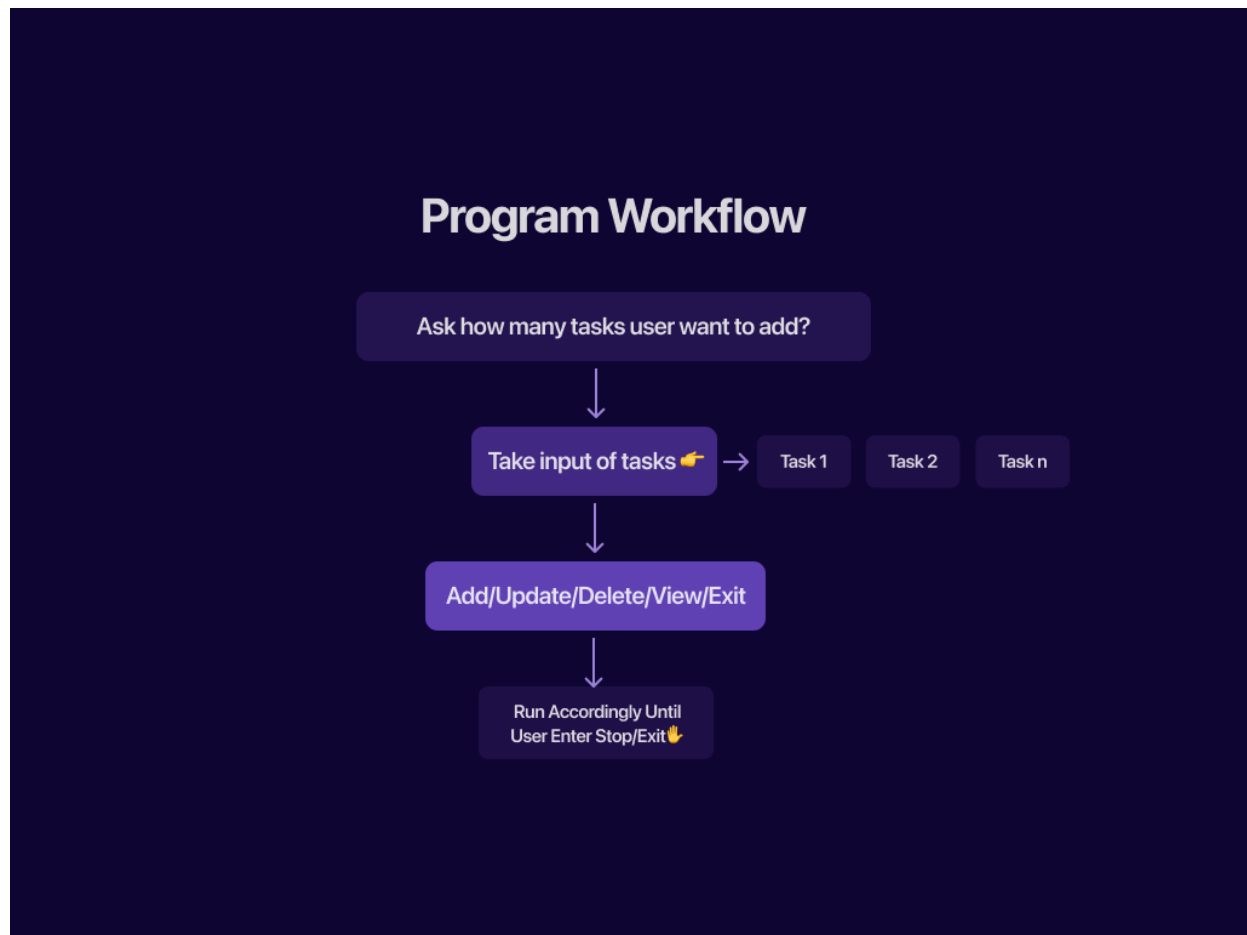


Task Management App



Task Management App Documentation

Overview

The Task Management App is a command-line interface (CLI) Python application that allows users to manage tasks effectively. Users can add, update, delete, and view tasks using simple text-based commands. The app operates in a continuous loop, processing user commands until the user exits.

Program Structure

The app consists of a single Python function, `task()`, which orchestrates all operations.

Detailed Functionality

1. Initialization

```
tasks = []
```

```
print("----WELCOME TO THE TASK MANAGEMENT APP----")
```

- tasks (list): An empty list that will hold the tasks added by the user.
- Welcome Message: Displays a greeting and introductory message when the app starts.

2. Adding Initial Tasks

```
total_task = int(input("Enter how many task you want to add = "))
for i in range(1, total_task + 1):
    task_name = input(f"Enter task {i} = ")
    tasks.append(task_name)
print(f"Today's tasks are\n{tasks}")
```

- Input Total Tasks: The user is prompted to specify how many tasks they intend to add initially.
- Task Input Loop: For each task, the app prompts for the task's name and appends it to the tasks list.
- Display Initial Tasks: After all initial tasks are entered, they are displayed to the user.

3. Main Operation Loop

```
while True:
    operation = int(input("Enter \n1-Add\n2-Update\n3-Delete\n4-View\n5-Exit/Stop/"))
    ...
```

This section initiates an infinite loop to continuously accept user commands until the exit option is chosen. Each option corresponds to a task management operation:

- Add (1): Adds a new task to the tasks list.

```
add = input("Enter task you want to add = ")
tasks.append(add)
print(f"Task {add} has been successfully added...")
```

- Update (2): Updates an existing task. The user specifies which task to update and provides a new task name.

```
updated_val = input("Enter the task name you want to update = ")
if updated_val in tasks:
    up = input("Enter new task = ")
    ind = tasks.index(updated_val)
    tasks[ind] = up
    print(f"Updated task {up}")
```

- Delete (3): Removes a specified task from the tasks list.

```

del_val = input("Which task you want to delete = ")
if del_val in tasks:
    ind = tasks.index(del_val)
    del tasks[ind]
    print(f"Task {del_val} has been deleted...")

```

View (4): Displays all tasks currently in the list.

```

print(f"Total tasks = {tasks}")

```

-
- Exit (5): Exits the application.

```

print("Closing the program....")
break

```

-
- Invalid Input Handling: If a non-valid option number is entered, the program alerts the user.

```

print("Invalid Input")

```

-

Usage

To use the app, run the Python script and follow the on-screen prompts to manage tasks. Choose from the provided options to add, update, delete, or view tasks. When finished, select the exit option to close the application.

Error Handling

Currently, the application handles invalid task operations gracefully by checking if tasks exist before updating or deleting. However, it could benefit from additional error handling for non-numeric input when choosing operations.

Conclusion

This documentation provides a comprehensive guide to the Task Management App. Each component and its functionality are detailed, providing clear instructions on how the app is structured and how to interact with it. Future improvements could include robust error handling and data persistence.

```

def task():

```

```

tasks = [] # Initialize an empty list to store tasks
print("----WELCOME TO THE TASK MANAGEMENT APP----")
total_task = int(input("Enter how many tasks you want to add = "))
for i in range(1, total_task + 1):
    task_name = input(f"Enter task {i} = ")
    tasks.append(task_name)
print(f"Today's tasks are\n{tasks}")

while True:
    try:
        operation = int(input("Enter \n1-Add\n2-Update\n3-Delete\n4-View\n5-Exit/Stop\n"))
        if operation == 1:
            add = input("Enter task you want to add = ")
            tasks.append(add)
            print(f"Task '{add}' has been successfully added.")
        elif operation == 2:
            updated_val = input("Enter the task name you want to update = ")
            if updated_val in tasks:
                up = input("Enter new task = ")
                ind = tasks.index(updated_val)
                tasks[ind] = up
                print(f"Updated task '{updated_val}' to '{up}'.")
            else:
                print("Task not found.")
        elif operation == 3:
            del_val = input("Which task you want to delete = ")
            if del_val in tasks:
                ind = tasks.index(del_val)
                del tasks[ind]
                print(f"Task '{del_val}' has been deleted.")
            else:
                print("Task not found.")
        elif operation == 4:
            print(f"Total tasks = {tasks}")
        elif operation == 5:
            print("Closing the program....")
            break
        else:
            print("Invalid Input. Please enter a number from 1 to 5.")
    except ValueError:
        print("Invalid Input. Please enter a valid number.")

```

task()

