Evgeniia Lipatova

Project: Calendar

Project Description:
The Calendar is a Python-based application that allows users to manage and interact with reminders. Reminders are organised by date, and users can perform various actions such as viewing reminders for the day, adding new reminders, searching by date or keyword, and more.

Solution Principle:
The solution is built on a Python script using the *datetime* module for handling date-related functionalities and the *csv* module for reading and writing reminders to a CSV file. The application uses a simple command-line interface where users can input commands to perform different actions.

Project Structure:
The project is organised into the following components:

1. Main Script (`calendar3.py`):
   - Imports the required modules (`datetime` and `csv`).
   - Defines the current date, reads reminders from the CSV file, and initialises the user interface.

2. Functions:
   - `read_file()`: Reads reminders from the CSV file or creates an empty file if it doesn't exist.
   - `check_command()`: Checks user input and calls the corresponding function based on the command.
   - `print_today()`: Prints reminders for the current day.
   - `help()`: Displays available commands.
   - `print_month()`: Prints a calendar view for a specified month.
   - `print_week()`: Prints reminders for a specified week.
   - `add_reminder()`: Adds a new reminder.
   - `check_date(entry_date)`: Checks if a given date has reminders.
   - `find_date()`: Finds and displays reminders for a specified date.
   - `find_reminder()`: Finds and displays reminders containing a specified keyword.

3. User Interface:
   - Prompts users with available commands and processes user input.

Functions and Interrelationships:
- `read_file()` is called at the beginning to read existing reminders.
- `check_command()` interprets user commands and calls corresponding functions.
- Various functions (`print_today()`, `print_month()`, `print_week()`, `add_reminder()`, `find_date()`, `find_reminder()`) provide specific functionalities and interact with the reminders data.

External Libraries:
- `datetime`: Used for date-related operations.
- `csv`: Utilised for reading from and writing to a CSV file.

Usage:

1. Run the script (`calendar3.py`).
2. Input commands as prompted to interact with the Reminder System.