A black text on a white background

Description automatically generatedA white and black background

Description automatically generated with medium confidence

**University of Westminster**

**Module Name:**  Software Development 1

**Module leader**: Guhanathan Poravi

**Assignment type:** Individual coursework

**Group ID**: 01

**Submission date:** 21.04.2025

|  |  |  |
| --- | --- | --- |
| **Student Id** | **Uow ID** | **Student Name** |
| 20241636 | w2149533 | M.N.M.Riyas |

**School of Computing**

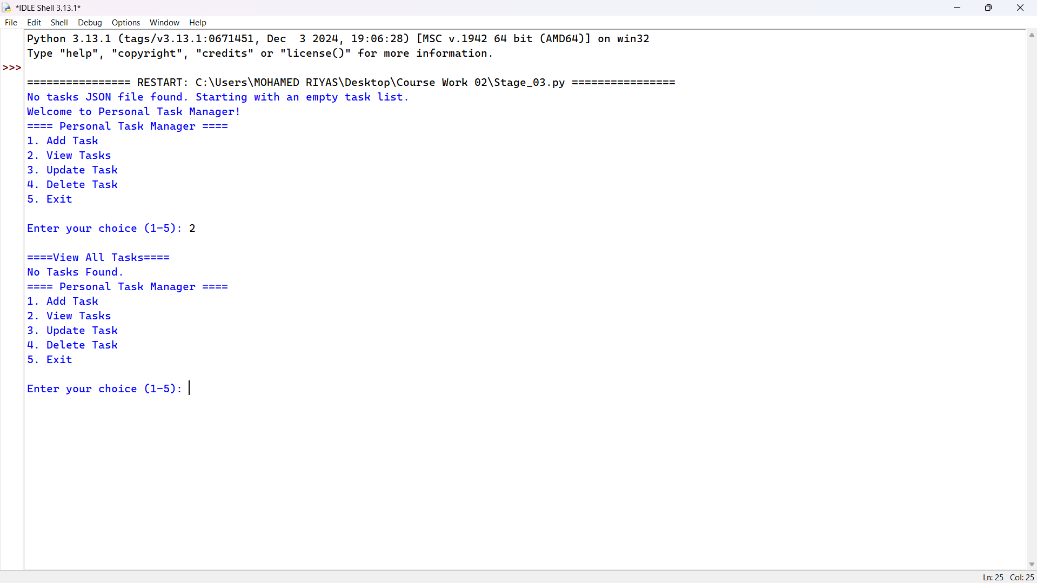
**Informatics Institute of Technology (IIT Campus)**

Table of Contents.

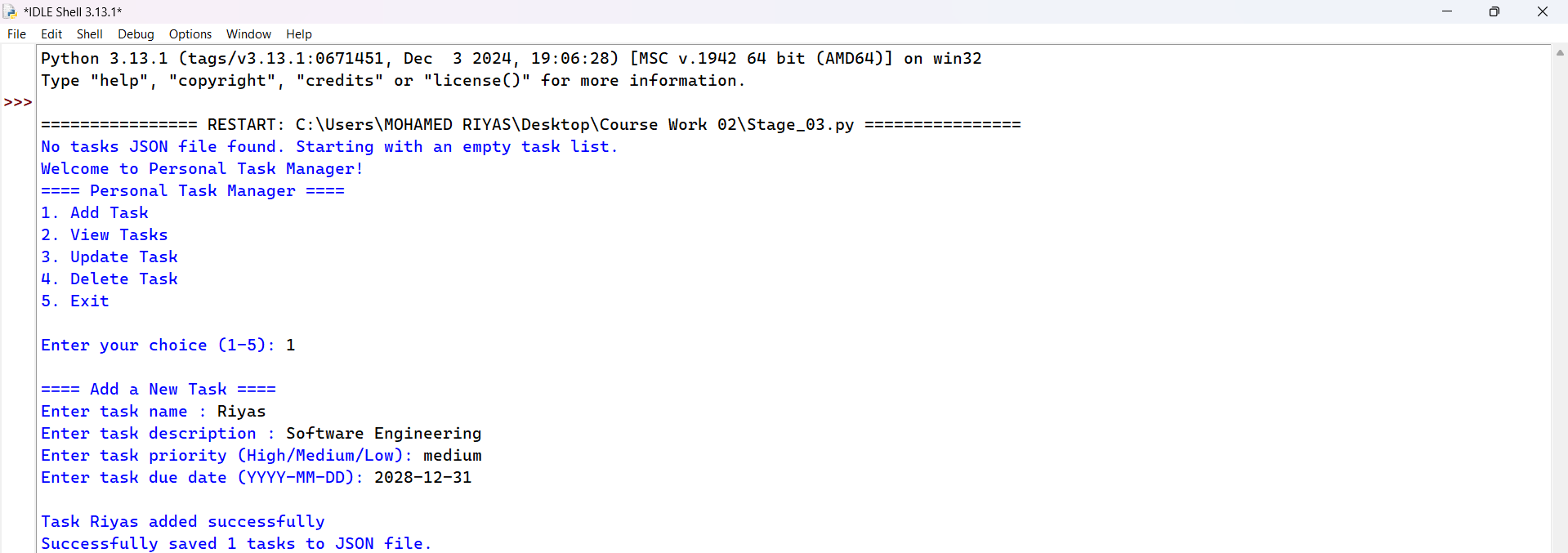
1. **Stage 03 - Test cases and their outcomes:** ………………………………..... 3
2. Test Case 1 Outcomes: No tasks available message ………………………… 4
3. Test Case 2 Outcomes: Successfully added task and save to **JSON file** ……. 4
4. Test Case 3 Outcomes: Display the all task with detail correctly and **JSON file**……………………………………………………………………………. 4
5. Task Case 4 Outcomes: Updated task and saved into **JSON file** ………….... 5
6. Test Case 5 Outcomes: Task successfully deleted from **JSON file** …………. 5
7. Test Case 6 Outcomes: Task saved in tasks.json ……………………………. 5
8. Test Casse 7 Outcomes: Restart after saving tasks in tasks.json ……………. 5
9. **Stage 04 - Task Management System (Tkinter GUI)** …………………... 6
10. Introduction ………………………………………………………………….. 6
11. Features Implemented ……………………………………………………….. 6
12. Test Cases & Results ………………………………………………………… 7
13. Figure 1: Main Interface …………………………………………………. 8
14. Figure 2: Filtering Tasks …………………………………………………. 8
15. Figure 3: Add New Task …………………………………………………. 9
16. Figure 4: Invalid Date Format ………………………………………….. 10
17. **Stage 03 - Test Cases and their Outcomes**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Test Case No:*** | ***Input*** | ***Expected Outcome*** | ***Actual Outcome*** | ***Result*** |
| 1 | Viewing tasks when no tasks exist | Display 'No tasks found' | As Expected | Pass |
| 2 | Adding a task with valid inputs | Task successfully added and stored in **JSON file** | As Expected | Pass |
| 3 | Viewing tasks after added tasks | Display the all task with detail correctly and **JSON file** | As Expected | Pass |
| 4 | Updating a task that exists | Task updated and saved in **JSON file** | Task updated successfully in **JSON file** | Pass |
| 5 | Deleting a task that exists | Task removed from **JSON file** | Task deleted successfully in **JSON file** | Pass |
| 6 | Save tasks with valid inputs | Data saved in tasks.json | Task saved in tasks.json | Pass |
| 7 | Restart after saving tasks | Tasks loaded successfully **JSON** | Except tasks are display | Pass |

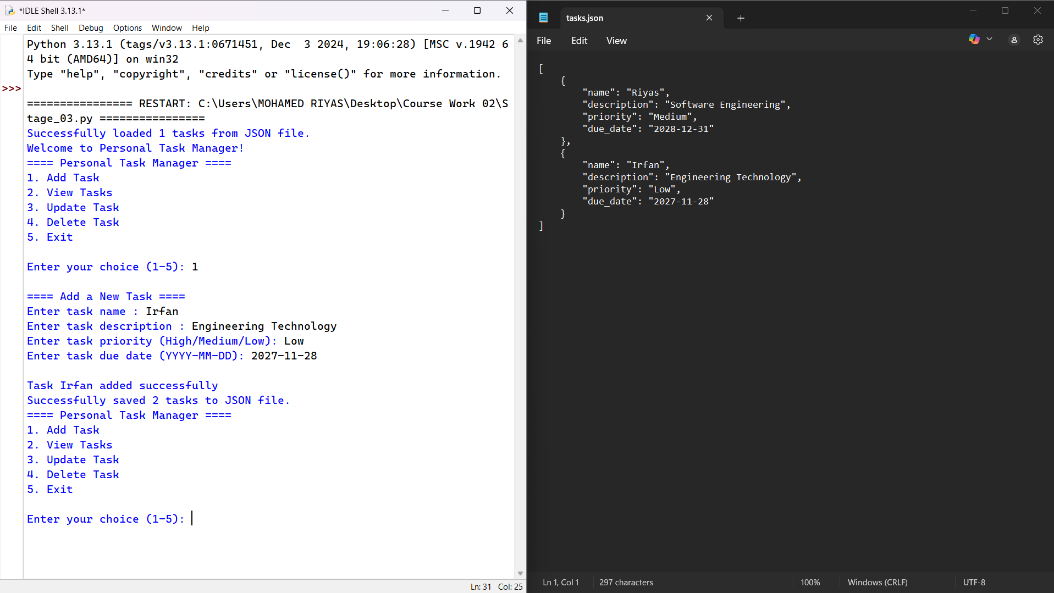
1. Test Case 1 Outcomes: No tasks available message



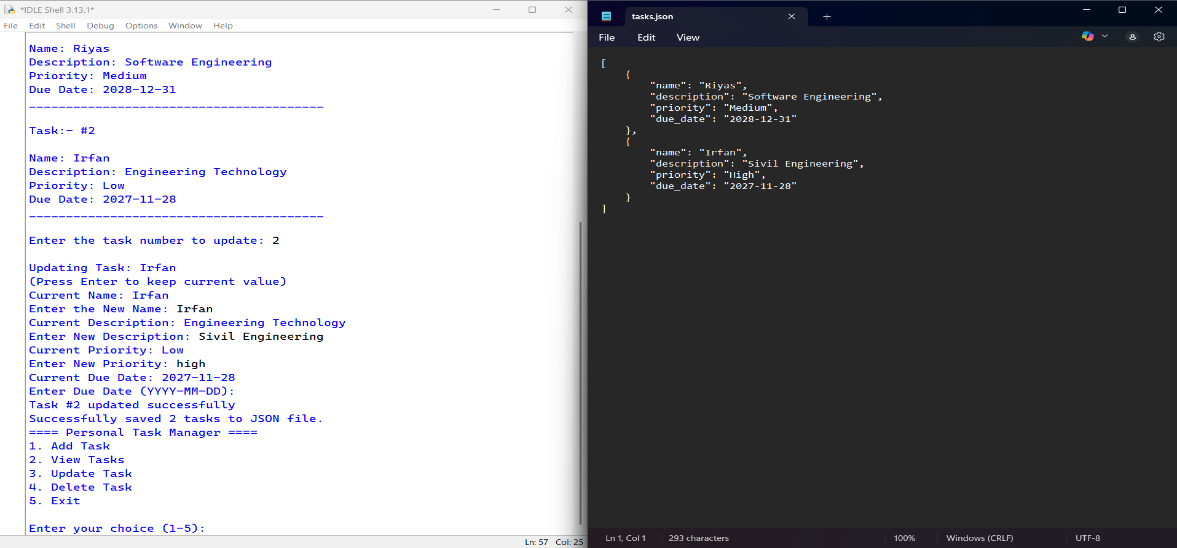
1. Test Case 2 Outcomes: Successfully added task and save to **JSON file**



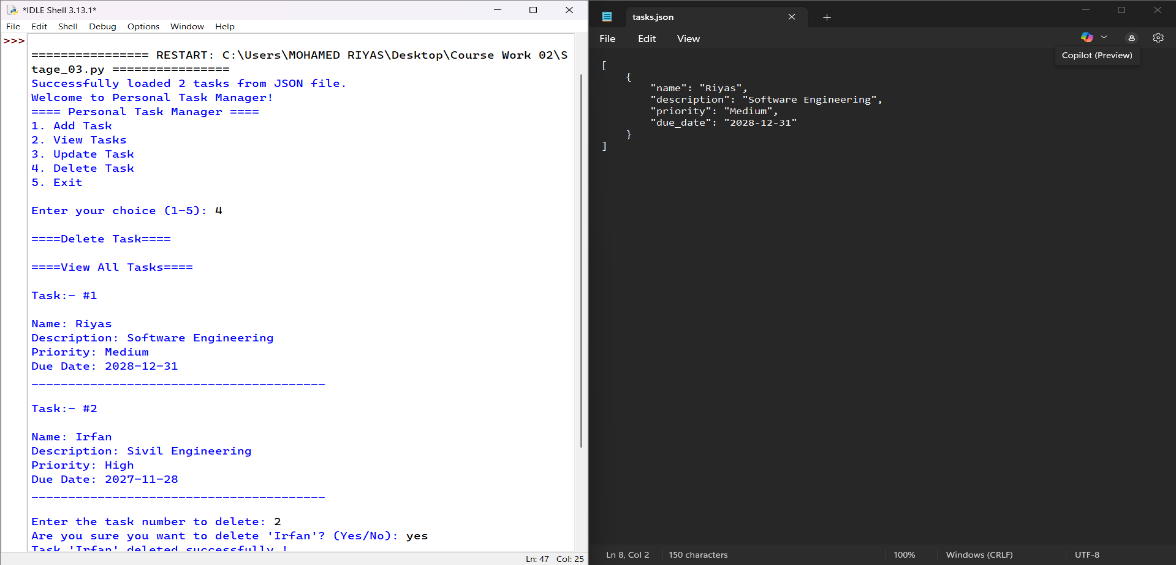
1. Test Case 3 Outcomes: Display the all task with detail correctly and **JSON file**



1. Task Case 4 Outcomes: Updated task and saved in to **JSON file**



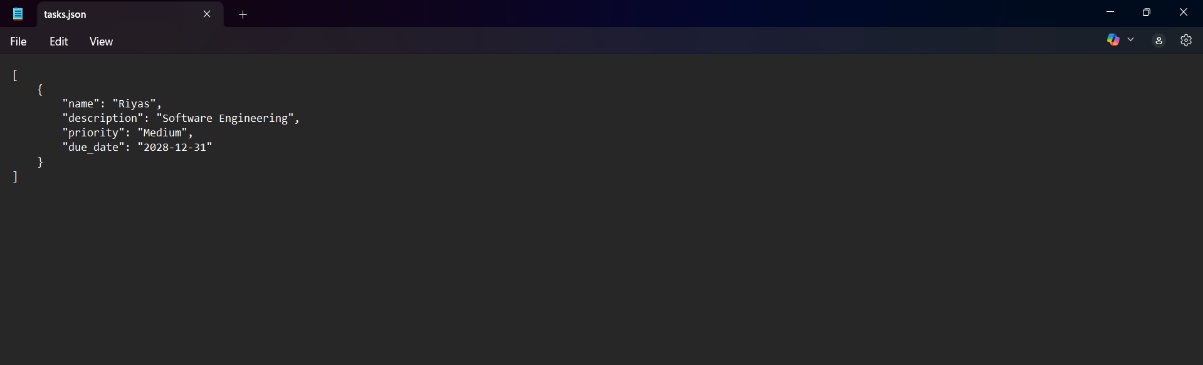
1. Test Case 5 Outcomes: Task successfully deleted from **JSON file**



1. Test Case 6 Outcomes: Task saved in tasks.json



1. Test Casse 7 Outcomes: Restart after saving tasks in tasks.json



1. **Stage 04 - Task Management System (Tkinter GUI)**
2. Introduction:

This report documents the implementation of **Stage 04**, a **Tkinter-based GUI** for the Task Management System. The system allows users to:

* **View, filter, and sort** tasks
* **Add, edit, and delete** tasks
* **Persist data** in a JSON file

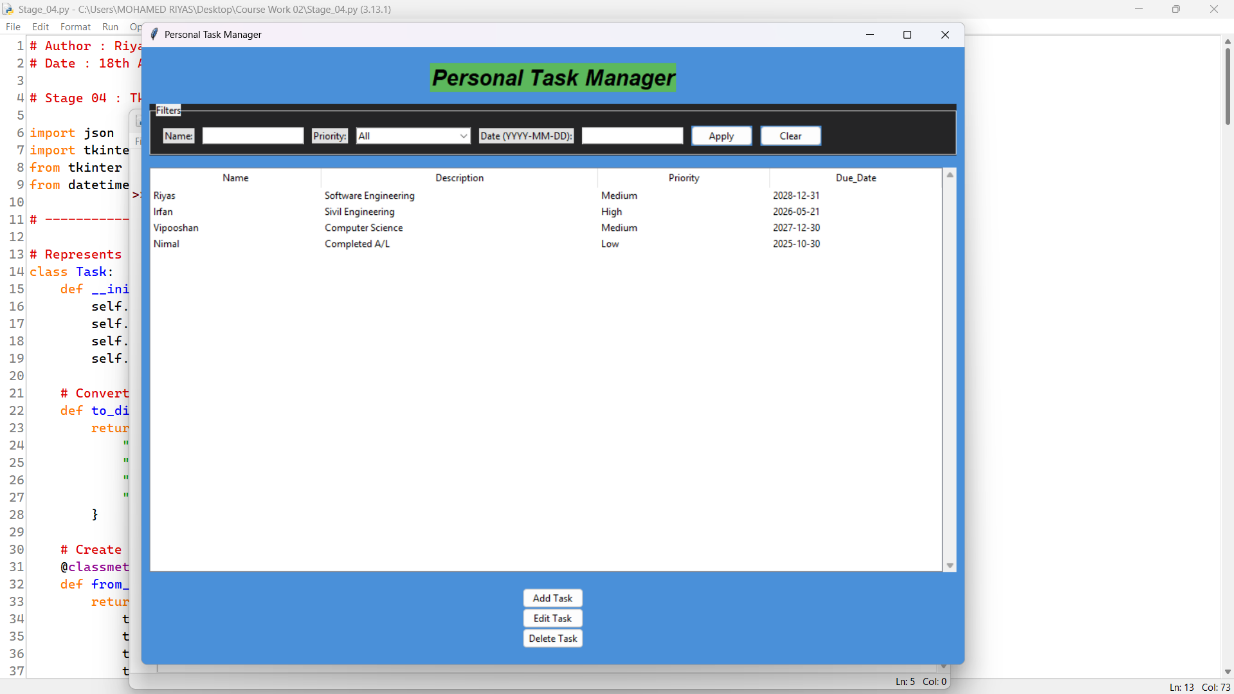
1. Features Implemented:

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Interactive Table | Displays tasks with sortable columns (Name, Description, Priority, Due Date) |
| Filtering | Filter tasks by **Name, Priority,** and **Due Date** |
| Sorting | Clickable headers sort tasks in **ascending/descending** order |
| CRUD Operation | Add, Edit, Delete tasks with validation |
| JSON Integration | Tasks saved in **tasks.json** and reloaded on restart |
| UI Design | Blue-themed interface with responsive layout |

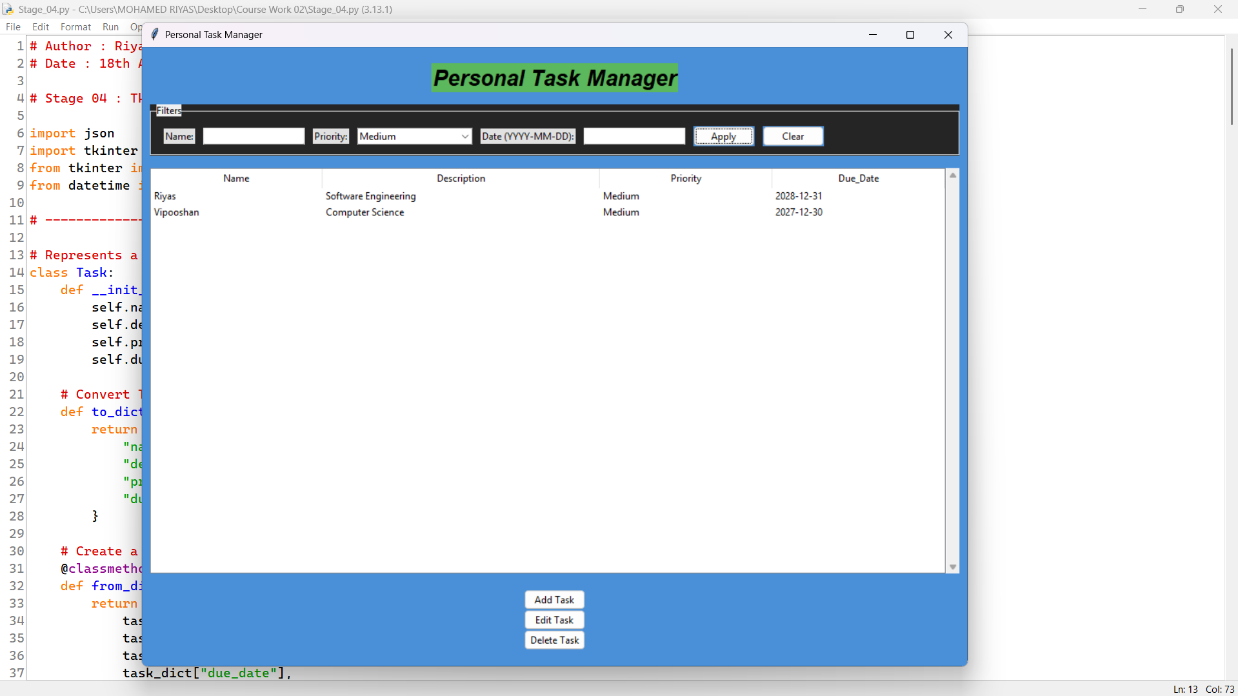
1. Test Cases & Results

| **Test ID** | **Test Scenario** | **Input** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| --- | --- | --- | --- | --- | --- |
|  | Launch Application | Run Stage\_04.py | GUI loads with all components | GUI displayed correctly | **Pass** |
|  | Filter by Priority (Medium) | Priority = "Medium" | Only medium-priority tasks shown | Filter worked as expected | **Pass** |
|  | Add New Task | Enter valid task details | Task appears in table & JSON | Task added successfully | **Pass** |
|  | Invalid Date Format | Enter "01/05/2025" (wrong format) | Error message shown | Validation triggered | **Pass** |

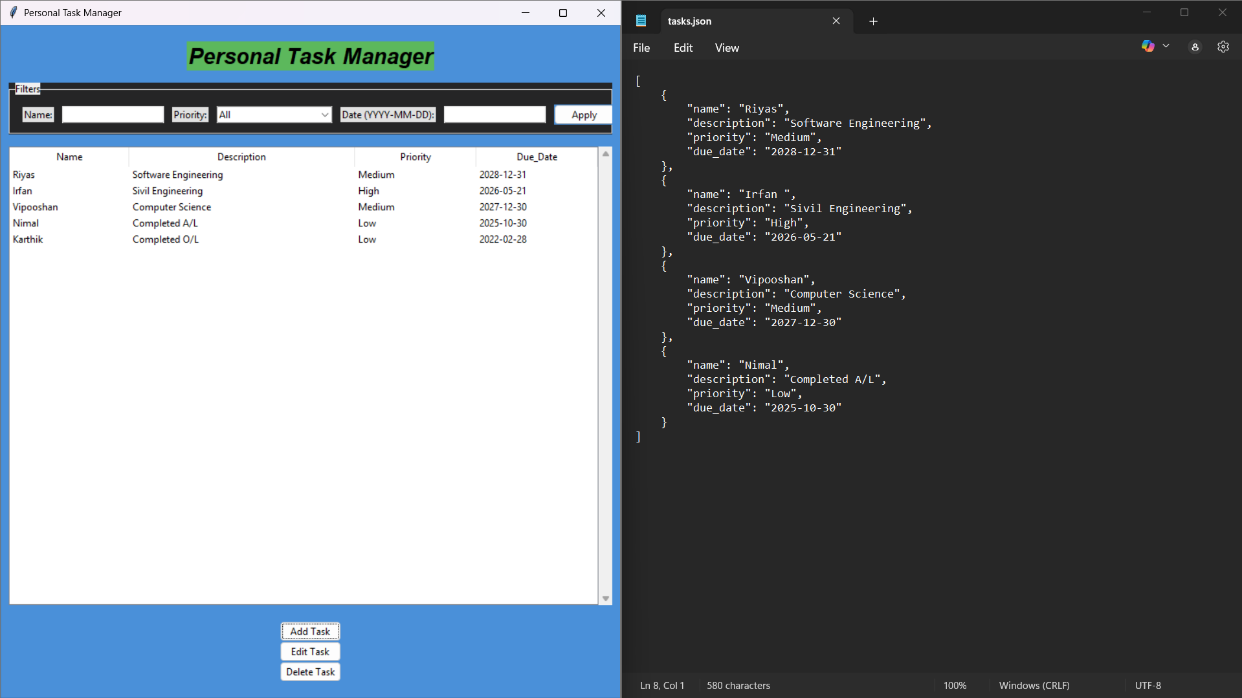
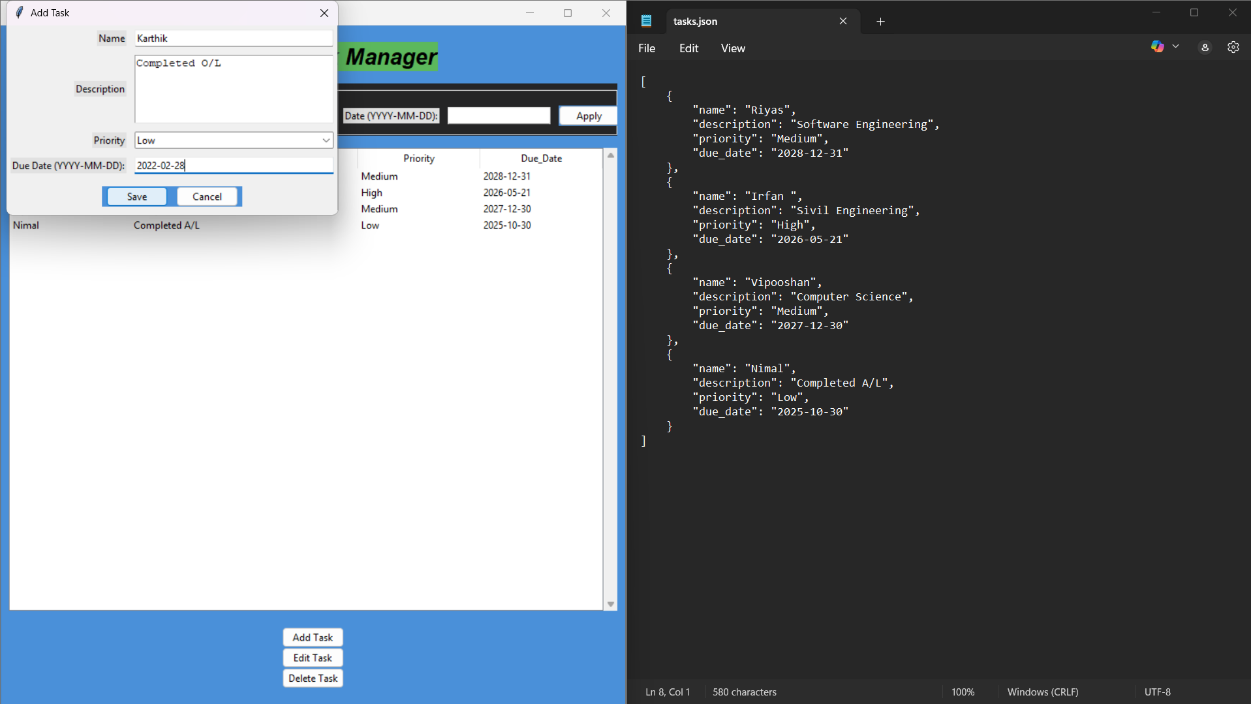
1. Figure 1: Main Interface



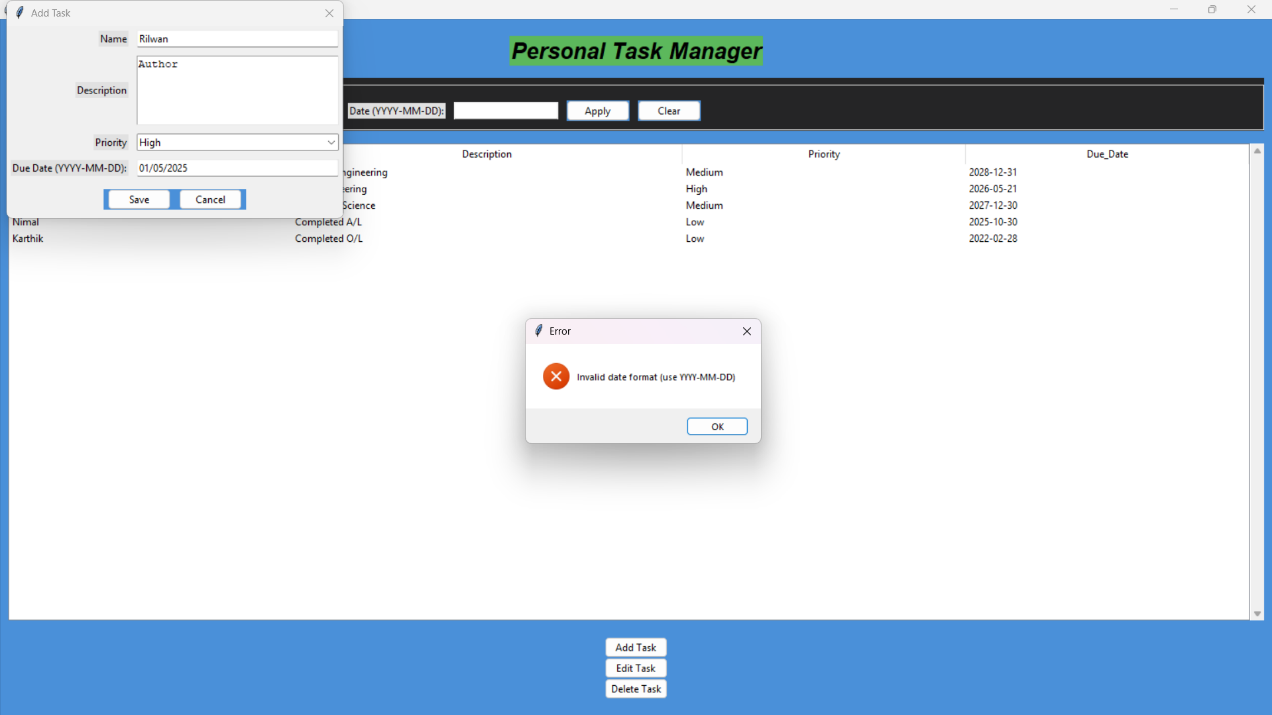
1. Figure 2: Filtering Tasks



1. Figure 3: Add New Task



1. Figure 4: Invalid Date Format

****