

## Simple task list

```
import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.metrics.pairwise import cosine_similarity


class TaskManager:

    def __init__(self):

        try:

            self.tasks = pd.read_csv('tasks.csv')

        except FileNotFoundError:

            self.tasks = pd.DataFrame(columns=['Task'])


    def add_task(self, task):

        self.tasks = self.tasks.append({'Task': task}, ignore_index=True)

        self.tasks.to_csv('tasks.csv', index=False)

        print("Task added successfully!")


    def remove_task(self, task_index):

        try:

            self.tasks.drop(task_index, inplace=True)

            self.tasks.reset_index(drop=True, inplace=True)

            self.tasks.to_csv('tasks.csv', index=False)

            print("Task removed successfully!")

        except IndexError:

            print("Task index out of range!")


    def list_tasks(self):

        if len(self.tasks) == 0:
```

```

        print("No tasks found!")
    else:
        print("List of tasks:")
        for index, task in enumerate(self.tasks['Task']):
            print(f"{index+1}. {task}")

def prioritize_tasks(self):
    if len(self.tasks) == 0:
        print("No tasks found!")
    else:
        # Calculate TF-IDF matrix
        vectorizer = TfidfVectorizer()
        tfidf_matrix = vectorizer.fit_transform(self.tasks['Task'])

        # Calculate cosine similarity matrix
        cosine_sim = cosine_similarity(tfidf_matrix, tfidf_matrix)

        # Sort tasks based on cosine similarity
        task_indices = sorted(range(len(cosine_sim[0])), key=lambda i: cosine_sim[0][i],
reverse=True)

        print("Recommended order of tasks:")
        for index in task_indices:
            print(f"- {self.tasks['Task'][index]}")

def main():
    task_manager = TaskManager()

    while True:

```

```
print("\nTask Management App Menu:")
print("1. Add Task")
print("2. Remove Task")
print("3. List Tasks")
print("4. Prioritize Tasks")
print("5. Exit")

choice = input("Enter your choice: ")

if choice == '1':
    task = input("Enter task description: ")
    task_manager.add_task(task)
elif choice == '2':
    task_manager.list_tasks()
    task_index = int(input("Enter the index of the task to remove: ")) - 1
    task_manager.remove_task(task_index)
elif choice == '3':
    task_manager.list_tasks()
elif choice == '4':
    task_manager.prioritize_tasks()
elif choice == '5':
    print("Exiting the application...")
    break
else:
    print("Invalid choice! Please select a valid option.")

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