In [ ]:

```python
import os
import json
from datetime import datetime

class TaskManager:
    def __init__(self, file_path):
        self.file_path = file_path
        self.tasks = []
        self.load_tasks()

    def load_tasks(self):
        if os.path.exists(self.file_path):
            with open(self.file_path, 'r') as f:
                self.tasks = json.load(f)

    def save_tasks(self):
        with open(self.file_path, 'w') as f:
            json.dump(self.tasks, f, indent=4)

    def add_task(self, title, priority='medium', due_date=None):
        new_task = {'title': title, 'priority': priority, 'due_date': due_date, 'completed': False}
        self.tasks.append(new_task)
        self.save_tasks()

    def remove_task(self, task_index):
        if 0 <= task_index < len(self.tasks):
            del self.tasks[task_index]
            self.save_tasks()
        else:
            print("Invalid task index!")

    def mark_completed(self, task_index):
        if 0 <= task_index < len(self.tasks):
            self.tasks[task_index]['completed'] = True
            self.save_tasks()
        else:
            print("Invalid task index!")

    def list_tasks(self):
        for i, task in enumerate(self.tasks):
            status = "Completed" if task['completed'] else "Not Completed"
            due_date = task['due_date'] if task['due_date'] else "None"
            print(f"{i + 1}. {task['title']} - Priority: {task['priority']}, Due Date: {due_date}, Status: {sta


def main():
    file_path = "tasks.json"
    task_manager = TaskManager(file_path)

    while True:
        print("\n====== To-Do List ======")
        print("1. Add Task")
        print("2. Remove Task")
        print("3. Mark Task as Completed")
        print("4. List Tasks")
        print("5. Exit")

        choice = input("Enter your choice: ")

        if choice == '1':
            title = input("Enter task title: ")
            priority = input("Enter priority (high/medium/low): ")
            due_date = input("Enter due date (YYYY-MM-DD) or leave blank: ")
            if due_date:
                due_date = datetime.strptime(due_date, "%Y-%m-%d").strftime("%Y-%m-%d")
            task_manager.add_task(title, priority, due_date)
            print("Task added successfully!")

        elif choice == '2':
            task_index = int(input("Enter task index to remove: ")) - 1
            task_manager.remove_task(task_index)
            print("Task removed successfully!")

        elif choice == '3':
            task_index = int(input("Enter task index to mark as completed: ")) - 1
            task_manager.mark_completed(task_index)
            print("Task marked as completed!")

        elif choice == '4':
            task_manager.list_tasks()

        elif choice == '5':
            print("Exiting...")
            break

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


if __name__ == "__main__":
    main()
```