**TASK MANAGER CLI APPLICATION:**

OVERVIEW:

The Task Manager CLI Application is a Python-based command-line tool designed to help users efficiently manage their tasks. With a focus on simplicity and usability, this application allows users to add, view, update, and delete tasks from the command line.

Objectives:

The objectives of the Task Manager CLI Application are:

* To provide an intuitive command-line interface for users to manage their tasks without the need for a graphical user interface (GUI).
* To allow users to add, view, and delete tasks, as well as mark tasks as completed, promoting better organization and task tracking.
* To implement persistent storage, ensuring that users can save and load their tasks from a file, enabling seamless continuation of their task management.

Features

Add Task: Create new tasks with a title and description.

View Tasks: Display all tasks along with their completion status.

Update Task: Modify task details, including changing status.

Delete Task: Remove unnecessary tasks from the list.

Search Tasks: Find tasks using keywords in titles or descriptions.

Technologies Used

* **Programming Language**: Python
* **Data Storage**: JSON file format for saving and loading tasks

How to run it on command line interface

1.Open command Prompt.

2. Use the cd (change directory) command to navigate to the location where you want to create your project folder.



3. Create the folder (directory).

Use the mkdir command (short for "make directory") to create a folder. Name the folder task\_manager.



4.Navigate into folder

After creating the folder, you can navigate inside it using the cd command.



5. Once you’ve created task\_manager.py, you can start writing your Python code and run it on command prompt

Program Explaination:

import json: This imports the JSON module, which is used for saving and loading tasks from a file.

**class Task**: This defines a Task class that represents a task in the application.

**Attributes**:

i).id: An integer that uniquely identifies each task.

ii).title: A string representing the title of the task.

completed: A boolean that indicates whether the task has been completed.

iii).\_\_repr\_\_ method: This method defines how to represent a Task object as a string, showing the task's ID, title, and status.

iv)tasks: A list to store all the Task objects created by the user.: A list to store all the Task objects created by the user.

v). Adding a Task: This function creates a new task:

* Calculates the task\_id based on the current length of the tasks list.
* Creates a new Task object and appends it to the tasks list.

vi) Viewing Tasks: This function displays all tasks:

* If no tasks are available, it informs the user.
* Otherwise, it iterates through the tasks list and prints each task using its string representation.

vii).Delete Task: This function removes a task by its ID:

viii).Save task function: this function saves the current tasks to a JSON file:

It uses json.dump() to serialize the list of task dictionaries into a file

ix).Loading Task: This function loads tasks from a JSON file:

It reads the JSON file and reconstructs Task objects from the stored dictionaries.

If the file does not exist, it catches the exception and does nothing.

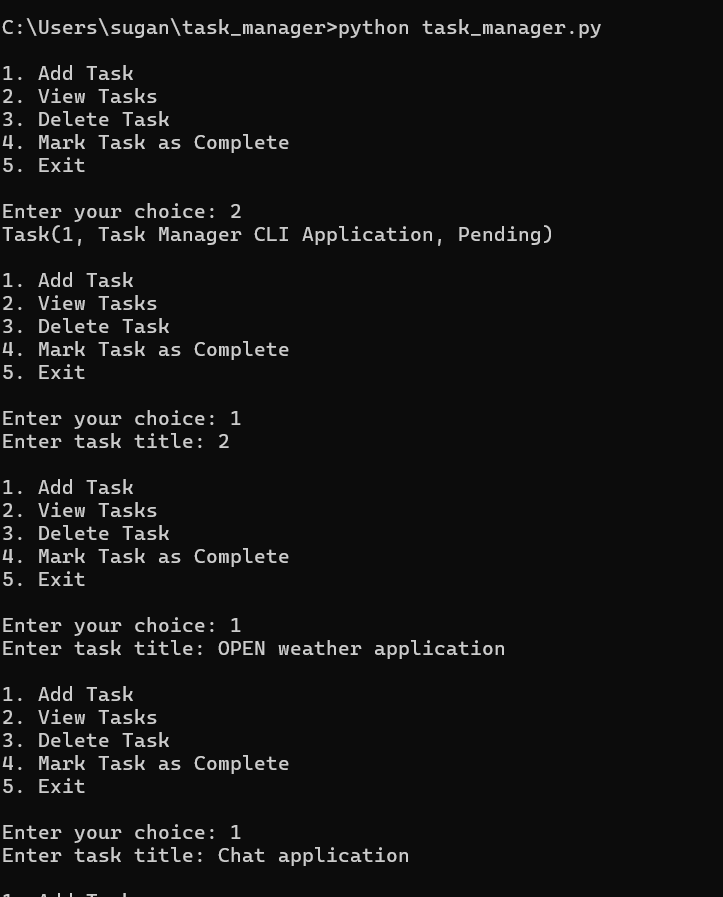
x).command line interface:display menu wgile executing.

xi)The Main fun():

This is the main loop of the application:

* It loads tasks when the application starts.
* Displays the menu and takes user input.
* Based on the user's choice, it calls the corresponding function to manage tasks.
* When exiting, it saves the tasks to ensure no data is lost.

Output:



Functionalities:

* The application starts, and the user is presented with a menu of options.
* The user selects an option (e.g., add a task) and follows the prompts.
* Based on user inputs, the corresponding function is executed (adding, viewing, deleting, or completing tasks).
* The user can repeat this process until they choose to exit, at which point their tasks are saved.

Conclusion:

The Task Manager CLI Application allows users to manage their tasks through a command-line interface, providing essential features like adding, viewing, deleting, and completing tasks. With persistent storage through JSON files, users can maintain their tasks across sessions, making it a valuable tool for personal productivity.