

PYTHON

Assignment

Contact Book Project

Objective:

Create a simple command-line Contact Book application in Python that allows users to add, delete, search, and view contacts.

Step-by-Step Instructions:

- 1. Set Up Your Environment:**
 - Make sure you have Python installed on your computer.
 - Choose an IDE or text editor (like VS Code, PyCharm, or Sublime Text) to write your Python code.
- 2. Create a New Python File:**
 - Open your text editor or IDE.
 - Create a new file and name it `contact_book.py`.
- 3. Write the `display_menu()` Function:**
 - This function will display the main menu of the Contact Book. It shows options like adding, deleting, searching, viewing all contacts, and exiting the program.
 - Use the `print()` function to display the menu.
- 4. Write the `add_contact(contact_book)` Function:**
 - This function will allow the user to add a new contact to the contact book.
 - The contact book is a dictionary where the name is the key, and the phone number is the value.
 - Use `input()` to get the contact name and phone number from the user.
- 5. Write the `delete_contact(contact_book)` Function:**
 - This function will allow the user to delete a contact by name.
 - Use `input()` to get the name of the contact to delete.
- 6. Write the `search_contact(contact_book)` Function:**
 - This function will allow the user to search for a contact by name.
 - If the contact is found, it will display the name and phone number.
- 7. Write the `view_all_contacts(contact_book)` Function:**
 - This function will display all contacts stored in the contact book.
 - If there are no contacts, it will inform the user.
- 8. Write the `main()` Function:**
 - This is the main function where the program will start executing.
 - Create an empty dictionary called `contact_book` to store contacts.
 - Use a `while` loop to continuously display the menu and prompt the user to choose an option.

- Call the appropriate function based on the user's choice.
- Use a conditional statement to exit the loop and end the program if the user chooses to exit.

9. Run the Program:

- Save your file.
- Run the program by executing `python contact_book.py` in your terminal or command prompt.
- Test all the features of your Contact Book by adding, deleting, searching, and viewing contacts.

Model Output:

```
----- Contact Book Menu -----
1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit
-----
Choose an option (1-5): 1
Enter contact name: John Doe
Enter contact phone number: 123-456-7890
Contact 'John Doe' added successfully.

----- Contact Book Menu -----
1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit
-----
Choose an option (1-5): 4

--- All Contacts ---
Name: John Doe, Phone: 123-456-7890
-----

----- Contact Book Menu -----
1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit
-----
Choose an option (1-5): 3
Enter contact name to search: John Doe
```

```
Name: John Doe, Phone: 123-456-7890
```

```
----- Contact Book Menu -----
```

1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit

```
-----
```

```
Choose an option (1-5): 2
```

```
Enter contact name to delete: John Doe
```

```
Contact 'John Doe' deleted successfully.
```

```
----- Contact Book Menu -----
```

1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit

```
-----
```

```
Choose an option (1-5): 4
```

```
No contacts found.
```

```
----- Contact Book Menu -----
```

1. Add Contact
2. Delete Contact
3. Search Contact
4. View All Contacts
5. Exit

```
-----
```

```
Choose an option (1-5): 5
```

```
Exiting Contact Book. Goodbye!
```