Mini Project on

To-Do List Application

Created by

Sunit Patil

Contents

1 Introduction

The To-Do List Application project aims to create a functional task management system using Python. This project showcases practical application of Python programming concepts including functions, data structures, file handling, and user interface design. The application allows users to manage their tasks by adding, deleting, marking as complete, and resetting tasks as needed.

2 Project Overview

The objective of this project is to develop a To-Do List Application using Python with an emphasis on modular code structure and efficient data management. Key components of the project include:

- **Functions**: Modular functions for adding tasks, deleting tasks, marking tasks as complete, and resetting tasks.

- **Data Structures**: Use of JSON file for persistent data storage and retrieval.

- **User Interface**: Command-line interface (CLI) implemented in `app.py` to interact with users and manage task operations.

3 Code Explanation

3.1 Task Management Functions

The task management functions handle various operations on tasks within the application:

- `add_task(task)`: Adds a new task to the list with a unique ID.

- `delete_task(task_id)`: Deletes a task based on its ID.

- `delete_all_tasks()`: Deletes all tasks from the list.

- `reset_all_tasks()`: Resets all tasks to an incomplete status.

- `reset_task(task_id)`: Resets a specific task to an incomplete status based on its ID.

- `mark_task_complete(task_id)`: Marks a task as complete based on its ID.

3.2 User Interface (`app.py`)

The `app.py` file contains the main user interface of the application, providing a menu-driven CLI for users to interact with the To-Do List:
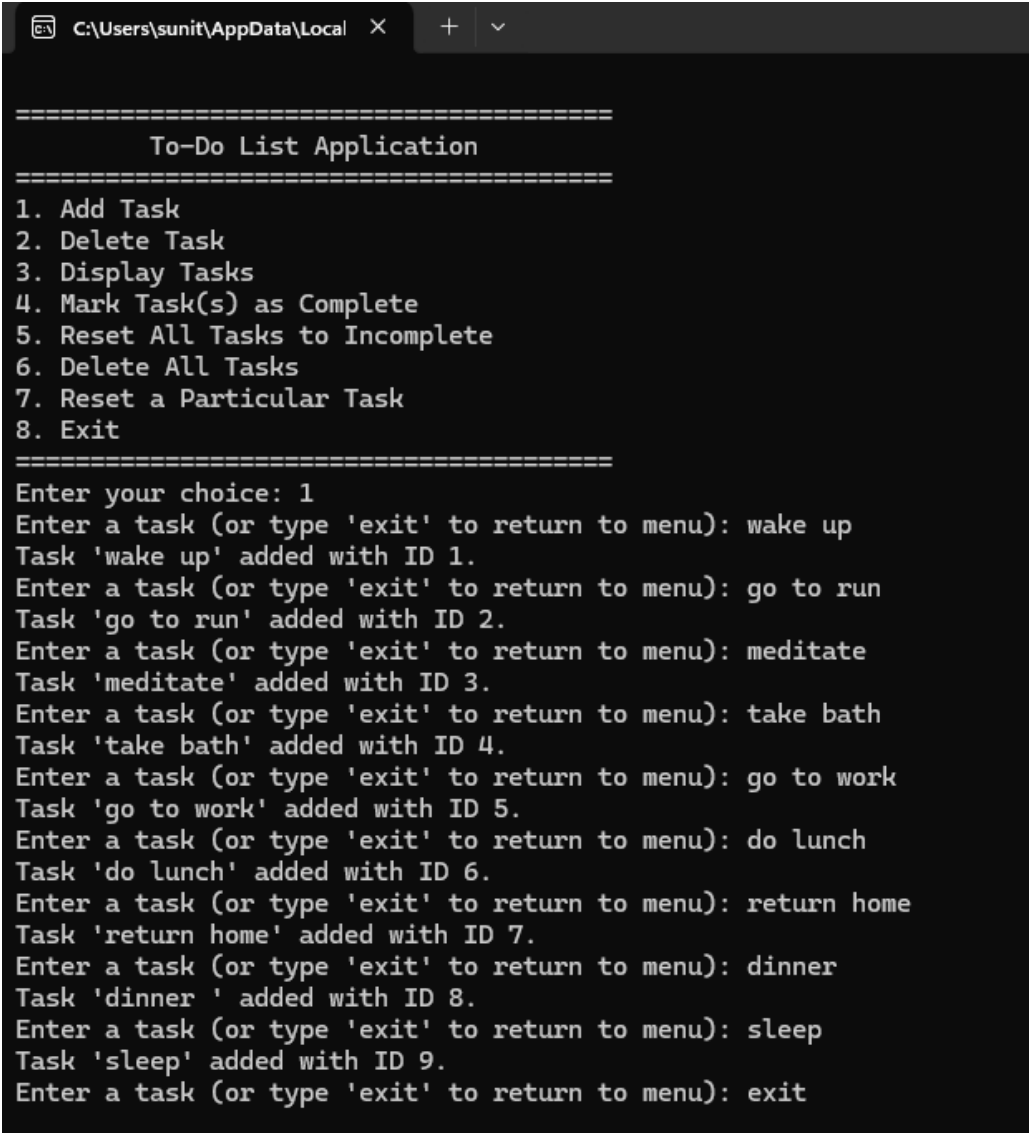
- `show_menu()`: Displays a menu with options for users to add, delete, display, mark tasks, and manage task status.

- **Main Loop**: Continuously prompts the user for input and executes corresponding functions based on their choices.

3.3 Data Management with JSON

The project utilizes JSON (JavaScript Object Notation) for storing and managing task data. JSON provides a lightweight data-interchange format that is easy to read and write for both humans and machines.

- `load_tasks()`: Loads existing tasks from a JSON file into memory.

- `save_tasks(tasks)`: Saves the updated list of tasks back to the JSON file after modifications.

4 Screenshots


```
========================================
          To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 1
Enter a task (or type 'exit' to return to menu): wake up
Task 'wake up' added with ID 1.
Enter a task (or type 'exit' to return to menu): go to run
Task 'go to run' added with ID 2.
Enter a task (or type 'exit' to return to menu): meditate
Task 'meditate' added with ID 3.
Enter a task (or type 'exit' to return to menu): take bath
Task 'take bath' added with ID 4.
Enter a task (or type 'exit' to return to menu): go to work
Task 'go to work' added with ID 5.
Enter a task (or type 'exit' to return to menu): do lunch
Task 'do lunch' added with ID 6.
Enter a task (or type 'exit' to return to menu): return home
Task 'return home' added with ID 7.
Enter a task (or type 'exit' to return to menu): dinner
Task 'dinner ' added with ID 8.
Enter a task (or type 'exit' to return to menu): sleep
Task 'sleep' added with ID 9.
Enter a task (or type 'exit' to return to menu): exit
```

Figure 1: ADD TASKS

Figure 2: Display tasks

```
===================================
        To-Do List Application
===================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
===================================
Enter your choice: 4

===================================
ID   Task               Status
-----------------------------------
1    wake up            Incomplete
2    go to run          Incomplete
3    meditate           Incomplete
4    take bath          Incomplete
5    go to work         Incomplete
6    do lunch           Incomplete
7    return home        Incomplete
8    dinner             Incomplete
9    sleep              Incomplete
===================================
Enter task IDs to mark as complete (comma-separated), or 'exit' to return to menu: 1
Task with ID 1 marked as complete.

===================================
ID   Task               Status
-----------------------------------
1    wake up            Complete
2    go to run          Incomplete
3    meditate           Incomplete
4    take bath          Incomplete
5    go to work         Incomplete
6    do lunch           Incomplete
7    return home        Incomplete
8    dinner             Incomplete
9    sleep              Incomplete
===================================
Enter task IDs to mark as complete (comma-separated), or 'exit' to return to menu: 2
Task with ID 2 marked as complete.

===================================
ID   Task               Status
-----------------------------------
1    wake up            Complete
2    go to run          Complete
3    meditate           Incomplete
4    take bath          Incomplete
5    go to work         Incomplete
6    do lunch           Incomplete
7    return home        Incomplete
8    dinner             Incomplete
9    sleep              Incomplete
===================================
Enter task IDs to mark as complete (comma-separated), or 'exit' to return to menu: 3
Task with ID 3 marked as complete.
```

Figure 3: Mark tasks for completion

Figure 4: Tasks display after completion

Figure 5: Deletion of task

```
========================================
         To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 2

========================================
ID   Task                 Status
----------------------------------------
1    wake up              Complete
2    go to run            Complete
3    meditate             Complete
5    go to work           Incomplete
6    do lunch             Incomplete
7    return home          Incomplete
8    dinner               Incomplete
9    sleep                Incomplete
========================================
Enter the task ID to delete: 4
Task with ID 4 not found.

========================================
         To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 3

========================================
ID   Task                 Status
----------------------------------------
1    wake up              Complete
2    go to run            Complete
3    meditate             Complete
5    go to work           Incomplete
6    do lunch             Incomplete
7    return home          Incomplete
8    dinner               Incomplete
9    sleep                Incomplete
========================================
```

Figure 6: Deleting tasks and display after deleting

```
========================================
        To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 5
All tasks have been reset to incomplete.

========================================
        To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 3

========================================
ID    Task                Status
----------------------------------------
1     wake up             Incomplete
2     go to run           Incomplete
3     meditate            Incomplete
5     go to work          Incomplete
6     do lunch            Incomplete
7     return home         Incomplete
8     dinner              Incomplete
9     sleep               Incomplete
========================================

========================================
        To-Do List Application
========================================
1. Add Task
2. Delete Task
3. Display Tasks
4. Mark Task(s) as Complete
5. Reset All Tasks to Incomplete
6. Delete All Tasks
7. Reset a Particular Task
8. Exit
========================================
Enter your choice: 6
All tasks have been deleted.
```

Figure 7: Above screenshot shows following outputs

1) Reset all tasks

2) Display after reset

3) Delete All tasks

9

Figure 8: ALL TASKS DELETED

5   Conclusion

In conclusion, the To-Do List Application project has provided hands-on experience in applying Python programming skills to develop a practical solution for task management. The project highlights the importance of modular coding practices, efficient data handling, and user interface design in software development.