

Contact Book Management System

Application Architecture

Application Menus:

- Add Contacts
- View Contacts
- Remove Contacts
- Search Contacts
- Update Contacts
- Delete All File Data
- Exit()

Folder Directory

1. Directory Name: ContactManagementSystem
2. Directory File Name:
 - a. main.py
 - b. AddContacts.py
 - c. ViewContacts.py
 - d. RemoveContact.py
 - e. SearchContact.py
 - f. SaveFile.py
 - g. UpdateContact.py
 - h. DeleteFileData.py
 - i. RestoredData.py
3. ContactHistory.csv(CSV file format)

Books Contacts Management System

□ main.py

Global Variable

```
saveContact=[]  
FileName="ContactHistory.csv"
```

- AddContacts.py
- ViewContacts.py
- RemoveContact.py
- SearchContact.py
- UpdateContact
- SaveFile.py
- DeleteFileData.py
- RestoredData.py
- ContactHistory.csv

All the External files are connected to the main.py file



main.py

```
while True:  
    Contact Book  Management  
                    System  
        1. Add Contacts  
        2. View Contacts  
        3. Remove Contact  
        4. Search Contacts  
        5. Update Contact  
        6. Delete All Data From File  
        0. Exit  
        option=input()  
    contactManagement(option)
```



```
contactManagement(option):  
    match option:  
        case 1:  
            addContact()  
            saveFile()  
        case 2:  
            viewContacts()  
        case 3:  
            removeContact()  
        case 4:  
            searchcontact()  
        case 5:  
            updatecontact()  
            saveFile()  
        case 6:  
            deleteFileData()  
            saveFile()
```

Books Contacts Management System

□ AddContacts.py

```
def addContact(savedata,filename):  
    lastName=input()  
    firstName=input()  
    dateOfBirth=input()  
    address=input()  
    emailAddress=input()  
    password=input()  
    nationalID=input()  
    contactNumber=input()
```

Note: all the logic (Prevent Duplicate Numbers, Add all data to file, Error Handling, Modular Code, No External Libraries) are implemented in addContact.py file



All the inputs are added to the
SaveContact=[]
FileName="ContactHistory.csv"



Finally, Save all Data in the
"ContactHistory.csv" file

□ ViewContacts.py

```
def viewContacts(fileName):  
  
    with open(fileName,'r') as f:  
        readData=f.read()  
        print(readData)
```

Note: All data has been read
from the **"ContactHistory.csv"**
file

Books Contacts Management System

❑ RemoveContact.py

```
def removeContact(savedata,filename):  
    email=input("Enter Your Email-Address: ")  
    password=input("Enter Your Password: ")
```

Using email, and password , firstly you
are LogIn the system

If Yes

Enter Your Removing
Contact Number.

```
contact=input()
```

If (your contact
number are match to
the file Contact data
then)= program are
executed successfully
and removed data
Successfully

If No

Break the program

Finally

**Update file data
"ContactHistory.csv"**

Books Contacts Management System

☐ SearchContact.py

```
def removeContact(savedata,filename):  
    email=input("Enter Your Email-Address: ")  
    password=input("Enter Your Password: ")
```

Using email, and password , firstly you are LogIn the system

If Yes

Enter Your Searching Contact Number.

contact=input()

If (your contact number match to the file Contact data then)= program are executed successfully and Show all data Successfully

If No

Break the program

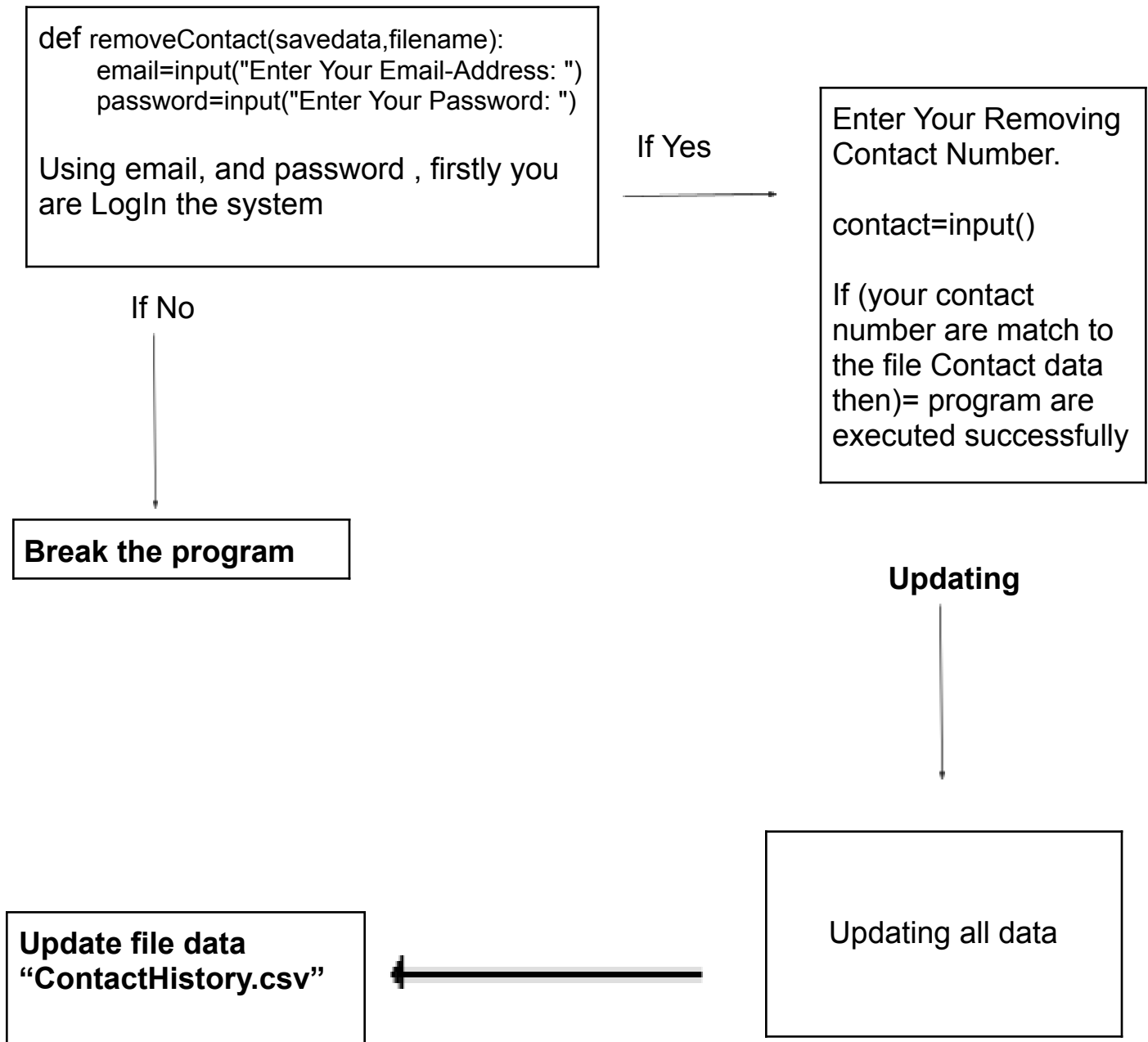
☐ SaveFile.py

```
def saveFile(ContactFileName,savedata):  
    with open(ContactFileName,'w') as fileName:  
        for data in savedata:  
            fileName.write(data[0].....)
```

Note: Save all data successfully in "ContactHistory.csv"

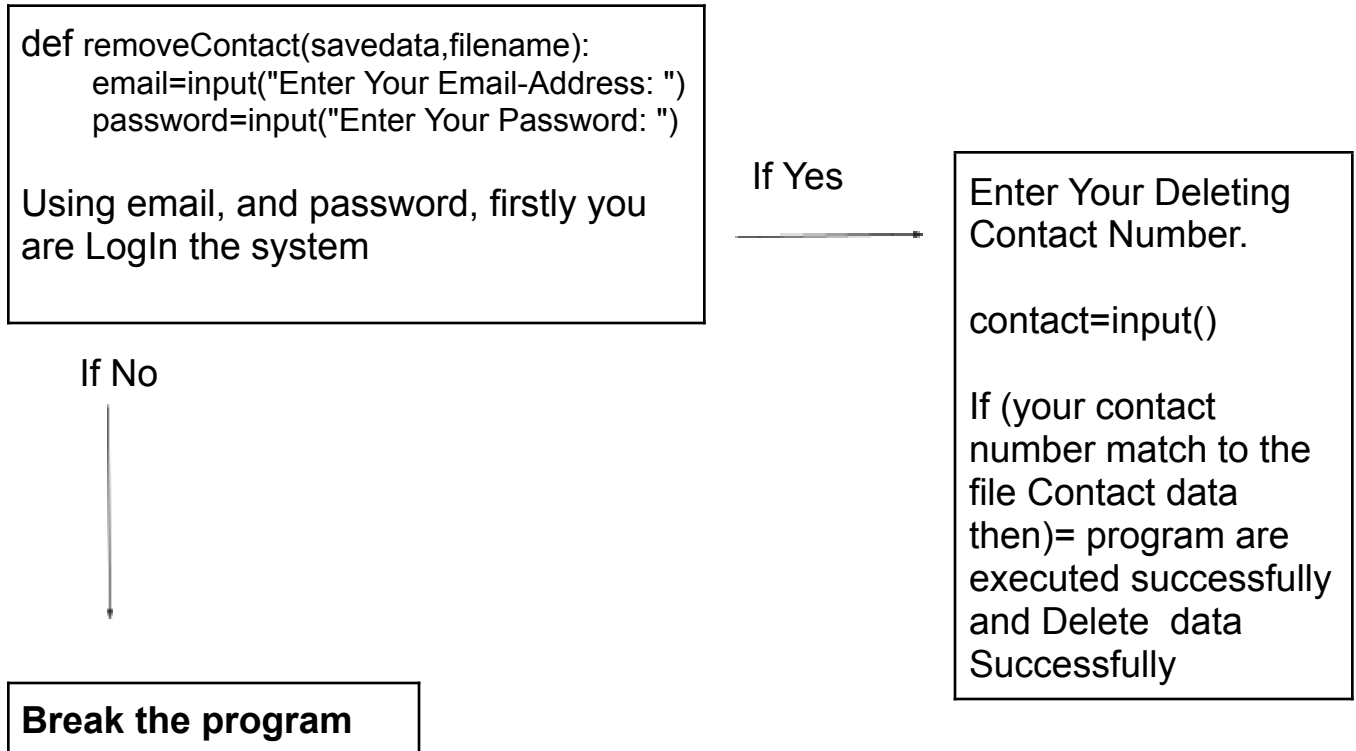
Books Contacts Management System

□ UpdateContact.py



Books Contacts Management System

☐ DeleteFileData.py



☐ RestoredData.py

