Phonebook Application Project Report

#### - By Sughuman Mahajan

# 1. Introduction

This project involves creating a simple Phonebook application using Python within a Jupyter Notebook. The application allows users to perform basic CRUD operations on phonebook entries, providing a practical exercise in handling Python dictionaries and conditional statements.

# 2. Objectives

- To implement a phonebook application that allows users to add, search, delete, and list contacts.  
- To reinforce understanding of Python dictionaries for storing and managing contact information.  
- To practice using conditional statements and loops for menu-driven user interaction.

# 3. Implementation

## 3.1. Project Setup

The project is implemented in a Jupyter Notebook named phonebook.ipynb. The application uses a dictionary to store phonebook entries, where each contact name serves as a key and the associated phone number as the value.

## 3.2. User Interface

The user interface is text-based and menu-driven. The application presents a series of options to the user, allowing them to choose an action. The available options include adding a new contact, searching for a contact, deleting a contact, listing all contacts, and exiting the application.

## 3.3. CRUD Operations

- \*\*Add a New Contact\*\*: The application prompts the user for a contact name and phone number. It checks if the name already exists in the phonebook before adding the new contact.  
- \*\*Search for a Contact\*\*: The user can search for a contact by entering the name. The application checks if the name exists in the dictionary and displays the corresponding phone number if found.  
- \*\*Delete a Contact\*\*: The application allows the user to delete a contact by entering the name. If the contact exists, it is removed from the dictionary.  
- \*\*List All Contacts\*\*: The application iterates through the dictionary and displays all contacts in a formatted list.

## 3.4. Error Handling

The application includes basic error handling to manage cases where the user attempts to search for or delete a non-existent contact. It also handles invalid menu choices by prompting the user to try again.

## 3.5. Testing

The application was thoroughly tested within the Jupyter Notebook environment to ensure it behaves correctly in various scenarios:  
- Adding new contacts  
- Searching for existing and non-existent contacts  
- Deleting contacts  
- Listing all contacts when the phonebook is empty and when it contains multiple entries

# 4. Challenges

During development, one challenge was ensuring the correct handling of edge cases, such as attempting to delete a contact that does not exist or adding a contact with a name that is already in the phonebook. These were addressed by implementing appropriate conditional checks.

# 5. Conclusion

The Phonebook application successfully demonstrates the use of Python dictionaries and conditional statements in a real-world scenario. It provides a practical example of CRUD operations and user interaction in a Jupyter Notebook-based Python project.

# 6. Future Enhancements

Possible future enhancements to the project could include:  
- Adding functionality to update a contact's phone number.  
- Implementing file storage to persist contacts between program executions.  
- Enhancing error handling and input validation.