



INFORMATICS
INSTITUTE OF
TECHNOLOGY

UNIVERSITY OF
WESTMINSTER[®]

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

Table of Contents.....	2
Pseudocode for Task management system program:	4
Program description:	4
Steps of the Program:	4
Tast cases	11 , 12
Test cases and their outcomes	11
1. Add Task Inputs.....	11
2. Date Validation	11
3. View Task.....	11
4. Update Tasks.....	11
5. Delete Task.....	12
6. File Handling.....	12
Test cases and their outcomes with screenshot	
I. Figure 1 Test case.....	12
II. Figure 2 Test case.....	12
III. Figure 3 Test case.....	13
IV. Figure 4 Test case.....	13
V. Figure 5 Test case.....	14
VI. Figure 6 Test case.....	14
VII. Figure 7 Test case.....	15
VIII. Figure 8 Test case.....	15
IX. Figure 9 Test case.....	16
X. Figure 10 Test case.....	16
XI. Figure 11 Test case.....	17
XII. Figure 12 Test case.....	17
XIII. Figure 13 Test case.....	18
XIV. Figure 14 Test case.....	18
XV. Figure 15 Test case.....	19
XVI. 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 DISPLAY "Exiting The Program Thank you. !"
        BREAK
    ELSE
        DISPLAY "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

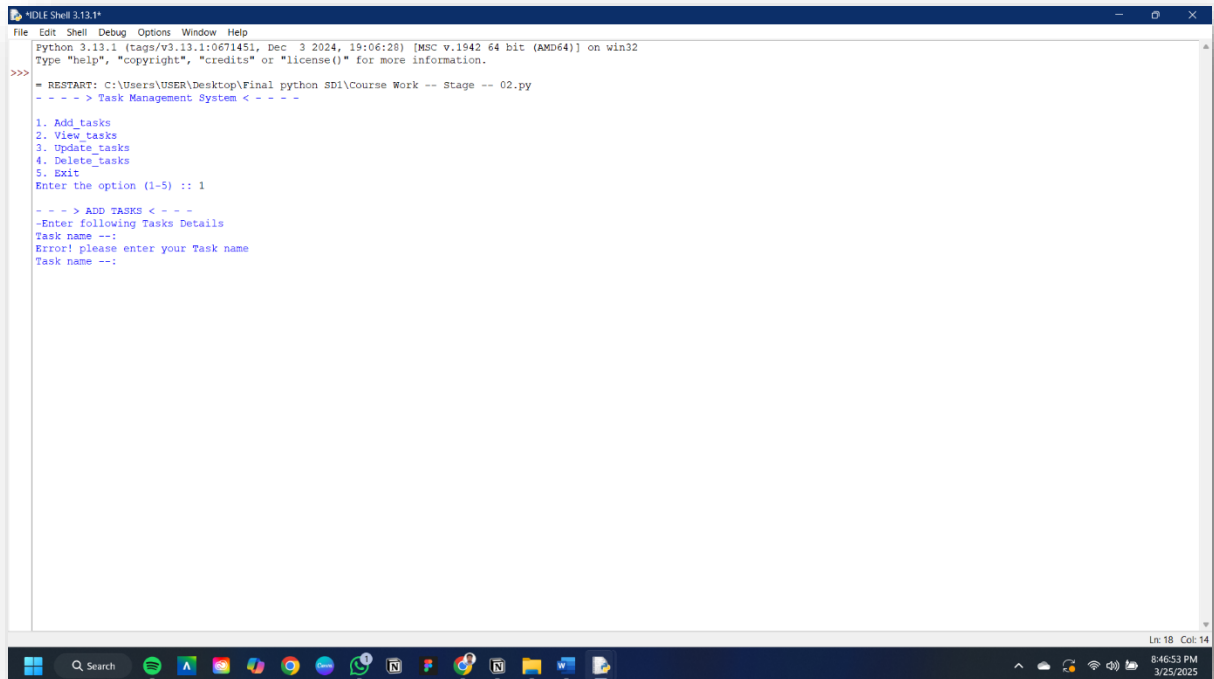
Test Case ID	Input	Expected Outcome	Description	Actual Outcome	Result (Pass/Fail)
Add Tasks Inputs					
TC_001	Task Name: " "	"Error! Please Enter your Task name"	Rejects Empty Task name	"Error! Please Enter your Task name"	Pass
TC_002	Valid Input Task	"Task Added Succuessfully"	Add the task with input 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 Succuessfully"	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

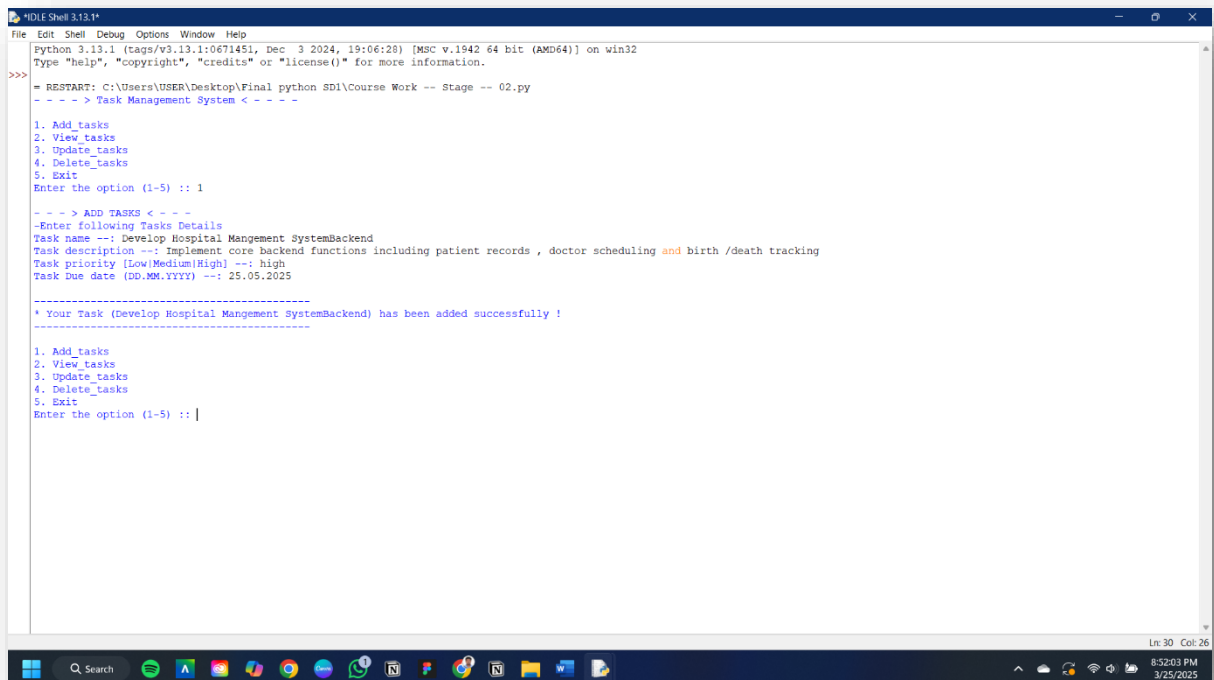
File Handling

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



```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - > Task Management System < - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - > ADD TASKS < - - -
-Enter following Tasks Details
Task name --:
Error! please enter your Task name
Task name --:
```

Figure 1 Test case TC_001



```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - > Task Management System < - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - > ADD TASKS < - - -
-Enter following Tasks Details
Task name --: Develop Hospital Management SystemBackend
Task description --: Implement core backend functions including patient records , doctor scheduling and birth /death tracking
Task priority [Low|Medium|High] --: high
Task Due date (DD.MM.YYYY) --: 25.05.2025
-----
* Your Task (Develop Hospital Management SystemBackend) has been added successfully !
-----
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 2 Test case TC_002

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SDI\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - -> ADD TASKS < - - - -
-Enter following Tasks Details
Task name --: Develop Mangement System Backend
Task description --:
Error! please enter your Task description
Task description --: |
```

Figure 3 Test case TC_003

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SDI\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - -> ADD TASKS < - - - -
-Enter following Tasks Details
Task name --: Develop Hospital Management system Backend
Task description --: Implement Core backend functions , including patient records , doctor scheduling , and birth/death tracking
Task priority [Low|Medium|High] --: Low
Invalid input please enter [Low|medium|high]
Task priority [Low|Medium|High] --:
```

Figure 4 Test case TC_004

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - - -> Task Management System < - - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - - -> ADD TASKS < - - - - -
-Enter following Tasks Details
Task name --: Develop Hospital Management System Backend
Task description --: Implement core backend functions , including patient records , doctor scheduling and birth / death tracking
Task priority [Low|Medium|High] --: Low
Task Due date (DD.MM.YYYY) --: 13/03/2023
Invalid date format. Please use DD.MM.YYYY this format.
Task Due date (DD.MM.YYYY) --: |
```

Figure 5 Test case TC_005

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - - -> Task Management System < - - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 1
- - - - -> ADD TASKS < - - - - -
-Enter following Tasks Details
Task name --: Develop Hospital Management system backend
Task description --: Implement core backend functions , including patient records , doctor scheduling and birth / death tracking
Task priority [Low|Medium|High] --: High
Task Due date (DD.MM.YYYY) --: 30.02.2024
Invalid inputs. Please enter between 1-29 for month 2.
Task Due date (DD.MM.YYYY) --: |
```

Figure 6 Test case TC_006

```

Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
* RESTART: C:\Users\USER\Desktop\Final python SD\Course Work -- Stage -- 02.py
- - - > Task Management System < - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 2
- - - > VIEW TASKS < - - -
* No Tasks available
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |

```

Figure 7 Test case TC_007

```

-Enter following Tasks Details
Task name --: Design real time statistics dashboard
Task Description --: Create an interactive UI to display real - time statistics for stilanka including population , health and economic data
Task priority [Low/Medium/High] --: Medium
Task Due date (DD.MM.YYYY) --: 20.09.2025

* Your Task (Design real time statistics dashboard) has been added successfully !

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 2
- - - > VIEW TASKS < - - -
Tasks 1
+-----+-----+
| Name          | Develop Hospital management system backend |
+-----+-----+
| Description    | Implement core backend functions , including patient records , doctor scheduling and birth/death tracking |
+-----+-----+
| Priority       | Low                                         |
+-----+-----+
| Due_Date      | 25.05.2025                                |
+-----+-----+
Tasks 2
+-----+-----+
| Name          | Design real time statistics dashboard       |
+-----+-----+
| Description    | Create an interactive UI to display real - time statistics for stilanka including population , health and economic data |
+-----+-----+
| Priority       | medium                                     |
+-----+-----+
| Due_Date      | 20.09.2025                                |
+-----+-----+

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |

```

Figure 8 Test case TC_008

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 3

- - - -> UPDATE TASK < - - - -
Enter the task number to update: 1

Your Current Task Is
-----
Name: Develop Hospital management system backend
Description: Implement core backend functions , including patient records , doctor scheduling and birth/death tracking
Priority: low
Due Date: 25.05.2025
-----
Press Enter to keep the current value
Enter the Task name: Develop e-commerce website
Enter the Task description: Develop the shopping cart feature for the e-commerce website , allowing users to add , remove , and update items.
Enter the priority (Low/Medium/High) (current: low): Low
Task Due date (DD.MM.YYYY) (current: 25.05.2025): 12.12.2025

-----
* Task Updated Successfully
-----

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 9 Test case TC_009

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 3

- - - -> UPDATE TASK < - - - -
Enter the task number to update: 111
Invalid Task number

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 10 Test case TC_010


```
"IDLE Shell 3.13.1"
File Edit Shell Debug Options Window Help
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 4
- - - -> DELETE TASK < - - -
Enter the task number to delete:999
Invalid Task Number.
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) ::
```

Figure 11 Test case TC_011

```
"IDLE Shell 3.13.1"
File Edit Shell Debug Options Window Help
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SD1\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 4
- - - -> DELETE TASK < - - -
Enter the task number to delete:1
-----
*Task Deleted Successfully !
-----
1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 12 Test case TC_012

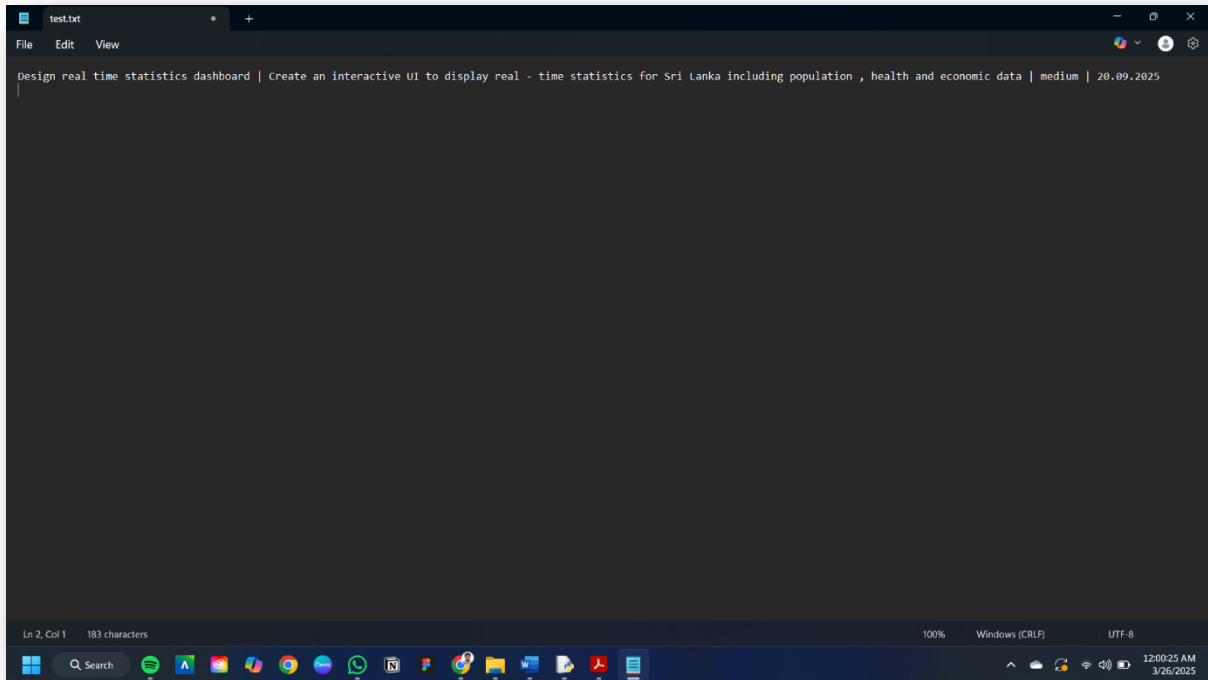


Figure 13 Test case TC_013

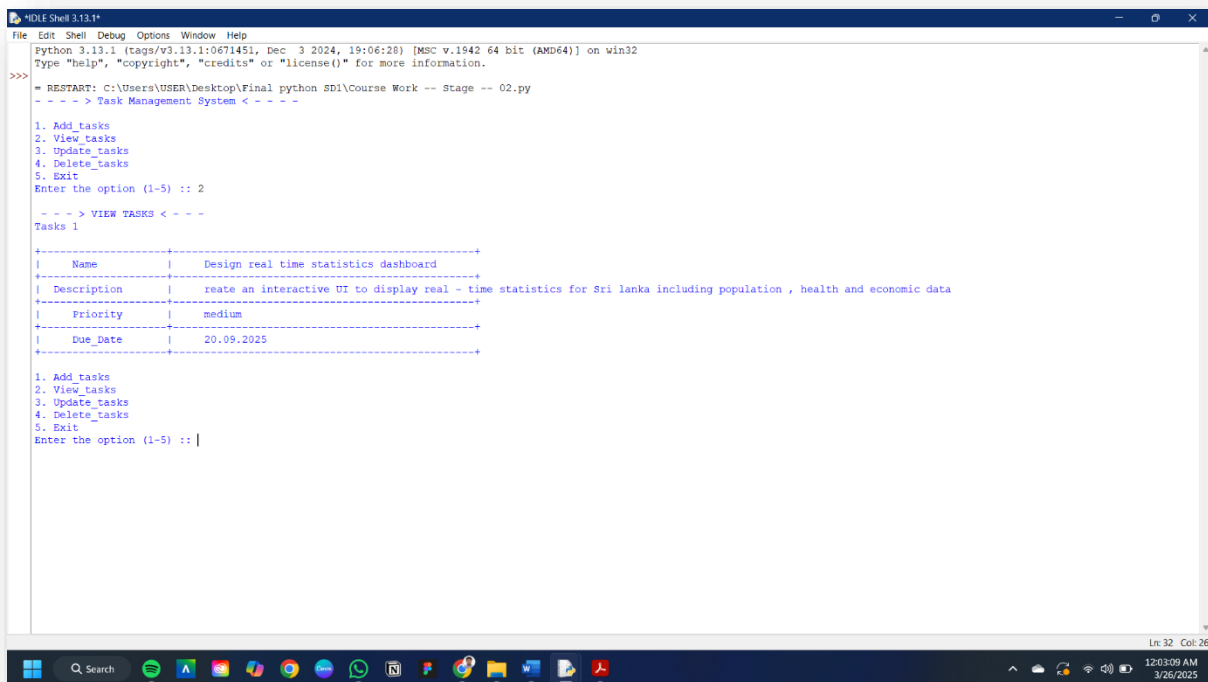


Figure 14 Test case TC_014

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SDI\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 4
- - - -> DELETE TASK < - - - -

* No Tasks to delete.

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 15 Test case TC_015

```
Python 3.13.1 (tags/v3.13.1:0671451, Dec 3 2024, 19:06:28) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\USER\Desktop\Final python SDI\Course Work -- Stage -- 02.py
- - - -> Task Management System < - - - -

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 4
- - - -> DELETE TASK < - - - -

* No Tasks to delete.

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: 3
- - - -> UPDATE TASK < - - - -

* No Tasks available to Update

1. Add_tasks
2. View_tasks
3. Update_tasks
4. Delete_tasks
5. Exit
Enter the option (1-5) :: |
```

Figure 16 Test case TC_016