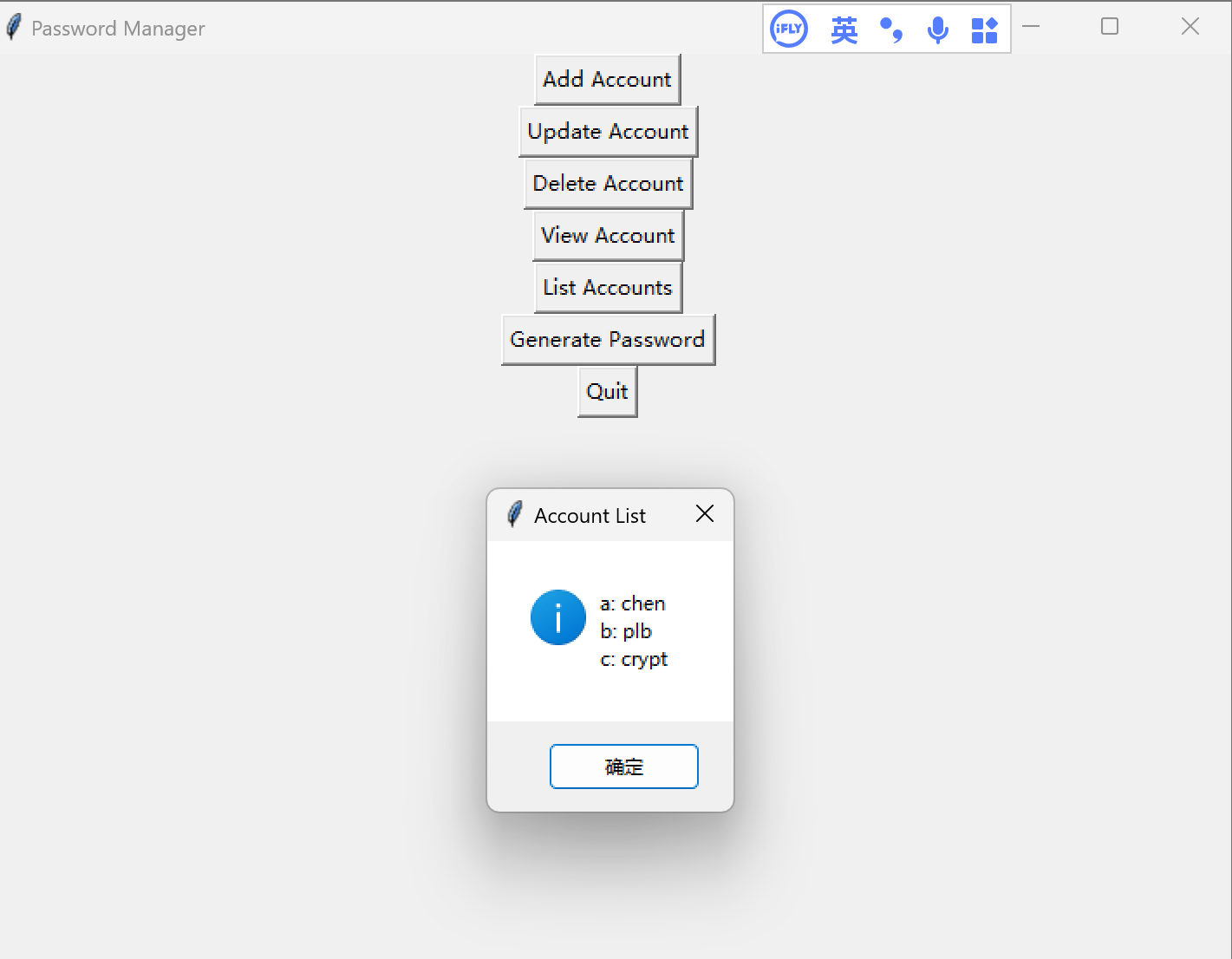
# 4,

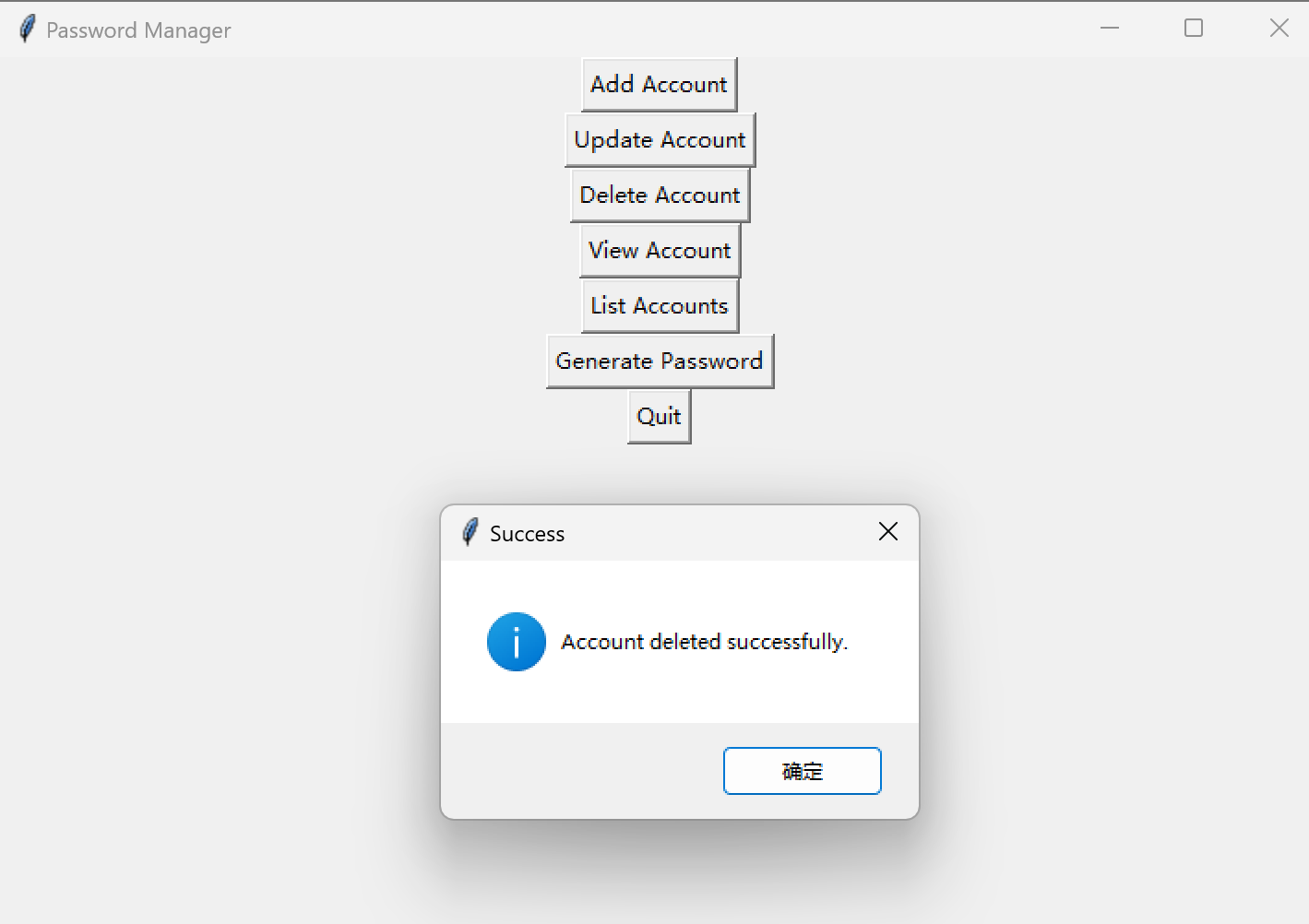


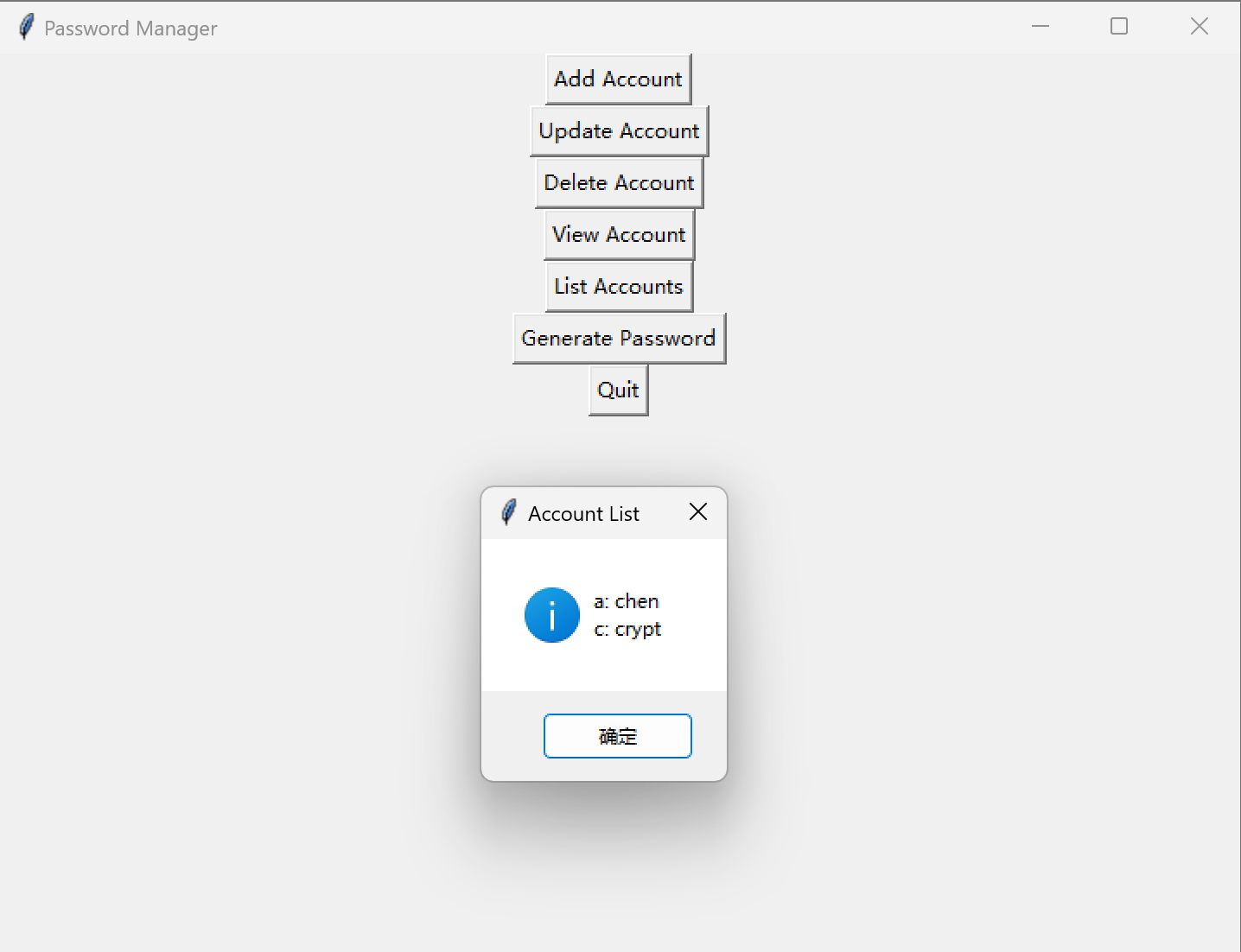
# 

# 

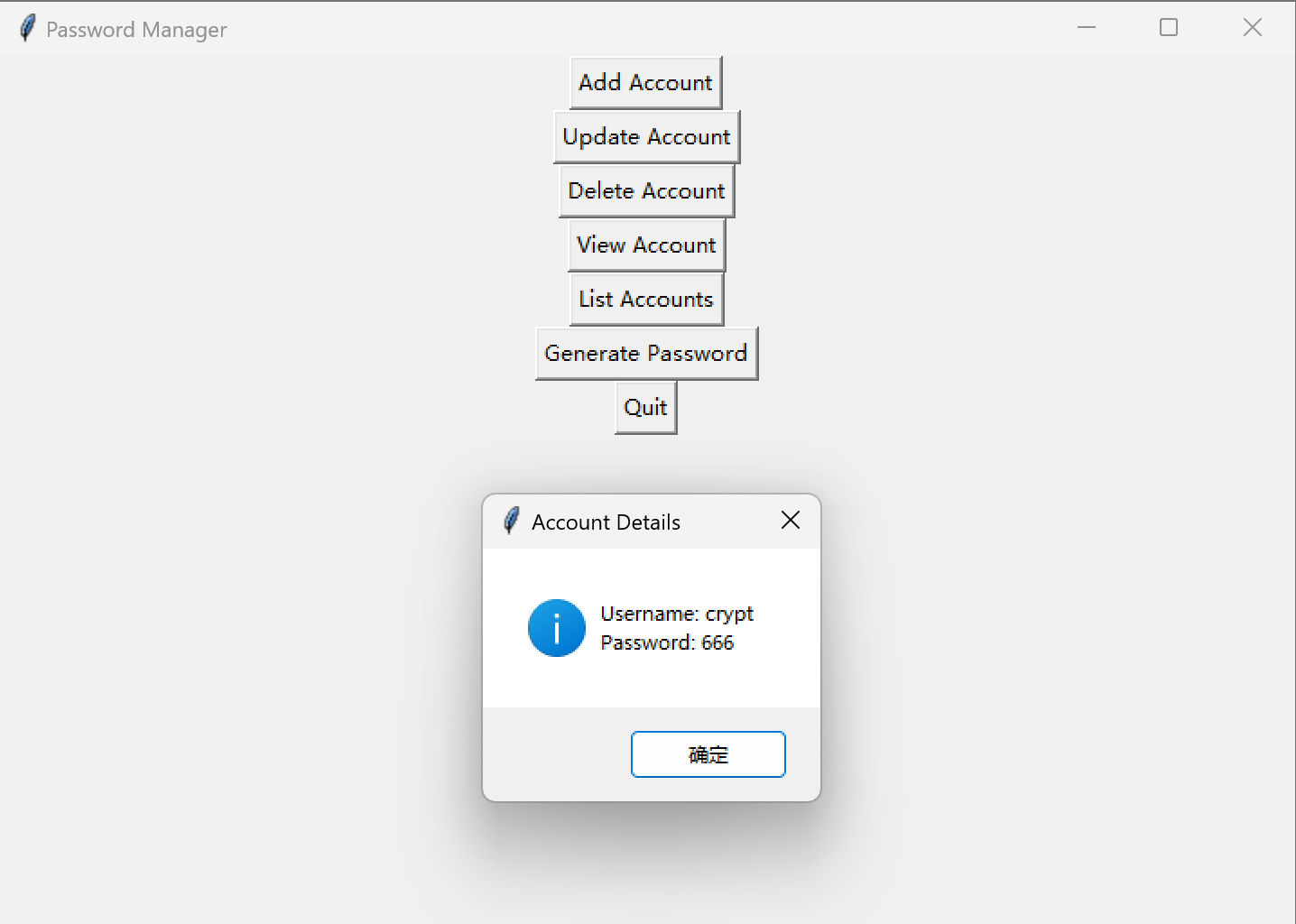
5,

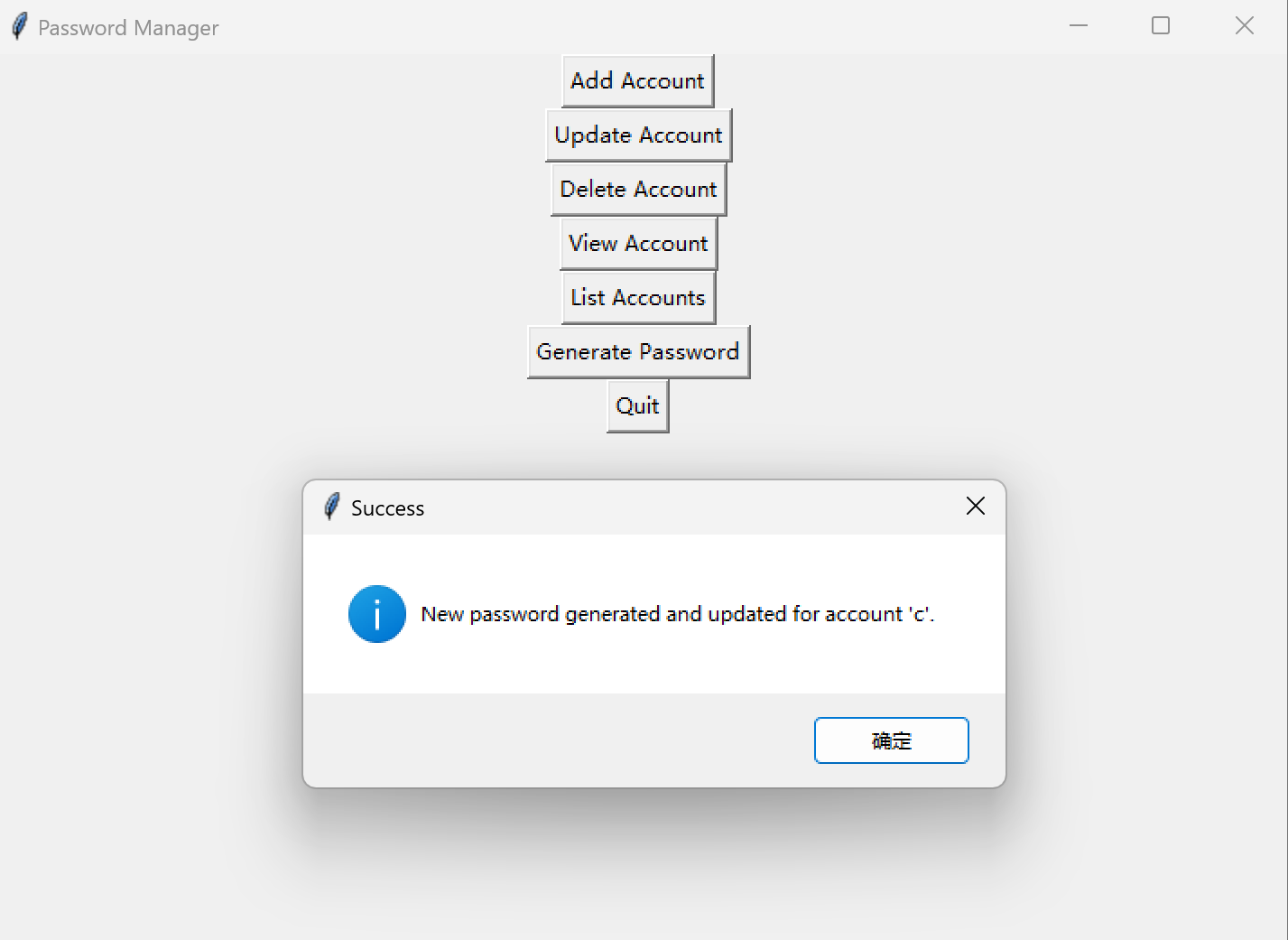


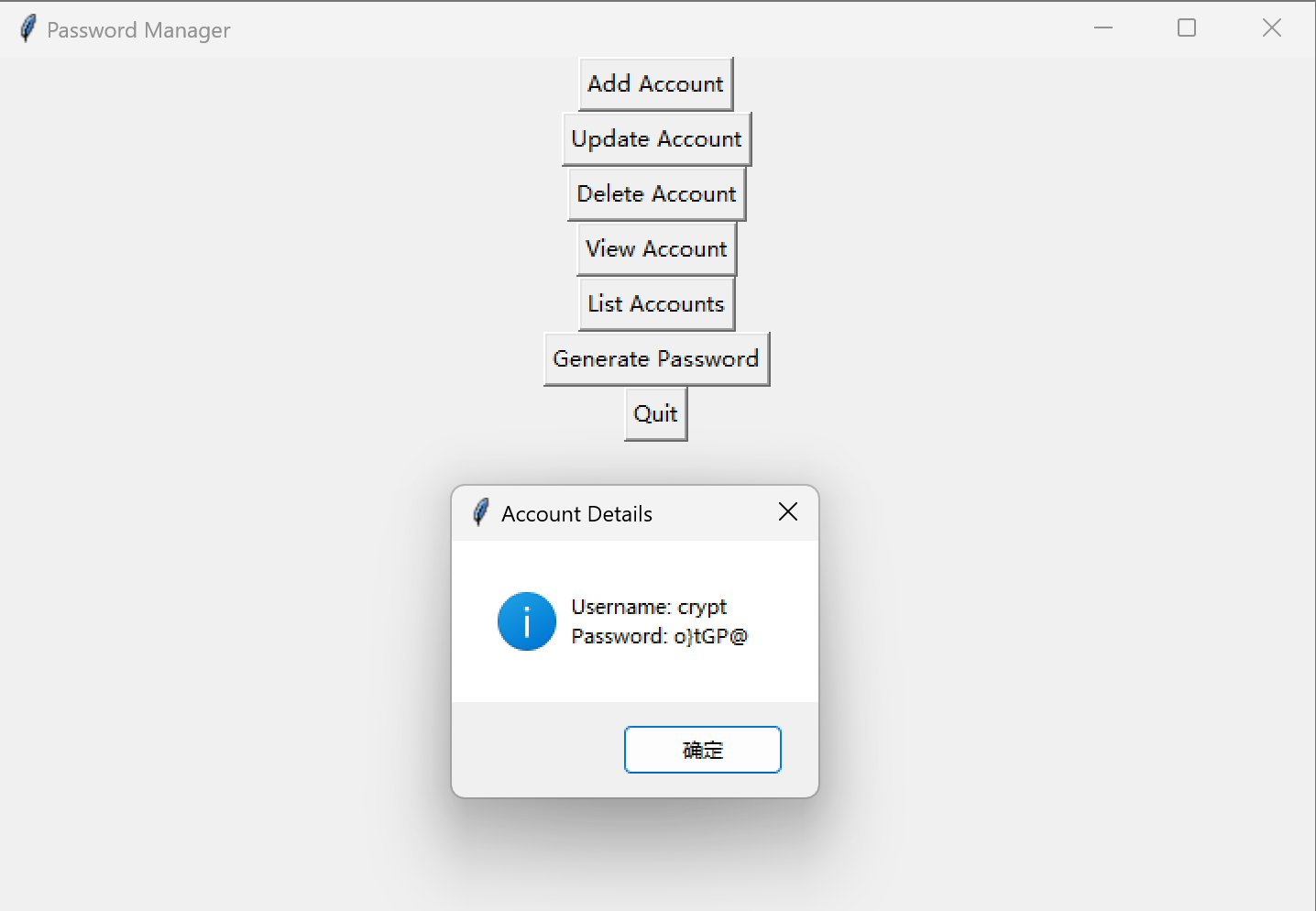




6,







7,正常退出。

# 4.应用前景

**除了基本功能所实现的存储用户输入的账户，该程序进行适当的功能扩展，不难实现用户在浏览网页时，首次使用的新账户的用户名密码的自动保存，当再次需要在该网址使用密码时，自动填充用户选定的密码。**

**此外，相比于目前市面上已经有的浏览器，ios的iCloud对密码的储存，该程序对小规模使用的个体用户来说不需要上网，更加便捷安全。**

# 5. 结论

密码的使用和保存需要严谨对待，日常生活中应当避免使用过于简单，容易猜中的密码，或者在数个不同账户上使用同一用户名及密码。适当利用密码学知识，可以更加安全的保存管理密码，也有机会使用更加随机复杂的密码。