Mini Project

Command-Line To-Do List Manager

Github Link:

https://github.com/UnnamDedeepya/Command-Line-to-do-list

Source code:

```
import json
from datetime import datetime, timedelta

# File where tasks will be saved
TASKS_FILE = 'tasks1.json'

# Priority levels
PRIORITY_LEVELS = ['low', 'medium', 'high']


# Load tasks from file
def load_tasks():
    try:
        with open(TASKS_FILE, 'r') as file:
            Tasks1 = json.load(file)
    except (FileNotFoundError, json.JSONDecodeError):
        Tasks1 = []
    return tasks1


# Save tasks to file
def save_tasks(tasks1):
    with open(TASKS_FILE, 'w') as file:
        json.dump(tasks1, file, indent=4)
```

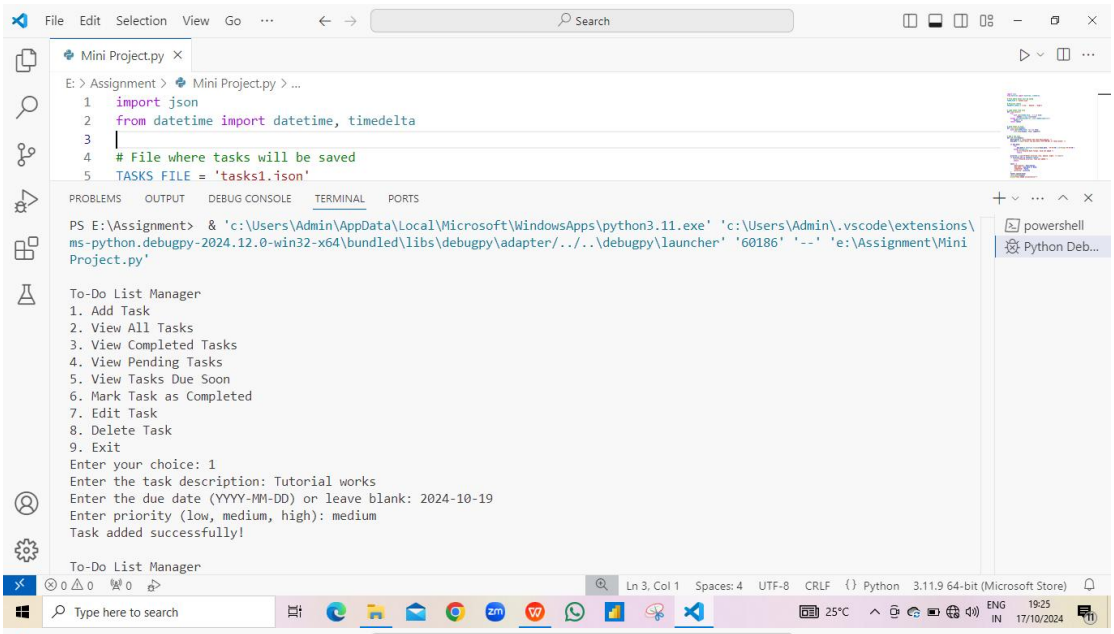
```
# Add a new task
def add_task(tasks1):
    description = input("Enter the task description: ")
    due_date = input("Enter the due date (YYYY-MM-DD) or leave blank: ")

    if due_date:
        try:
            due_date = datetime.strptime(due_date, '%Y-%m-
%d').strftime('%Y-%m-%d')
        except ValueError:
            print("Invalid date format. Task not added.")
            return

    priority = input(f"Enter priority (low, medium, high): ").lower()
    if priority not in PRIORITY_LEVELS:
        print("Invalid priority. Task not added.")
        return

    task = {
        'description': description,
        'due_date': due_date or None,
        'completed': False,
        'priority': priority
    }
    tasks.append(task)
    save_tasks(tasks1)
    print("Task added successfully!")
```

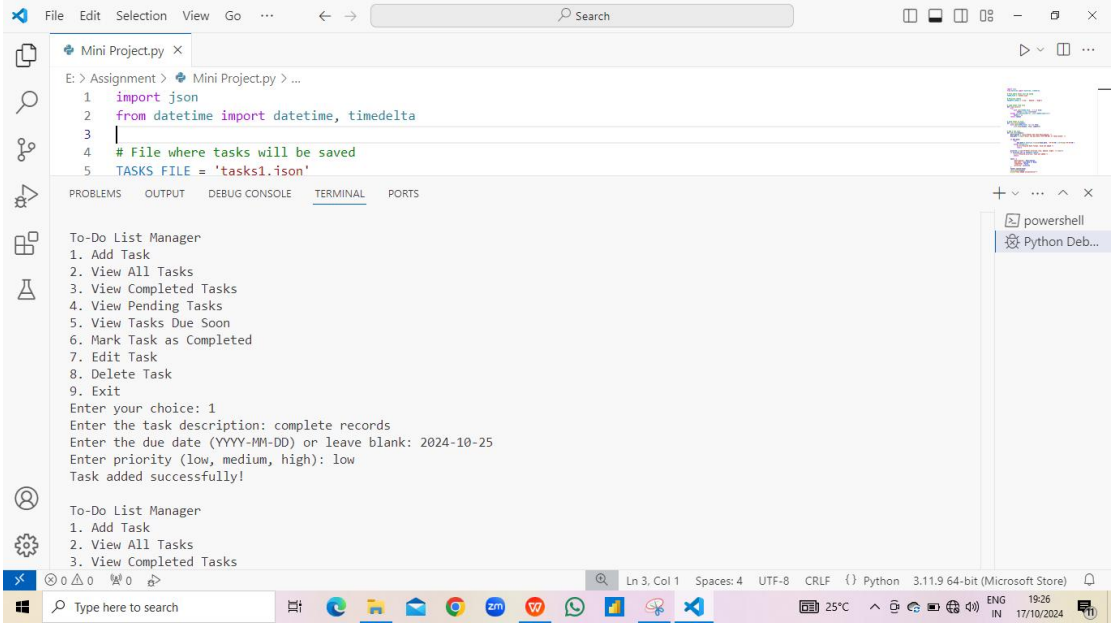
```
# View tasks based on filter
def view_tasks(tasks1, filter_by=None):
    if not tasks1:
        print("No tasks available.")
        return

    filtered_tasks = tasks1

    if filter_by == 'completed':
        filtered_tasks = [task for task in tasks1 if task['completed']]
    elif filter_by == 'pending':
        filtered_tasks = [task for task in tasks1 if not
task['completed']]
    elif filter_by == 'due_soon':
        today = datetime.now().date()
        soon = today + timedelta(days=3)
        filtered_tasks = [task for task in tasks1 if task['due_date']
and datetime.strptime(task['due_date'], '%Y-%m-%d').date() <= soon and
not task['completed']]

    if not filtered_tasks:
        print("No tasks found for the selected filter.")
        return

    for idx, task in enumerate(filtered_tasks, 1):
        status = "Completed" if task['completed'] else "Pending"
        print(f"{idx}. {task['description']} | Due: {task['due_date']}
| Status: {status} | Priority: {task['priority']}")


# Mark task as complete
def mark_task_completed(tasks1):
    view_tasks(tasks1, filter_by='pending')
    task_index = int(input("Enter task number to mark as completed: "))
- 1

    if 0 <= task_index < len(tasks1):
        Tasks1[task_index]['completed'] = True
        save_tasks(tasks1)
        print("Task marked as completed!")
    else:
        print("Invalid task number.")
```

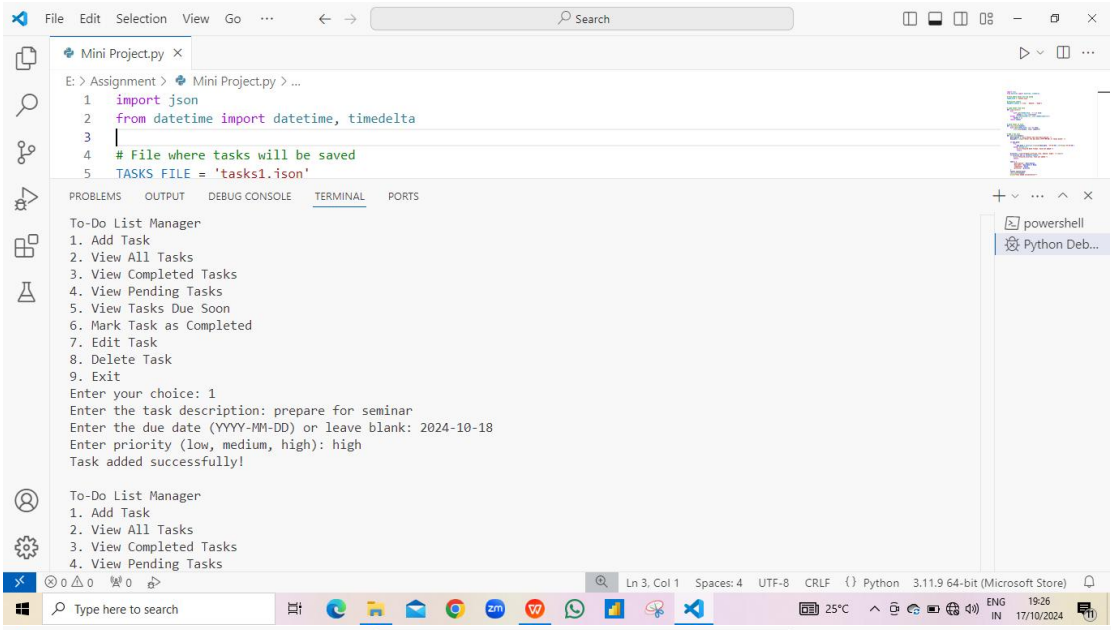
```
# Edit a task
def edit_task(tasks1):
    view_tasks(tasks1)
    task_index = int(input("Enter task number to edit: ")) - 1

    if 0 <= task_index < len(tasks1):
        task = tasks1[task_index]
        print(f"Editing task: {task['description']}")
        description = input(f"Enter new description (leave blank to
keep '{task['description']}'): ")
        due_date = input(f"Enter new due date (YYYY-MM-DD) or leave
blank to keep '{task['due_date']}': ")

        if due_date:
            try:
                due_date = datetime.strptime(due_date, '%Y-%m-
%d').strftime('%Y-%m-%d')
            except ValueError:
                print("Invalid date format. Task not updated.")
                return

        priority = input(f"Enter new priority (low, medium, high) or
leave blank to keep '{task['priority']}': ").lower()
        if priority and priority not in PRIORITY_LEVELS:
            print("Invalid priority. Task not updated.")
            return

        # Apply changes
        task['description'] = description or task['description']
        task['due_date'] = due_date or task['due_date']
        task['priority'] = priority or task['priority']

        save_tasks(tasks1)
        print("Task updated successfully!")
    else:
        print("Invalid task number.")


# Delete a task
def delete_task(tasks1):
    view_tasks(tasks1)
    task_index = int(input("Enter task number to delete: ")) - 1

    if 0 <= task_index < len(tasks1):
        Tasks1.pop(task_index)
        save_tasks(tasks1)
        print("Task deleted successfully!")
    else:
        print("Invalid task number.")
```

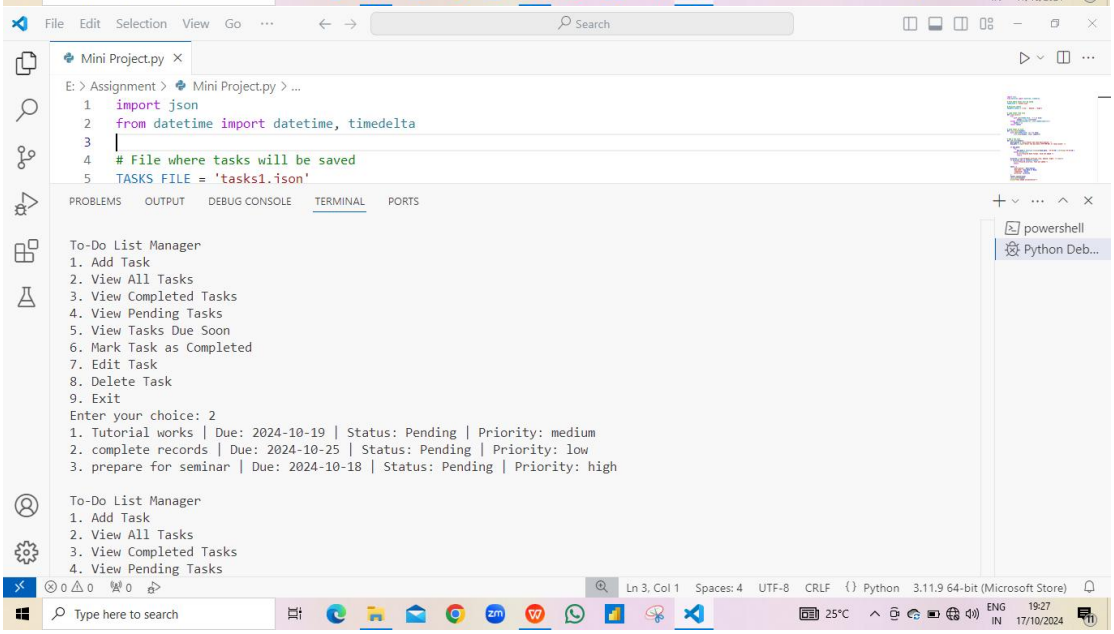
```python
# User Menu
def display_menu():
    print("\nTo-Do List Manager")
    print("1. Add Task")
    print("2. View All Tasks")
    print("3. View Completed Tasks")
    print("4. View Pending Tasks")
    print("5. View Tasks Due Soon")
    print("6. Mark Task as Completed")
    print("7. Edit Task")
    print("8. Delete Task")
    print("9. Exit")


# Main function
def main():
    Tasks1 = load_tasks()

    while True:
        display_menu()
        choice = input("Enter your choice: ")

        if choice == '1':
            add_task(tasks1)
        elif choice == '2':
            view_tasks(tasks1)
        elif choice == '3':
            view_tasks(tasks1, filter_by='completed')
        elif choice == '4':
            view_tasks(tasks1, filter_by='pending')
        elif choice == '5':
            view_tasks(tasks1, filter_by='due_soon')
        elif choice == '6':
            mark_task_completed(tasks1)
        elif choice == '7':
            edit_task(tasks1)
        elif choice == '8':
            delete_task(tasks1)
        elif choice == '9':
            print("Goodbye!")
            break
        else:
            print("Invalid choice. Please try again.")


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

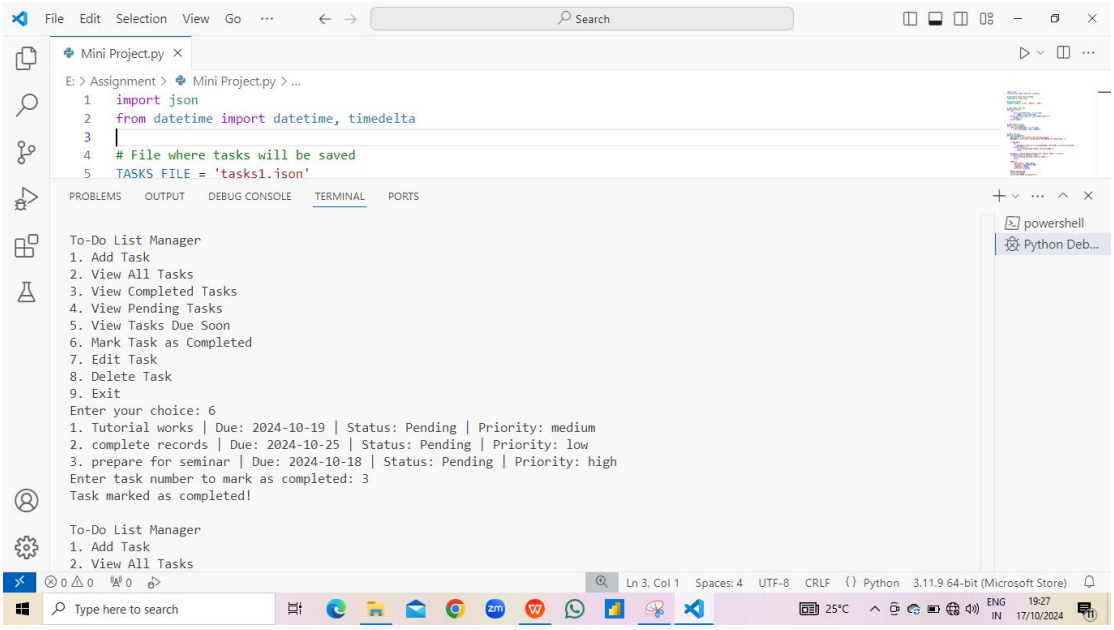
Output Screens:

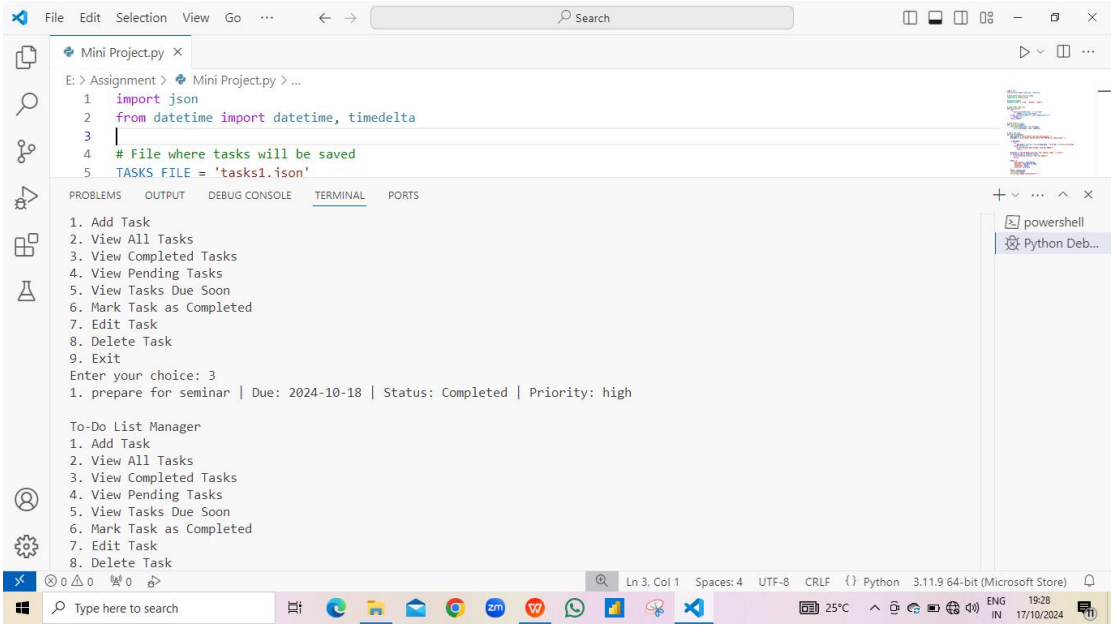


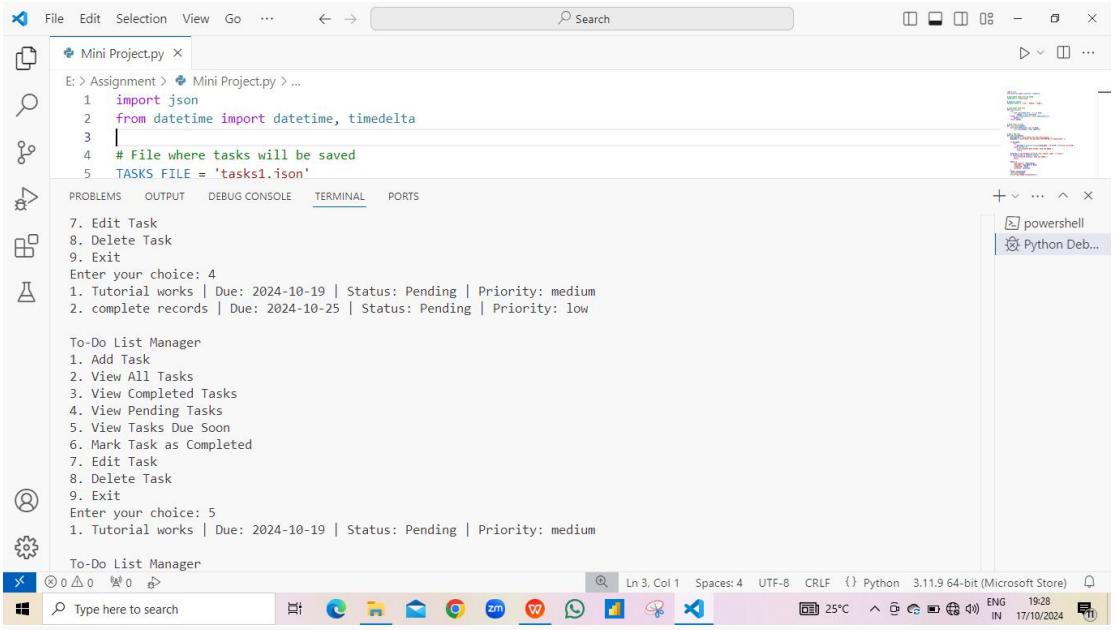


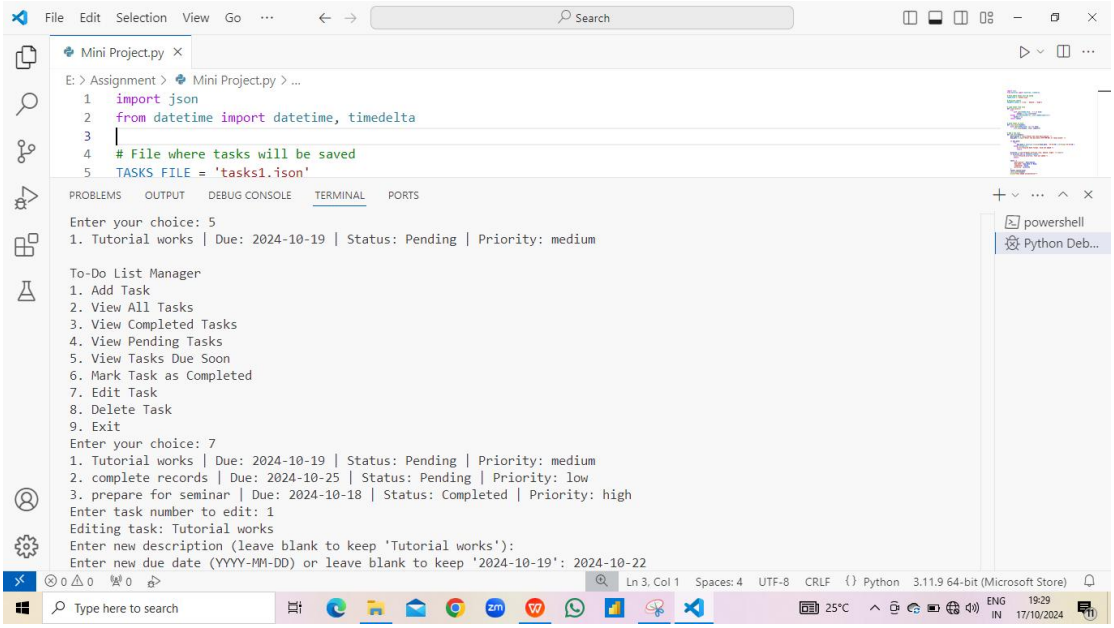

File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 1
Enter the task description: prepare for seminar
Enter the due date (YYYY-MM-DD) or leave blank: 2024-10-18
Enter priority (low, medium, high): high
Task added successfully!
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:26
17/10/2024

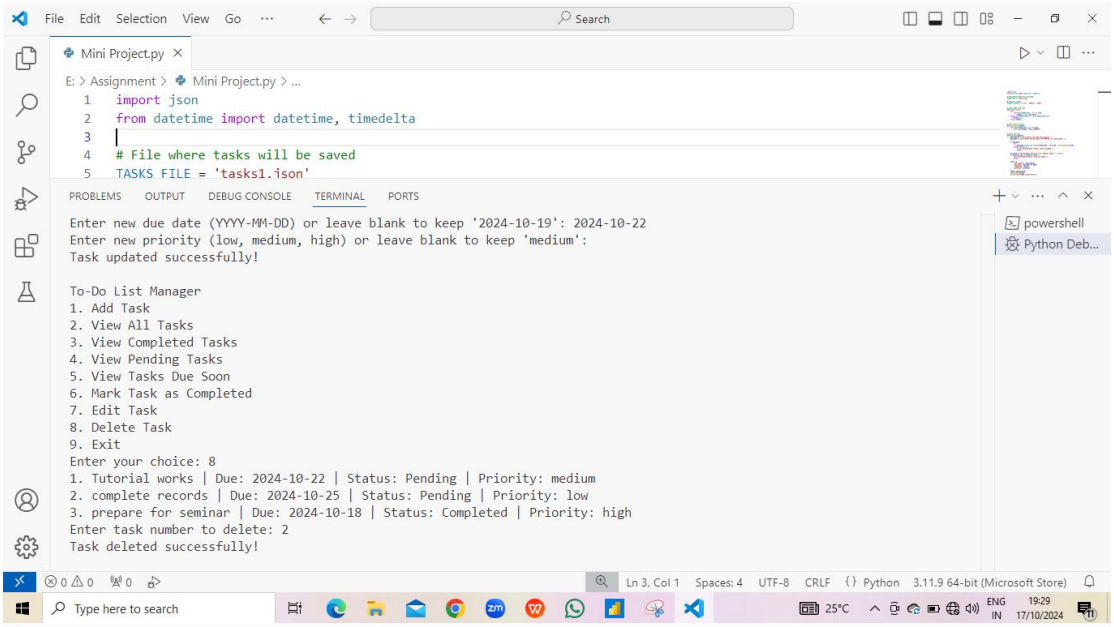

File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 2
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
3. prepare for seminar | Due: 2024-10-18 | Status: Pending | Priority: high
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:27
17/10/2024

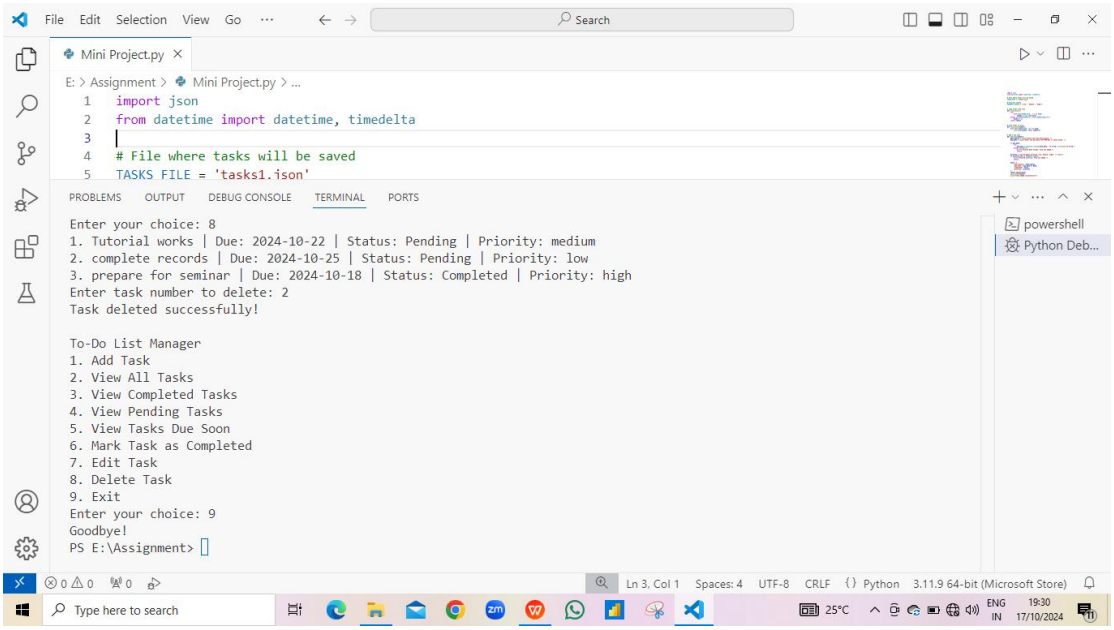

File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 6
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
3. prepare for seminar | Due: 2024-10-18 | Status: Pending | Priority: high
Enter task number to mark as completed: 3
Task marked as completed!
To-Do List Manager
1. Add Task
2. View All Tasks
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:27
17/10/2024


File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 3
1. prepare for seminar | Due: 2024-10-18 | Status: Completed | Priority: high
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:28
17/10/2024


File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 4
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 5
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
To-Do List Manager
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:28
17/10/2024


File Edit Selection View Go ···
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
Python Deb...
Enter your choice: 5
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 7
1. Tutorial works | Due: 2024-10-19 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
3. prepare for seminar | Due: 2024-10-18 | Status: Completed | Priority: high
Enter task number to edit: 1
Editing task: Tutorial works
Enter new description (leave blank to keep 'Tutorial works'):
Enter new due date (YYYY-MM-DD) or leave blank to keep '2024-10-19': 2024-10-22
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:29
17/10/2024


File Edit Selection View Go
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Enter new due date (YYYY-MM-DD) or leave blank to keep '2024-10-19': 2024-10-22
Enter new priority (low, medium, high) or leave blank to keep 'medium':
Task updated successfully!
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 8
1. Tutorial works | Due: 2024-10-22 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
3. prepare for seminar | Due: 2024-10-18 | Status: Completed | Priority: high
Enter task number to delete: 2
Task deleted successfully!
powershell
Python Deb...
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:29
17/10/2024


File Edit Selection View Go
Search
Mini Project.py
E: > Assignment > Mini Project.py > ...
1 import json
2 from datetime import datetime, timedelta
3
4 # File where tasks will be saved
5 TASKS_FILE = 'tasks1.json'
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Enter your choice: 8
1. Tutorial works | Due: 2024-10-22 | Status: Pending | Priority: medium
2. complete records | Due: 2024-10-25 | Status: Pending | Priority: low
3. prepare for seminar | Due: 2024-10-18 | Status: Completed | Priority: high
Enter task number to delete: 2
Task deleted successfully!
To-Do List Manager
1. Add Task
2. View All Tasks
3. View Completed Tasks
4. View Pending Tasks
5. View Tasks Due Soon
6. Mark Task as Completed
7. Edit Task
8. Delete Task
9. Exit
Enter your choice: 9
Goodbye!
PS E:\Assignment>
powershell
Python Deb...
Ln 3, Col 1 Spaces: 4 UTF-8 CRLF Python 3.11.9 64-bit (Microsoft Store)
Type here to search
25°C
ENG IN
19:30
17/10/2024