

ReadMe - contact book

Outline of Application

Open/launch the application from its installed location

Login to the app if security is enabled. If not skip this and proceed to step 3.

Select preferred action of 1. Create New Contact 2. View Contacts

Create New Contact: To create a new contact, do the following;

Enter all required fields including optional fields if desired.

Read through to verify all information is correct

Click on the “save” button and wait for confirmation.

On confirmation, new contact is saved. On error, please read the error message carefully to resolve.

View Contacts

This option displays all saved contacts on your current screen alphabetically with 3 sub actions to eat record. The sub actions are 1. View Contact 2. Update contact information OR 3. Delete Contact

View Contact: This simply displays the selected contact information to the screen.

Update Contact Information: This option enables the user to edit/update stored/saved information.

Delete: This option does the obvious of deleting the selected contact

Select close/exit button/option to exit application

Algorithm/Pseudocode

Database Design:

Contact Table:

Contact ID

First Name

Middle Name

Last Name

Mobile Number

Work Number

Email

URL

Address

Comments
StatusID
Date Of Creation
Date of Modification
Date of Deletion

Status Table:

Status ID
Status Name
Date Of Creation
Date of Modification
Date of Deletion

Pre loaded data: StatusID = 1, StatusName = "Delete"

Functions

Def CreateFunction(contact):

 FirstName and MobileNumber are required fields. We ensure they are not null.

 If first name is null:

 Show error(first name)

 Endif

 If mobile number is null;

 Show error(mobile number)

 End if

 If required fields were entered correctly, save the data to the database

 Database.save(contact)

 Return ContactID

End function

Def ReadContactsFunction():

 Get all contact data from database and return.

 Return database.read(contact)

End function

```
Def ReadContactFunction(ContactID):  
    Get all contact data from database and return.  
    Return database.read(contact)  
End function
```

```
Def UpdateFunction(contact):  
    ContactID, FirstName and MobileNumber are required fields. We ensure they are not  
    null.  
    If ContactID is null:  
        Show error(first name)  
    Endif  
    If FirstName is null:  
        Show error(first name)  
    Endif  
    If MobileNumber is null;  
        Show error(mobile number)  
    End if  
  
    If required fields were entered correctly, save the data to the database  
  
    Database.Update(contact)  
  
    Get the updated data from database and return  
    Return ReadContactFunction(ContactID):  
End function
```

```
Def Delete Function(ContactID):  
    Update StatusID row in ContactTable Record with the delete Status ID  
  
    Return database.delete(contactID)  
    database.delete returns true for success or false for failure  
End function
```

```
Def ShowError(text)
    print(text + " is required.!")
    Return false
    Halt the application processing
End function
```

```
Def Database.Create(SQLStatement)
    Process SQL Statement
    Return Result
End function
```

```
Def Database.ReadContact(SQLStatement)
    Process SQL Statement
    Return Result
End function
```

```
Def Database.ReadContacts(SQLStatement)
    Process SQL Statement
    Return Result
End function
```

```
Def Database.Update(SQLStatement)
    Process SQL Statement
    Return Result
End function
```

```
Def Database.Delete(SQLStatement)
    Process SQL Statement
    Return Result
End function
```

Process Flow

User Start Application Flow:

```
When user starts application
    Display all data
    Display ReadContactFunction()
```

User Submission Process Flow:

When User submits form

 Create List

 ContactList = (string FirstName, string MiddleName, String LastName, MobileNumber, WorkNumber, Email, URL, Address, Comments, StatusID)

 Call the create function and parse the contact list as a parameter

 Result = CreateFunction(ContactList)

 Create Function Returns the newly created ContactID field on success and 0 on fail

 If(Result > 0):

 NewCreatedContactData = ReadContactFunction(Result)

 End If

 Display NewCreatedContactData Information

User update Process Flow

 Select Contact to update

 SelectedContactData = ReadContactFunction(ContactID)

 Display data in a form

 User then modifies all necessary fields and submits upon completion

 On Submission

 Create List

 Create List

 ContactList = (int ContactID, string FirstName, string MiddleName, String LastName, MobileNumber, WorkNumber, Email, URL, Address, Comments, StatusID)

 Request Update by submitting list containing Contact ID

 Result = UpdateFunction(ContactID)

 Function returns success or failure on completion

 If success:

 Display newly updated data

 Display ReadContactFunction(Result)

 Else

 Display Error Message

User Delete Process Flow

When user selects to delete a contact

- Call Delete Function and parse ContactID

- DeleteFunction(ContactID)

- DeleteFunction returns true or false for success or failure respectively

- If success:

- Return all data

- Display ReadContactFunction

- Else:

- Return Error Message

Expected Test Results:

Test Create:

- Check if table actually records data

- Check if after creation actual ContactID is returned

Test Read:

- See if all data are returned

- See if on single contact request correct contact is returned

Test Delete:

- Check if row statusID is updated to the Delete StatusID

Test Update:

- Check if row is updated with correct information

- Check if correct information is returned at the end of processing