**Task Manager CLI Application**

**1. Project Title and Description**

**Task Manager CLI Application**  
This project is a simple command-line tool for managing tasks. Users can add, view, delete, and mark tasks as complete. Tasks are saved in a JSON file to retain them between sessions.

**2. Project Setup**

To set up the project:

1. Create a project directory, e.g., task\_manager.
2. Inside the directory, add the main file task\_manager.py and a README.md file.
3. Ensure you have Python installed to run the application.

**3. Task Structure**

Each task has the following attributes:

* **id**: An integer that serves as a unique identifier for each task.
* **title**: A string that represents the task description.
* **completed**: A boolean indicating whether the task is completed.

**4. Task Management Functionalities**

The application supports the following operations:

* **Add a Task**: Create a new task and assign it a unique ID.
* **View Tasks**: Display all tasks, indicating whether each is completed.
* **Delete a Task**: Remove a task using its ID.
* **Mark Task as Complete**: Update a task’s status to indicate it’s completed.

**5. File Handling**

Tasks are stored in a tasks.json file to save changes:

* **Save Tasks**: The application saves all tasks to tasks.json whenever they are modified.
* **Load Tasks**: The application loads existing tasks from tasks.json on startup to maintain task continuity.

**6.Code:**

import json

class Task:

  def \_\_init\_\_(self, id, title, completed=False):

    self.id = id

    self.title = title

    self.completed = completed

  def \_\_str\_\_(self):

    status = "Completed" if self.completed else "Pending"

    return f"{self.id}. {status}: {self.title}"

def load\_tasks():

  try:

    with open("tasks.json", "r") as f:

      tasks = json.load(f)

      return [Task(\*\*task) for task in tasks]

  except FileNotFoundError:

    return []

def save\_tasks(tasks):

  with open("tasks.json", "w") as f:

    json.dump([task.\_\_dict\_\_ for task in tasks], f, indent=4)

def add\_task():

  title = input("Enter task title: ")

  tasks.append(Task(len(tasks) + 1, title))

  print("Task added successfully!")

def view\_tasks():

  if not tasks:

    print("There are no tasks to display.")

    return

  for task in tasks:

    print(task)

def delete\_task():

  task\_id = int(input("Enter task ID to delete: "))

  for i, task in enumerate(tasks):

    if task.id == task\_id:

      del tasks[i]

      print(f"Task {task\_id} deleted successfully!")

      return

  print(f"Task with ID {task\_id} not found.")

def mark\_complete():

  task\_id = int(input("Enter task ID to mark complete: "))

  for task in tasks:

    if task.id == task\_id:

      task.completed = True

      print(f"Task {task\_id} marked as complete!")

      return

  print(f"Task with ID {task\_id} not found.")

def main\_menu():

  print("\nTask Manager")

  print("1. Add Task")

  print("2. View Tasks")

  print("3. Delete Task")

  print("4. Mark Task Complete")

  print("5. Exit")

def main():

  global tasks

  tasks = load\_tasks()

  while True:

    main\_menu()

    choice = input("Enter your choice: ")

    if choice == '1':

      add\_task()

    elif choice == '2':

      view\_tasks()

    elif choice == '3':

      delete\_task()

    elif choice == '4':

      mark\_complete()

    elif choice == '5':

      save\_tasks(tasks)

      print("Exiting Task Manager.")

      break

    else:

      print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

  main()

[

    {

        "id": 1,

        "title": "playing",

        "completed": false

    },

    {

        "id": 2,

        "title": "playing",

        "completed": false

    },

    {

        "id": 3,

        "title": "ststus update",

        "completed": false

    }

]

**7. Running the Application**

To run the task manager:

1. Open a terminal.
2. Navigate to the project directory.
3. Run the command:

bash

Copy code

python task\_manager.py

A menu will appear, allowing you to add, view, delete, complete tasks, or exit the application.