

Description

This script is a phone book that allows adding, deleting, updating contacts, and searching them by name, age, and email. It is possible to save the contact list to a file or open an existing file with phone contacts.

This code is a command-line interface for managing a phonebook. It reads a JSON file containing phonebook entries, allowing users to perform CRUD operations on them.

The code uses a set of utility functions for input validation, print statements, and file operations.

Modules

The following modules are used to work with the program:

- copy: for working with object copies
- json: for working with json file format
- argparse: for processing command-line arguments
- utilities.input: for collecting user input data
- utilities.decorator: for decorating functions
- utilities.print: for outputting messages to the user
- utilities.save_file: for saving the contact list to a file

Variables

- running: a flag variable indicating the state of the program
- unsaved_changes: a flag variable indicating whether changes have been made to the contact list
- file_name: the name of the file with phone contacts
- phone_book: the list of contacts

Functions

- print_phonebook(): prints the contact list to the screen
- add_new_entry_to_phonebook(): adds a new contact to the list
- find_entry_by_name_in_phonebook(): finds a contact by name
- find_entry_by_age_in_phonebook(): finds a contact by age
- find_email_by_name_in_phonebook(): finds the email address of a contact by name
- delete_entry_by_name_in_phonebook(): deletes a contact by name
- count_all_entries_in_phonebook(): counts the number of contacts in the list
- print_phonebook_by_age(): prints the contact list sorted by age

`update_contact_ages()`: updates the age of all contacts
`calculate_average_age_of_all_persons_in_phonebook()`: calculates the average age of all contacts in the list
`save_phonebook_to_file()`: saves the contact list to a file

Command Line Parameters

The code has an `argparse` module that allows the user to pass a filename as an argument while running the code in the command-line interface. It also enables verbose mode using the `-v/--verbose` option.

`url` - The first argument, `url`, is a positional argument that specifies the path to the file that contains the phone book data. The `type` parameter specifies that this argument should be a string, and the `help` parameter provides a brief description of the argument for the user.

`-v` or `--verbose` - The second argument, is an optional argument that enables decorators when passed. The `action` parameter specifies that the argument should be a boolean flag, and the `help` parameter provides a brief description of the argument for the user.