CodSoft Internship

Task 5: Contact Book

By: Hojaifa Ansari

Introduction:

The Contact Book project is a simple and efficient Python-based application designed to store and manage contact information. It allows users to add, view, search, update, and delete contacts easily through a user-friendly interface.
This project helps in practicing Python concepts like dictionaries, functions, loops, and conditional statements. By working on this project, we learn how to handle data, organize it efficiently, and create interactive console-based applications.

Code:

```
#  Contact Book Project

contacts = {}

def add_contact():
    name = input("Enter Name: ").strip()
```

```
    phone = input("Enter Phone Number: ").strip()
        email = input("Enter Email: ").strip()
    address = input("Enter Address: ").strip()

    contacts[name] = {
        'Phone': phone,
        'Email': email,
        'Address': address
    }
    print(f"\nContact '{name}' added successfully!\n")

def view_contacts():
    if not contacts:
        print("\nNo contacts found.\n")
        return
    print("\n---- Contact List ----")
    for name, info in contacts.items():
        print(f"\nName    : {name}")
        print(f"Phone   : {info['Phone']}")
        print(f"Email   : {info['Email']}")
        print(f"Address : {info['Address']}")

def search_contact():
    search_name = input("Enter name to search: ").strip()
    if search_name in contacts:
        info = contacts[search_name]
        print(f"\nName    : {search_name}")
        print(f"Phone   : {info['Phone']}")
        print(f"Email   : {info['Email']}")
        print(f"Address : {info['Address']}")
    else:
        print("\nContact not found.\n")

def update_contact():
    name = input("Enter the name of the contact to update: ").strip()
    if name in contacts:
```

```
        print("Enter new details (leave blank to keep unchanged):")
        phone = input("New Phone Number: ").strip()
        email = input("New Email: ").strip()
        address = input("New Address: ").strip()

        if phone:
            contacts[name]['Phone'] = phone
                if email:
            contacts[name]['Email'] = email
        if address:
            contacts[name]['Address'] = address

        print(f"\nContact '{name}' updated successfully!\n")
    else:
        print("\nContact not found.\n")

def delete_contact():
    name = input("Enter the name of the contact to delete: ").strip()
    if name in contacts:
        del contacts[name]
        print(f"\nContact '{name}' deleted successfully!\n")
    else:
        print("\nContact not found.\n")

def main():
    print("=" * 50)
    print("              Contact Book Application")
         print("=" * 50)

    while True:
        print("\nMenu:")
        print("1. Add Contact")
        print("2. View Contacts")
        print("3. Search Contact")
        print("4. Update Contact")
        print("5. Delete Contact")
```

```
    print("6. Exit")

      choice = input("Choose an option (1-6): ").strip()

    if choice == '1':
        add_contact()
    elif choice == '2':
        view_contacts()
    elif choice == '3':
        search_contact()
    elif choice == '4':
        update_contact()
    elif choice == '5':
        delete_contact()
    elif choice == '6':
        print("\nThank you for using Contact Book. Goodbye!\n")
        break
    else:
        print("\nInvalid choice! Please try again.\n")

if __name__ == "__main__":
    main()
```

Output:

Contact Book Application

Menu:
1. Add Contact
2. View Contact
3. Search Contact
4. Update Contact
5. Delete Contact

6. Exit Contact
Choose an option (1-6): 1
Enter Name: Huzaifa Badshah
Enter Phone Number: 82073..982
Enter Email: huzaufabadshah82073@gmail.com
Enter Address: Sitamarhi Bihar
Contact Huzaifa Badshah added successfully
Choose an option (1-6):6
Thank you for using Contact Book. Goodbye

Conclusion:

The Contact Book project is a practical implementation of Python programming for managing information. Through this project, we applied key concepts like dictionaries, functions, and loops to build a functional contact management system.
It not only improves programming skills but also develops problem-solving abilities, making it a valuable project for Python learners.