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 :** 01

**Submission date:** 31.03.2025

|  |  |  |
| --- | --- | --- |
| **Student\_Id** | **UOW\_ID** | **Student Name** |
| 20241795 | W2149541 | B.Vipooshan |

**School of Computing**

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

**Table of Contents**

**2`**

**Table of Contents……………………………………………………………**

**Pseudocode for Task management system program: ……………………**

**4**

**4**

**Program description: ………………………………………………..**

**Steps of the Program: ……………………………………………….**

**11 , 12**

**4**

**Tast cases ……………………………………………………………………**

**11**

**Test cases and their outcomes ….…………………………………...**

**11**

1. **Add Task Inputs…………………………………………...**

**11**

1. **Date Validation …………………………………………....**
2. **View Task………………………………………………......**

**11`**

**11`**

1. **Update Tasks…………………………………………….....**
2. **Delete Task………………………….………………………**

**12`**

1. **File Handling…………………….…………………………**

**12`**

**Test cases and their outcomes with screenshot**

1. **Figure 1 Test case………………………………………12**
2. **Figure 2 Test case………………………………………12**
3. **Figure 3 Test case………………………………………13**
4. **Figure 4 Test case………………………………………13**
5. **Figure 5 Test case………………………………………14**
6. **Figure 6 Test case………………………………………14**
7. **Figure 7 Test case………………………………………15**
8. **Figure 8 Test case………………………………………15**
9. **Figure 9 Test case………………………………………16**
10. **Figure 10 Test case……………………………………..16**
11. **Figure 11 Test case……………………………………..17**
12. **Figure 12 Test case…………………………………......17**
13. **Figure 13 Test case…………………………………......18**
14. **Figure 14 Test case…………………………………......18**
15. **Figure 15 Test case…………………………………......19**
16. **Figure 16 Test case…………………………………......19**

**Pseudocode for Task Management System**

**Program Description:**

**The Program process Task management system in python, Utilizing to store and organize tasks. The System supports basic CRUD (Create , Read , Update , Delete) operations, enabling users to add , view , update and remove tasks.**

**Pseudocode**

# Program Task Management System

# Group -- 01

# Author - - B.Vipooshan

# Date – 2025.03.25

BEGIN

DECLARE tasks AS LIST

FUNCTION add\_task()

DISPLAY blank line

DISLAY “ADD TASKS”

DISPLAY “Enter following Tasks details”

DISPLAY “Task name”

task < - INPUT

DISPLAY “Task description”

description < - INPUT

LOOP FOREVER

Create priority list to [low , medium , high] input validation

DISPLAY “Task Priority [low , medium , high]”

CONVERT priority TO LOWERCASE

IF priority IS IN [“high”, “medium”, ”low”]

BREAK LOOP

ELSE

DISPLAY “Invalid input please enter [low|medium|high] ”

END IF

END LOOP

LOOP FOREVER

DISPLAY “Task Due date (DD.MM.YYYY) [ex:- 05.05.2025]”

Due\_date < - INPUT

TRY

day < - CONVERT FIRST 2 CHARACTERS OF date TO INTEGER

month <- CONVERT CHARACTERS 4 – 6 OF date TO INTEGER

year < - CONVERT LAST 4 CHARACTERS OF date TO INTEGER

IF Year modular 400 == 0 OR Year modular 100 Not Equal 0 and year Modular 4 == 0

ELIF month in [1,3,5,7,8, , 10 , 12]:

Day\_end = 31

ELIF month in [4,6,9,11]:

Day\_end = 30

ELSE

Day\_end = 28

IF month IS BETWEEN 1 AND 12 AND day IS BETWEEN 1 AND 31 THEN

BREAK LOOP

ELSE

DISPLAY “Invalid date format. Please use DD.MM.YYY this format”

END IF

CATCH VALUE ERROR

DISPLAY “Invalid date format. Please use DD.MM.YYY this format”

END TRY

END LOOP

APPEND [task, description, priority, due\_date] TO tasks

DISPLAY “\* Your Task “+task+” has been added successfully !”

END FUNCTION

FUNCTION view\_tasks()

DISPLAY blank line

DISLAY “VIEW TASKS”

IF tasks

FOR each length of tasks

DISPLAY “Tasks{number}”

DISPLAY task details (task name, task description, priority , due\_date)

END FOR

ELSE

DISPLAY “No Tasks available”

END IF

END FUNCTION

FUNCTION update\_task()

DISPLAY blank line

DISPLAY “ UPDATE TASK ”

IF tasks THEN

TRY

DISPLAY “Enter the task number to update”

Update task number < - INPUT

SUBTRACTION TASK NUMBER stored in variable name update\_task\_id

IF update\_task\_id BETWEEN 0 and length of task list THEN

DISPLAY “Your Current Task IS ”

Name:

Description:

Priority:

Due Date:

DISPLAY “Press enter to keep current value if you don’t change it”

DISPLAY “Enter Task name” OR KEEP CURRENT VALUE

task name < - INPUT

DISPLAY “Enter Task description” OR KEEP CURRENT VALUE

Description < - INPUT

Again Create list name as priority list low , medium , high

LOOP FOREVER

DISPLAY “Enter the priority (current Value)”

IF NOT in priority list

priority < - tasks[update\_task\_id][2]

BREAK LOOP

ELIF LOWER CASE priority in priority list

priority < - new\_priority.lower()

ELSE

DISPLAY "Invalid priority! Please enter'High', ‘Low',or'Medium

END IF

END LOOP

LOOP FOREVER

DISPLAY “Task Due date (DD.MM.YYYY) [ex 05.05.2025]”

TRY

day < - CONVERT FIRST 2 CHARACTERS OF date TO INTEGER

month < -CONVERT FIRST 3-5 CHARACTERS date TO INTEGER

year <- CONVERT FIRST 6 -10 CHARACTERS date TO INTEGER

IF Year modular 400 == 0 OR Year modular 100 Not Equal 0 and year Modular 4 == 0

ELIF month in [1,3,5,7,8, , 10 , 12]:

Day\_end = 31

ELIF month in [4,6,9,11]:

Day\_end = 30

ELSE

Day\_end = 28

IF day BETWEEN 1 and 31 AND month BETWEEN 1 and 12 THEN

BREAK LOOP

ELSE

DISPLAY “Invalid date format please use DD.MM.YYYY ”

END IF

CATCH VALUE ERROR

DISPLAY “Invalid date format please use DD.MM.YYYY”

Tasks[update\_task\_id] <- [task\_\_name , description , priority , due\_date]

DISPLAY “TASK UPDATED SUCESSFULLY ”

ELSE

DISPLAY “Invalid task number”

CATCH VALUE ERROR

DISPLAY “Invalid inputs please enter the number”

ElSE

DISPLAY “No tasks to update Available”

FUNCTION delete\_tasK()

DISPLAY blank line

DISPLAY “DELETE TASK”

IF tasks THEN

TRY

DISPLAY “Enter task number to delete ”

Get\_user3task\_id < - INPUT

IF delete\_task BETWEEN 0 and length of tasks list THEN

Delete tasks[del\_task]

DISPLAY “TASK DELETED SUCUESSFULLY ”

ELSE  
 DISPLAY “Invalid task number ”

CATCH VALUE ERROR

DISPLAY “Invalid task number”

ELSE

DISPLAY “No task to delete”

DISPLAY “Task Management system ”

WHILE TRUE

DISPLAY blank line

DISPLAY “01. Add\_task”

DISPLAY “02. View\_task”.

DISPLAY ” 03. Update\_task”

DISPLAY “04.delete \_task ”

DISPLAY “05. Exit program”

TRY

DISPLAY “Enter the option(1-5)”

User\_choice < - INPUT

IF user\_choice Between 1 and 2 1 THEN

IF user\_chocie == 1

Call add\_task FUNCTION

ELIF user\_choice == 2

Call view\_task FUNCTION

ELIF user\_choice == 4

Call update\_task FUNCTION

ELIF user\_chioce == 5:

Call delete\_task FUNCTION

THEN DISAPLAY “Exiting The Program Thank you. !

BREAK

ELSE

DISAPLAY “Invalid please enter the number(1-5) ”

EXCEPT VALUE ERROR

DISPLAY “Invalid input please enter a number”

END

***Test Cases***

**Test cases and their outcomes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Input** | **Expected Outcome** | **Description** | **Actual Outcome** | **Result**  **(Pass/Fail)** |
| **Add Tasks Inputs** | | | | | |
| TC\_ 001 | Task Name: " " | “Error! Please Enter your Task name” | Rejects Empty Task name | “Error! Please Enter your Task name” | Pass |
| TC\_ 002 | Valid Input  Task | “Task Added Succuessfully” | Add the task with input validation | “Task Added Succuessfully” | Pass |
| TC\_ 003 | Task Description: " " | “Error! Please Enter your Task description”` | Rejects Empty Task description | “Error! Please Enter your Task description” | Pass |
| TC\_ 004 | Priority: "Loe" | Invalid Input please enter [low, medium, high]  message | Rejects unlisted priority | Invalid Input please enter [low, medium, high] | Pass |
| **Date Validation** | | | | | |
| TC\_ 005 | 13/03/2023 | Invalid date format. Please use DD.MM.YYYY this format | Rejects( YYYY / MM/ DD) date with slashes | Invalid date format. Please use DD.MM.YYYY this format | Pass |
| TC\_ 006 | 30.02.2023 | Invalid inputs. Please enter between 1-28 for month 2. | Invalid  Date – Feb 30 | Invalid inputs. Please enter between 1-28 for month 2. | Pass |
| **View Tasks** | | | | | |
| TC\_ 007 | No Tasks | No Tasks Available | Handles no-task scenario | "No tasks available" show | Pass |
| TC\_ 008 | Task Exist | Show all tasks | Show all task with details correctly | Task listed as expected | Pass |
| **Update Tasks** | | | | | |
| TC\_ 009 | Update All task 1 for Field | Task Updated Succuessfully | Verifies full update | “Your Task updated Successfully” | Pass |
| TC\_ 010 | Task index: 111 | Error for Task index: 111 | Invalid task index | “Invalid Task number” | Pass |
| **Delete Tasks** | | | | | |
| TC\_ 011 | Task index: 999 | Error for Task index: 999 | Invalid task index | “Invalid Task number” | Pass |
| TC\_ 012 | Task Index 1 | Task deleted Succuessfully | Valid deletion | “Task deleted Successfully” | Pass |
| **File Handling** | | | | | |
| TC\_ 013 | Save tasks with valid inputs | Data saved in tasks.txt | Ensures file handling write works | Task saved in text.txt | Pass |
| TC\_ 014 | Restart after saving tasks | Tasks loaded successfully view task | Loads from file if present | Except tasks are display | Pass |
| TC\_ 015 | Delete tasks.txt before run | Empty task list, no crash | Handles missing file | “No tasks to delete” | Pass |
| TC\_ 016 | Update tasks.txt before run | Empty task list, no crash | Handles missing file | “No tasks to update” | Pass |

`

**Figure 1 Test case TC\_001**

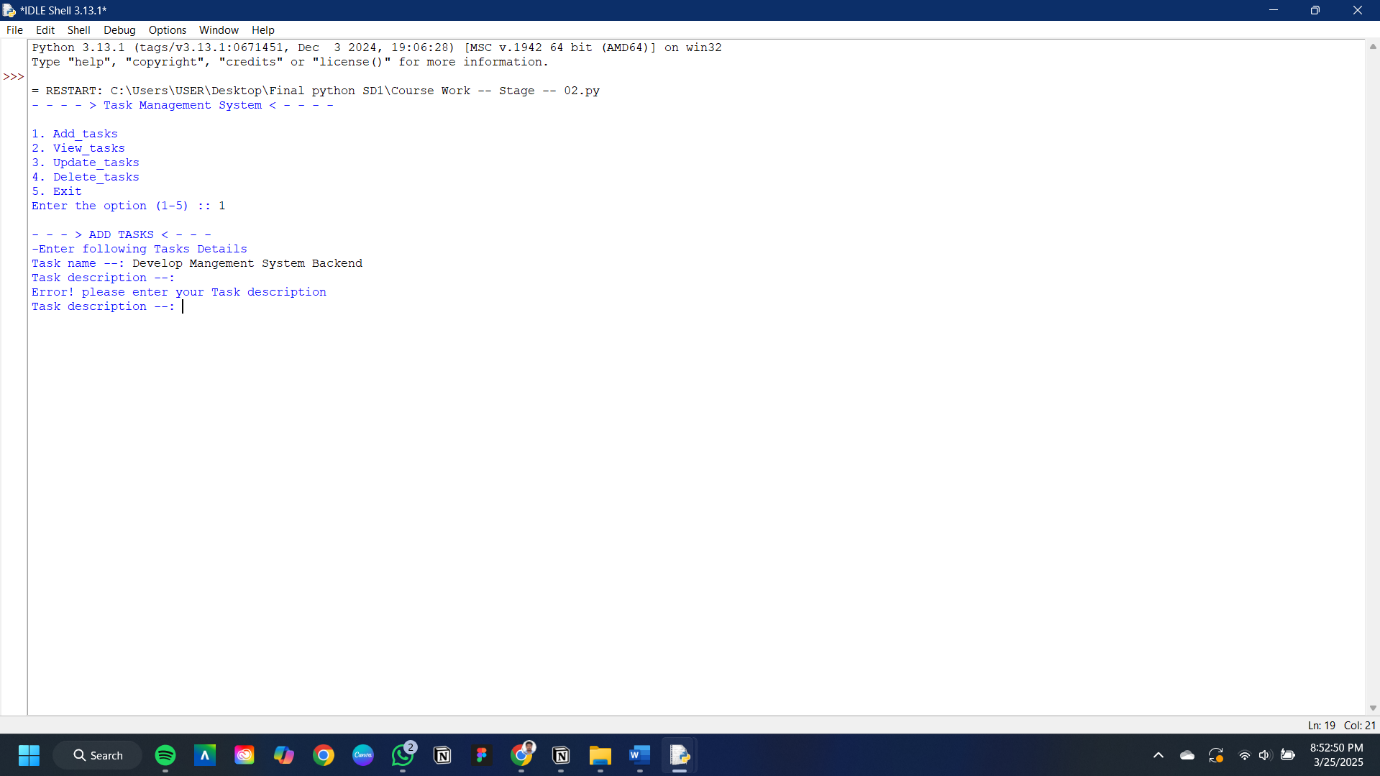
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

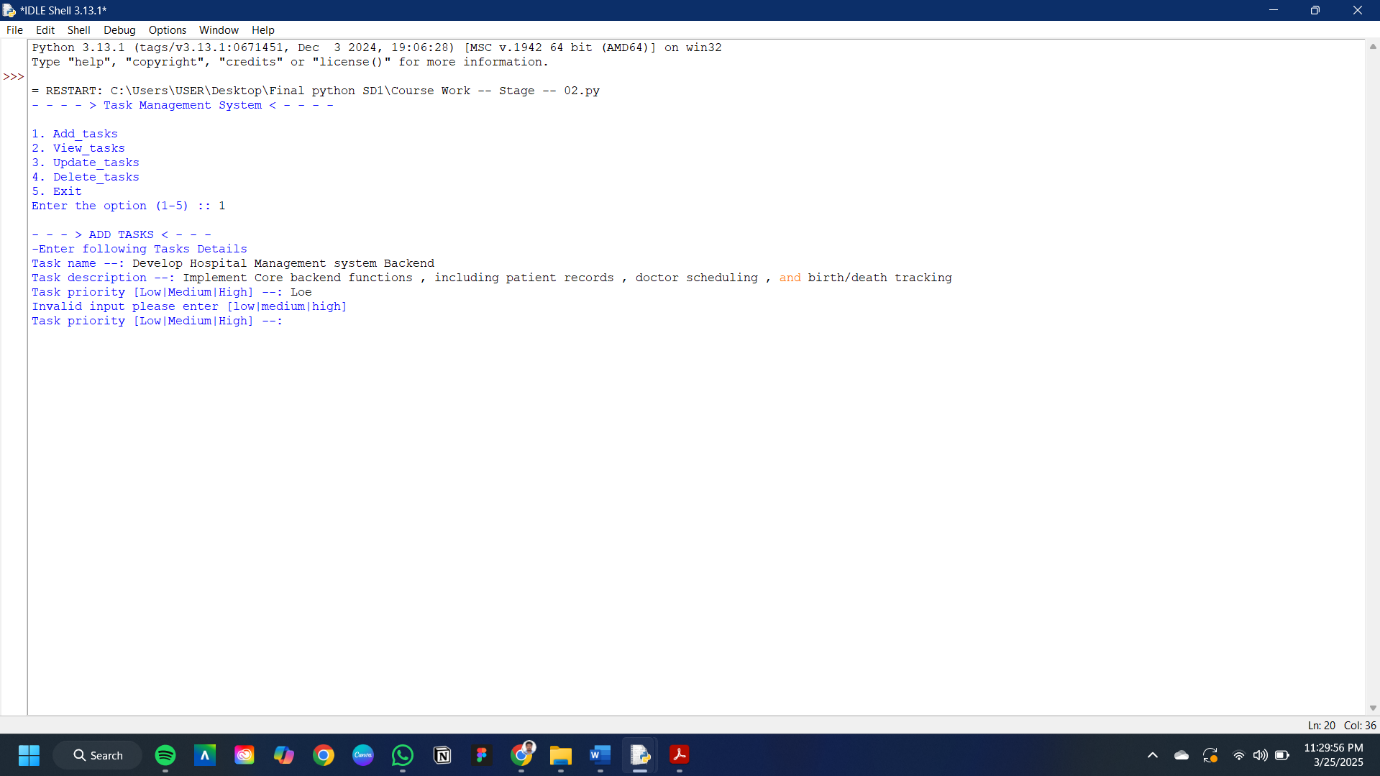
AI-generated content may be incorrect.

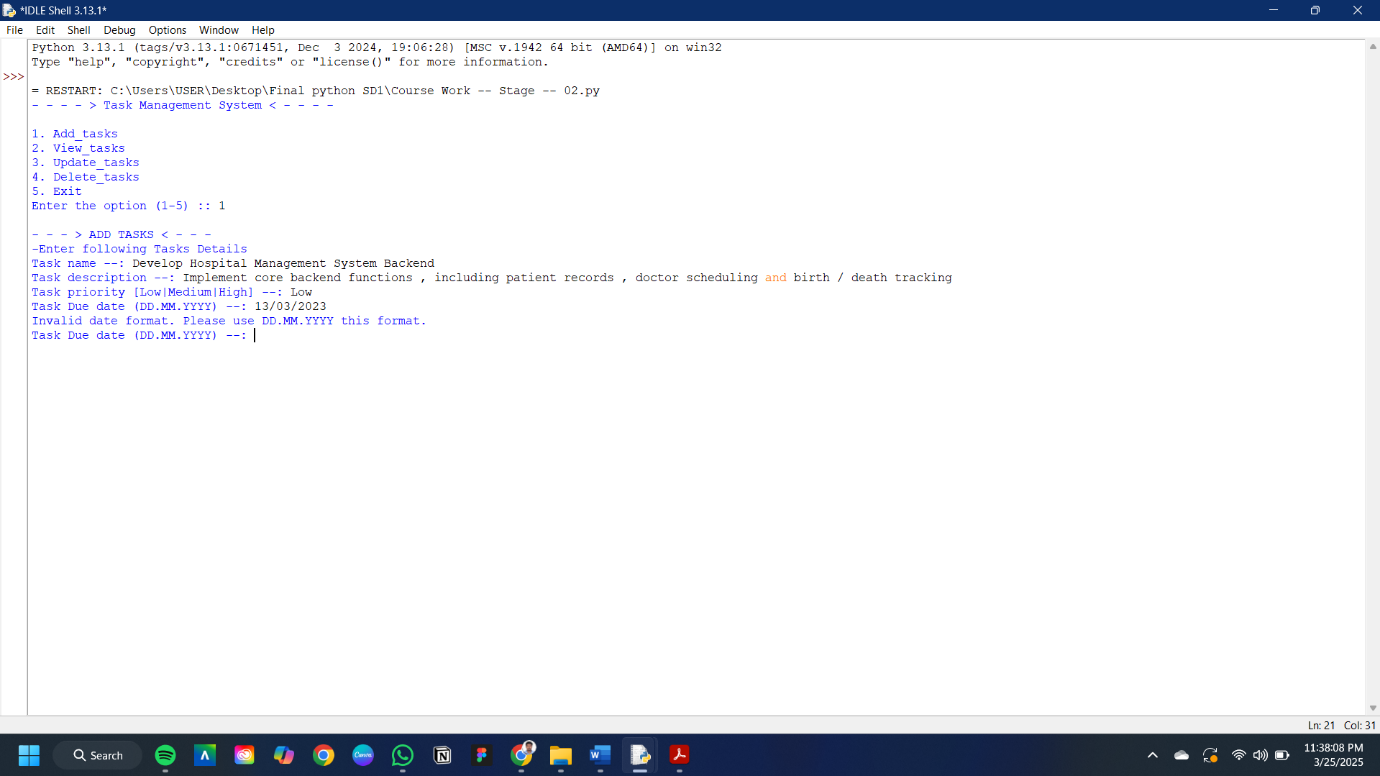
**Figure 2 Test case TC\_002**



**Figure 4 Test case TC\_004**

**Figure 3 Test case TC\_003**



A screenshot of a computer

AI-generated content may be incorrect.

**Figure 6 Test case TC\_006**

**Figure 5 Test case TC\_005**

A screenshot of a computer

AI-generated content may be incorrect.

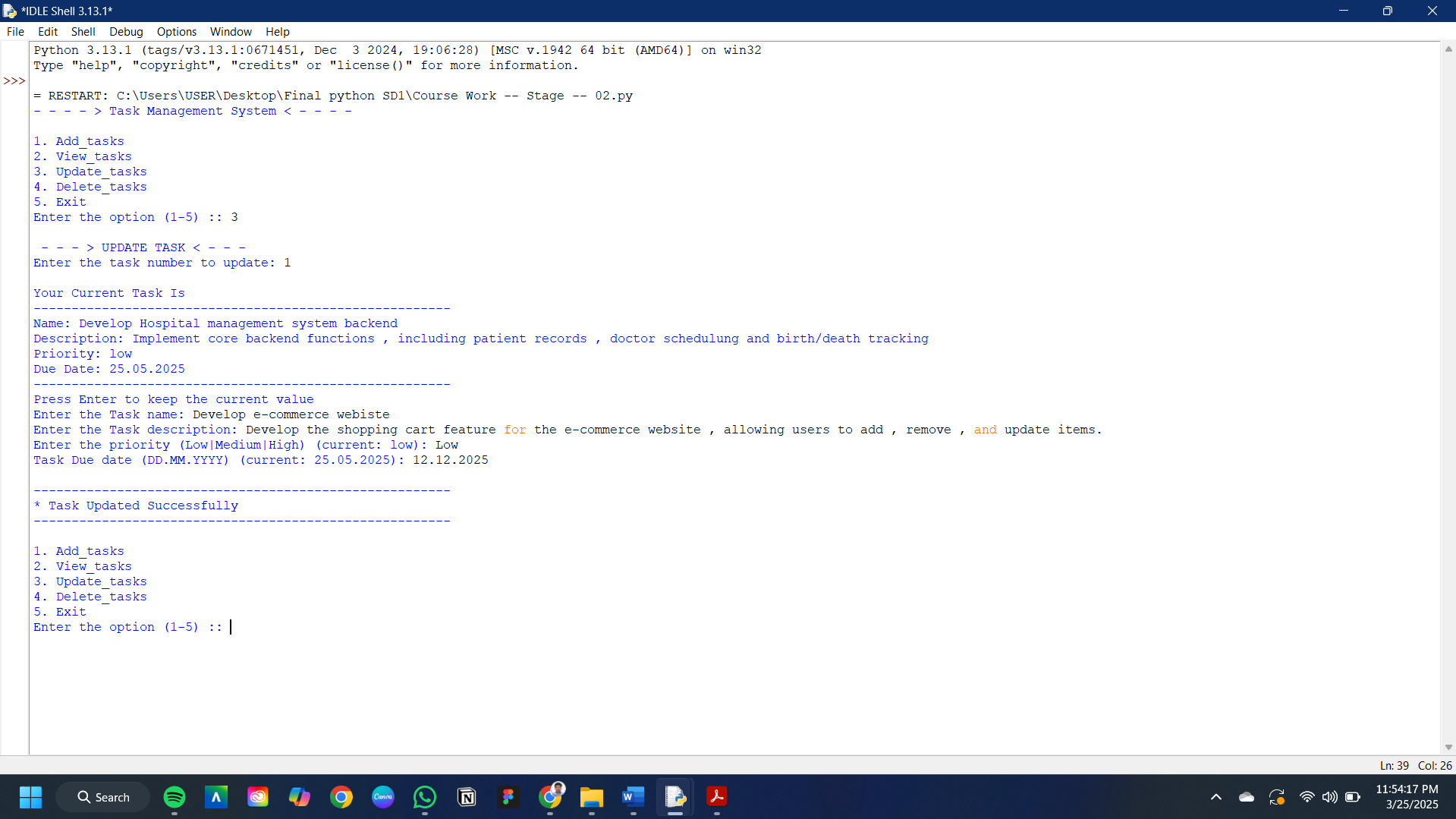
**Figure 8 Test case TC\_008**

**Figure 7 Test case TC\_007**

A screenshot of a computer

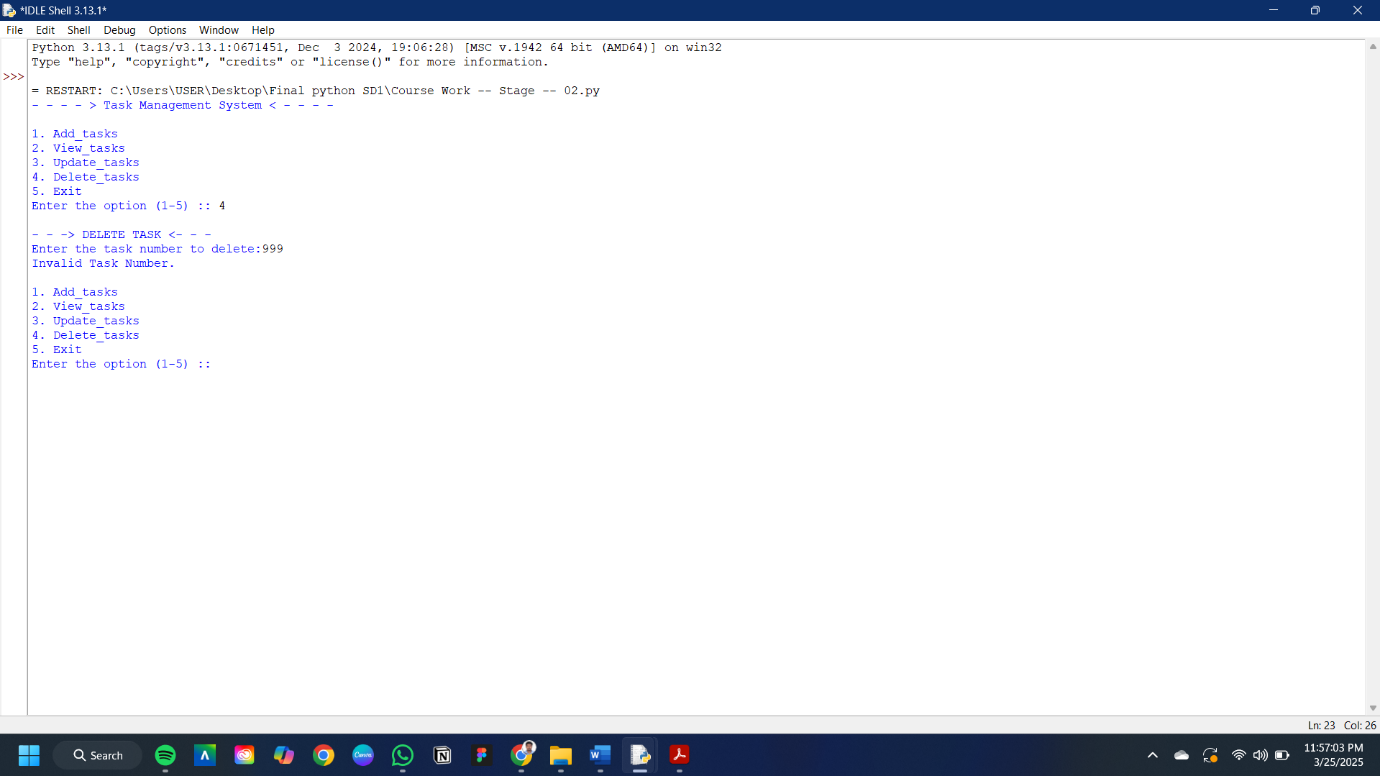
AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**Figure 10 Test case TC\_010**

**Figure 9 Test case TC\_009**

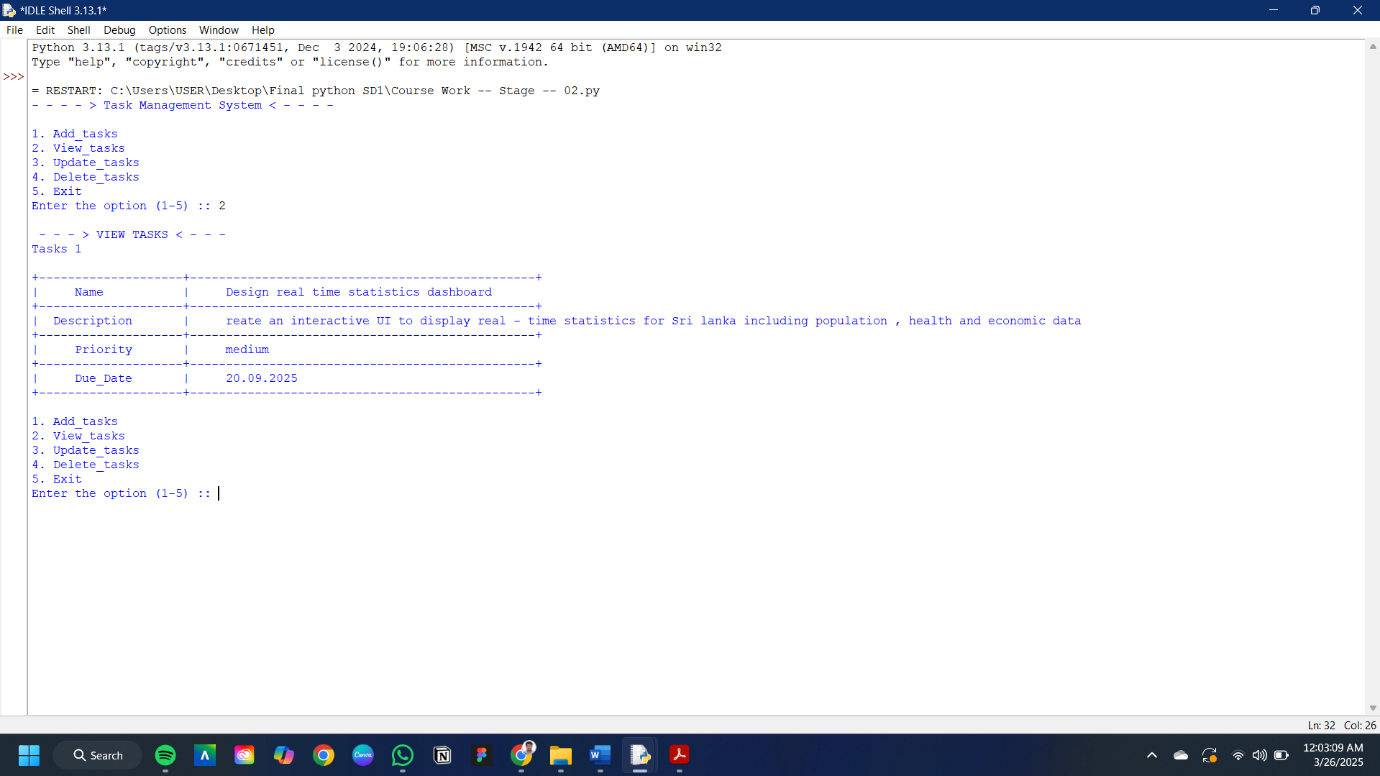
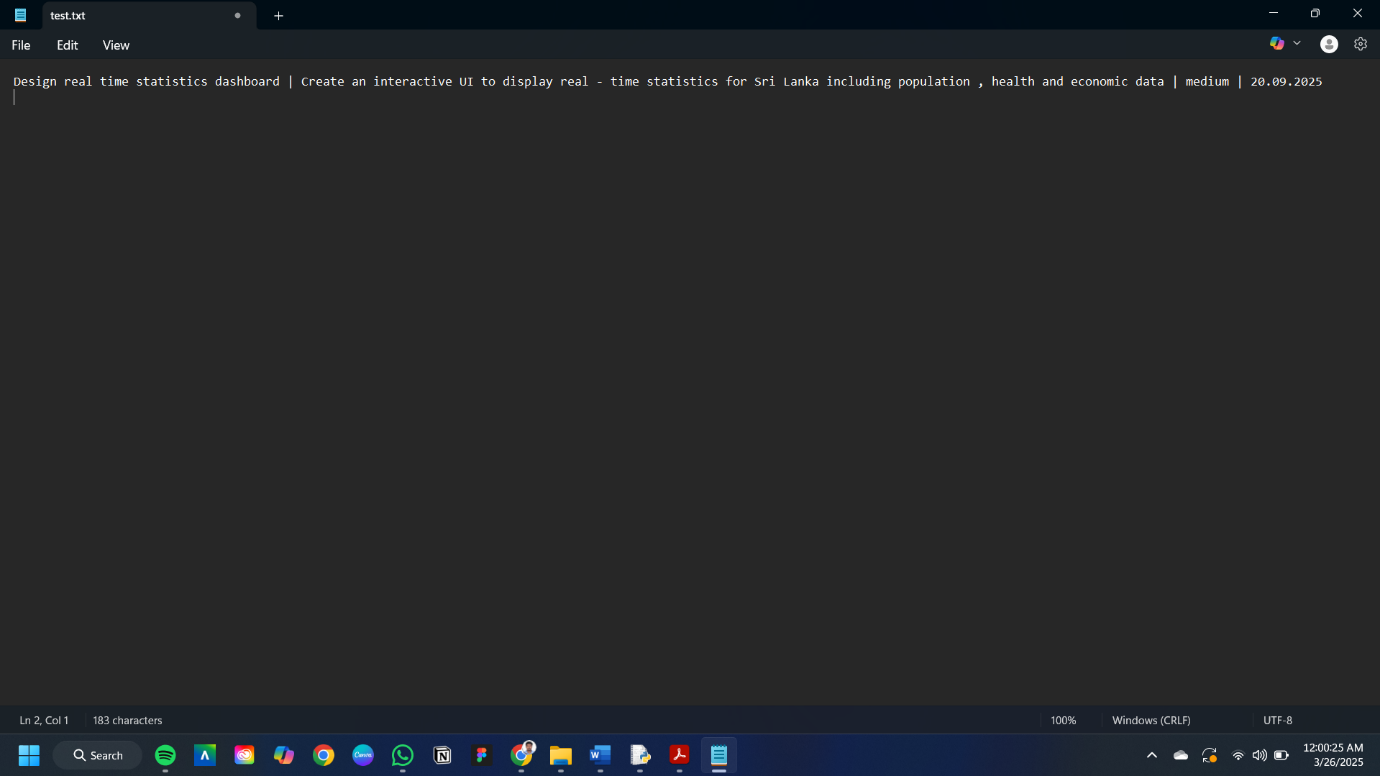


**Figure 11 Test case TC\_011**

A screenshot of a computer

AI-generated content may be incorrect.

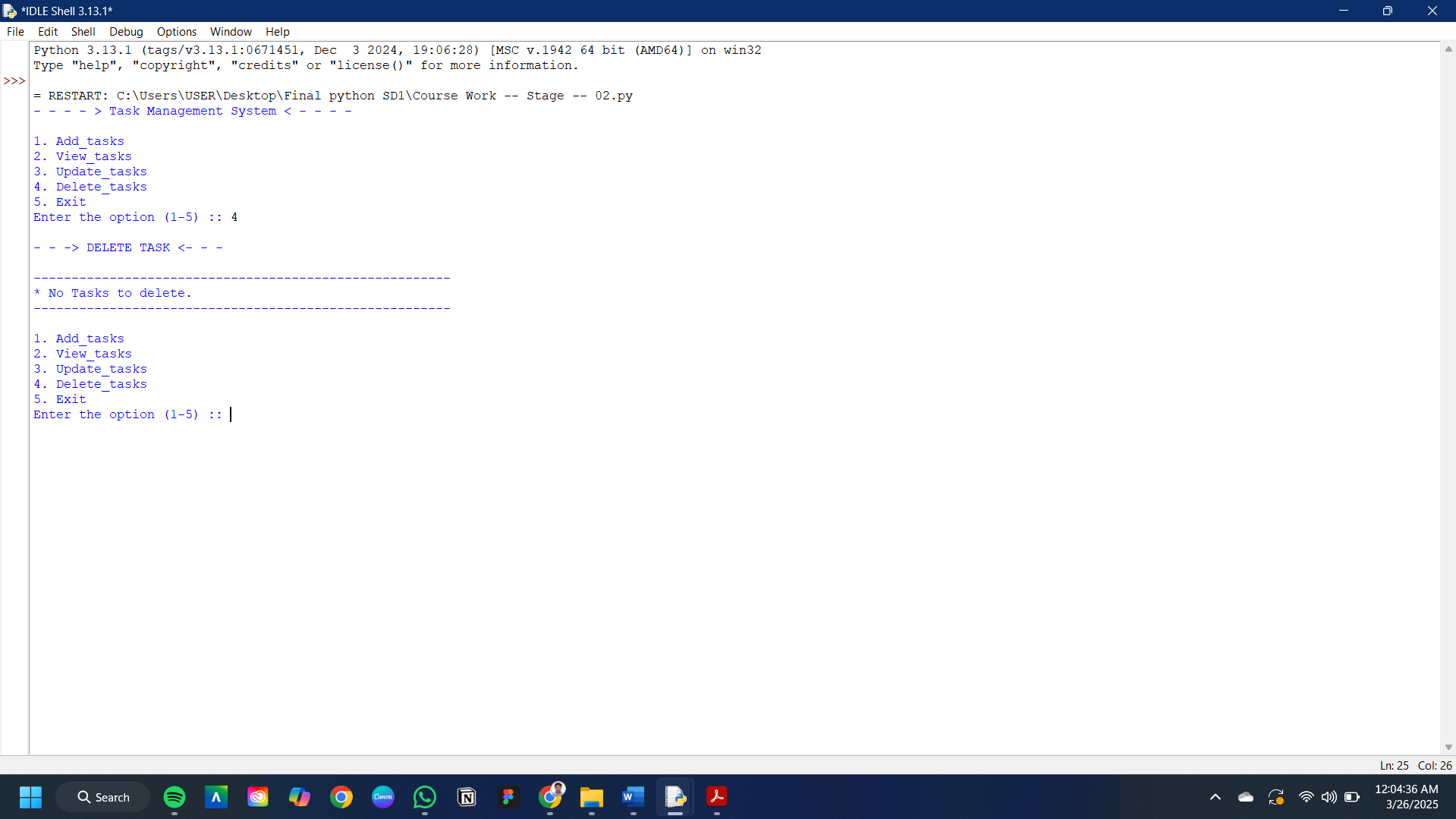
**Figure 12 Test case TC\_012**



**Figure 14 Test case TC\_014**

**Figure 13 Test case TC\_013**

A screenshot of a computer

AI-generated content may be incorrect.

**Figure 16 Test case TC\_016**

**Figure 15 Test case TC\_015**