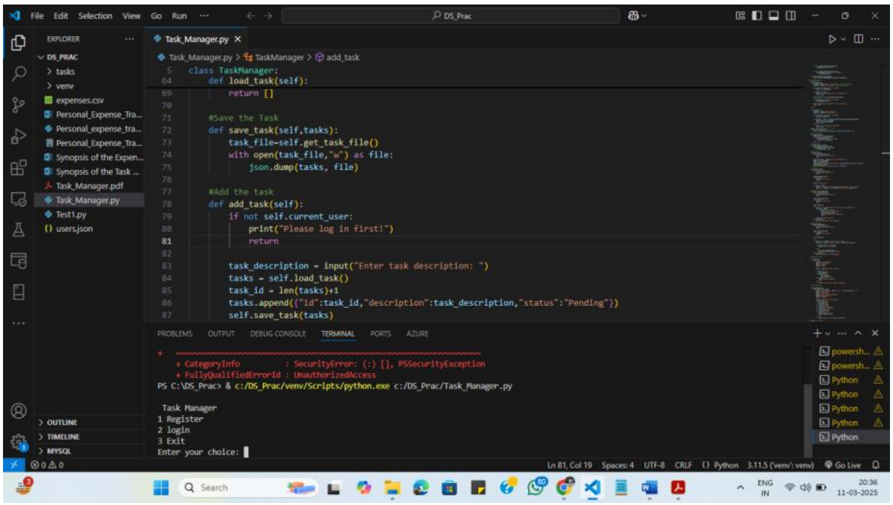
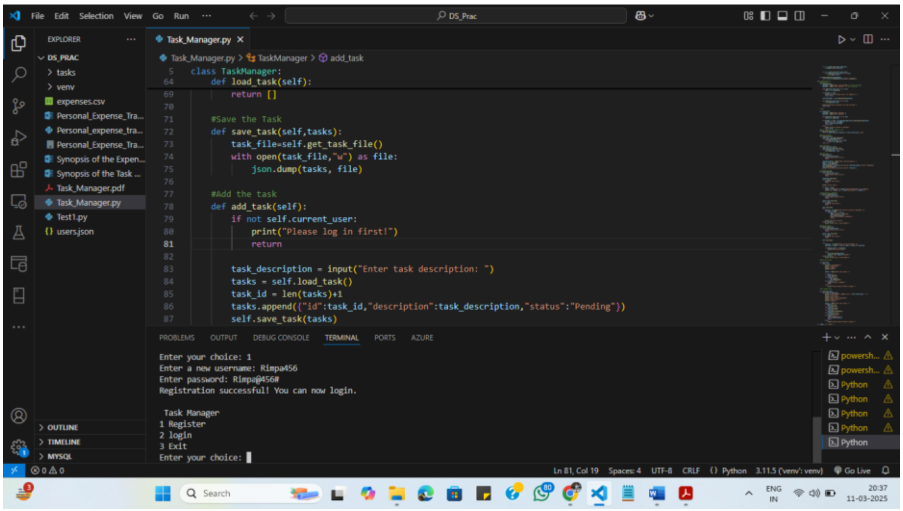
1. Run the Code



```
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
+    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : SecurityError: (:) [], PSSecurityException
    + FullyQualifiedErrorId : UnauthorizedAccess
PS C:\DS_Prac> & c:/DS_Prac/venv/Scripts/python.exe c:/DS_Prac/Task_Manager.py

 Task Manager
1 Register
2 login
3 Exit
Enter your choice: 
```

2. New User Registration

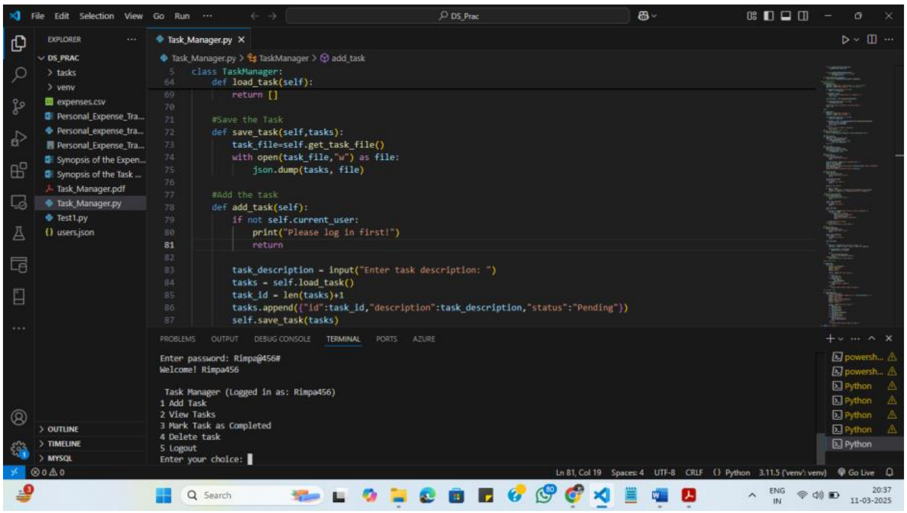


File Edit Selection View Go Run
DS_Prac
EXPLORER
DS_PRAC
Task_Manager.py
Task_Manager.py > TaskManager > add_task
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE
Ln 81, Col 19 Spaces: 4 UTF-8 CRLF {} Python 3.11.5 ('venv': venv) Go Live
20:37
11-03-2025

```
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter your choice: 1
Enter a new username: Rimpa456
Enter password: Rimpa@456#
Registration successful! You can now login.

 Task Manager
1 Register
2 login
3 Exit
Enter your choice: 
```

3. Login as user

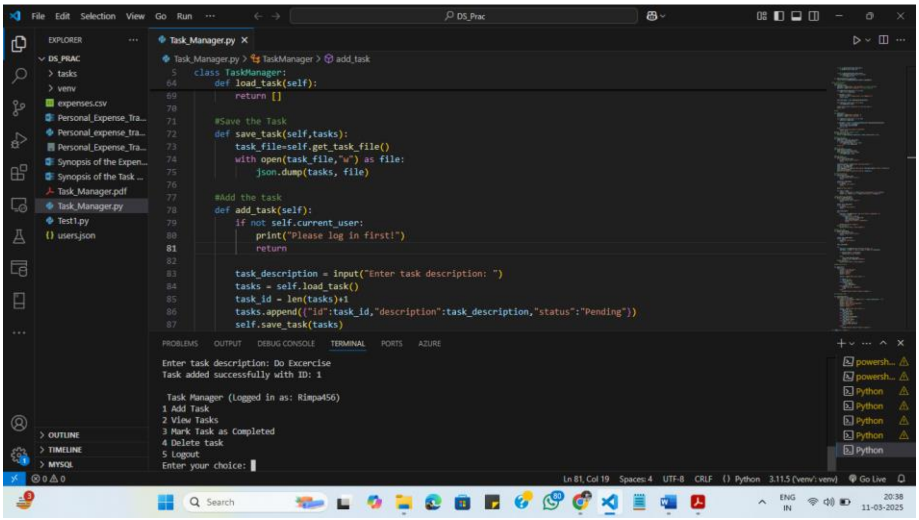


File Edit Selection View Go Run
DS_Prac
EXPLORER
DS_PRAC
Task_Manager.py
Task_Manager.py > TaskManager > add_task
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE
Ln 81, Col 19 Spaces: 4 UTF-8 CRLF {} Python 3.11.5 ('venv': venv) Go Live
20:37
11-03-2025

```
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter password: Rimpa@456#
Welcome! Rimpa456

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 
```

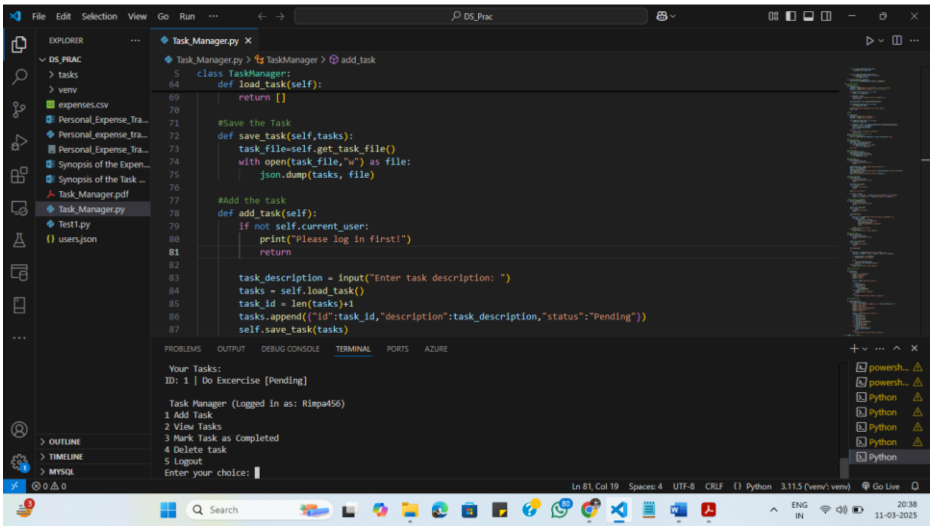
4. Add a Task



```
class TaskManager:
    def load_task(self):
        return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter task description: Do Excercise
Task added successfully with ID: 1

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 
```

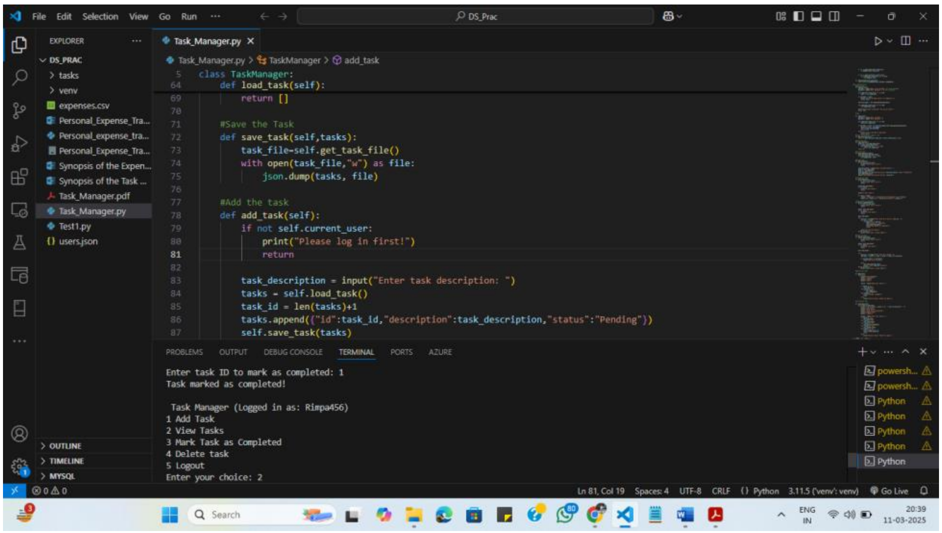
5. View Task



```
class TaskManager:
    def load_task(self):
        return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
 Your Tasks:
ID: 1 | Do Excercise [Pending]

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 
```

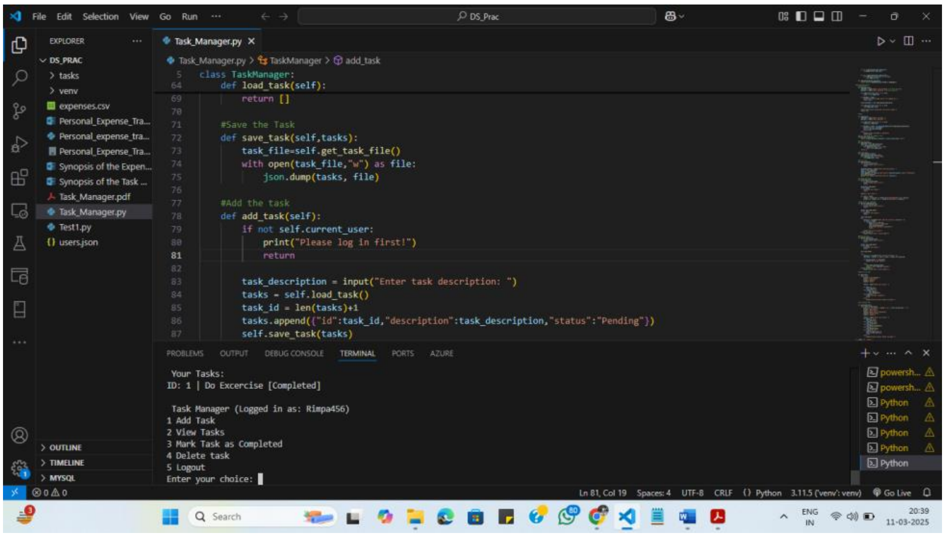
6. Mark Task as Completed



```
class TaskManager:
    def load_task(self):
        return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter task ID to mark as completed: 1
Task marked as completed!

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 2
```

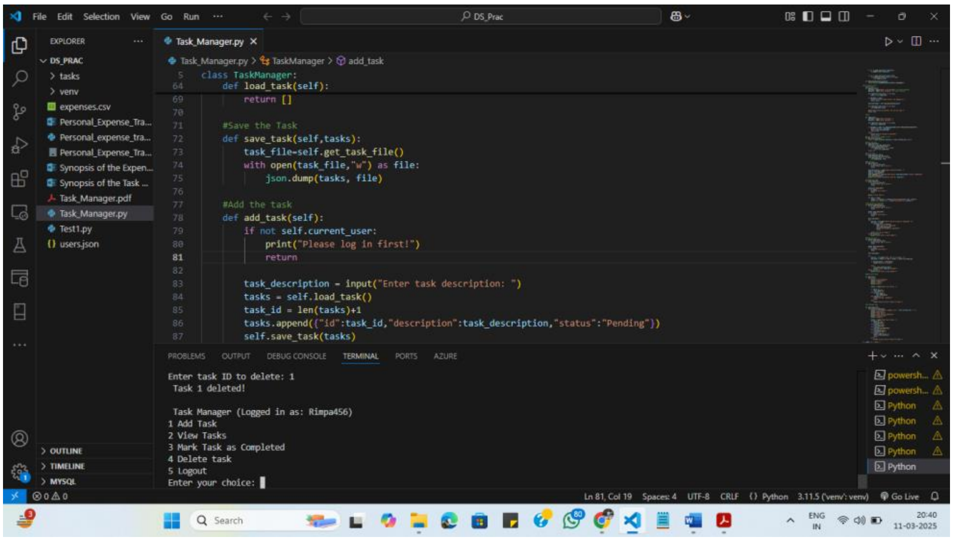
7. View Task as Completed


DS_Prac
Task_Manager.py
Task_Manager.py > TaskManager > add_task
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

```python
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
 Your Tasks:
ID: 1 | Do Excercise [Completed]

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 
```

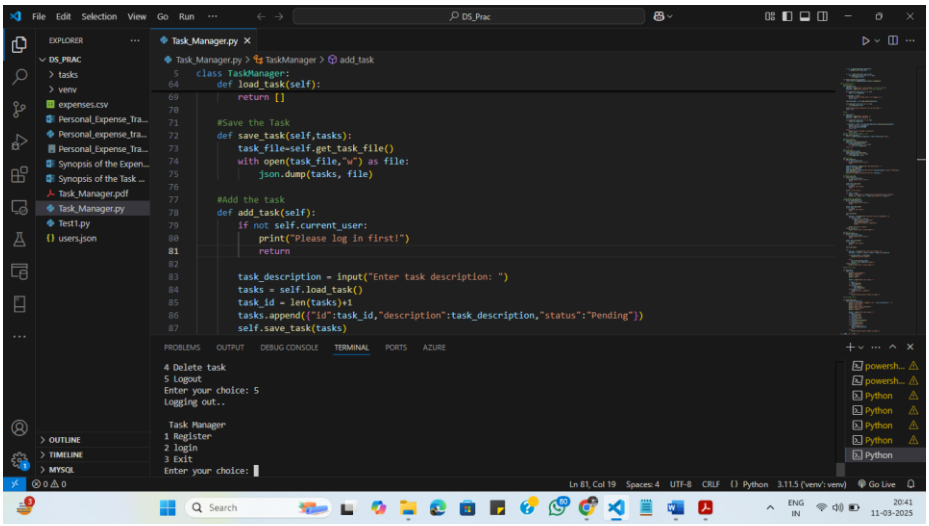
8. Delete the Task


DS_Prac
Task_Manager.py
Task_Manager.py > TaskManager > add_task
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

```python
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter task ID to delete: 1
 Task 1 deleted!

 Task Manager (Logged in as: Rimpa456)
1 Add Task
2 View Tasks
3 Mark Task as Completed
4 Delete task
5 Logout
Enter your choice: 
```

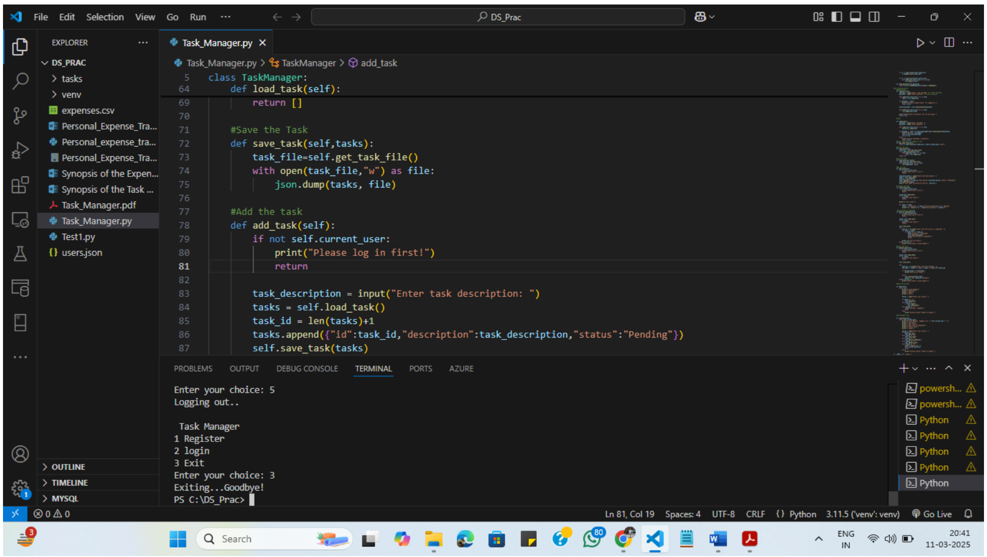
9. Logging Out


DS_Prac
Task_Manager.py
Task_Manager.py > TaskManager > add_task
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE

```python
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
4 Delete task
5 Logout
Enter your choice: 5
Logging out..

 Task Manager
1 Register
2 login
3 Exit
Enter your choice: 
```

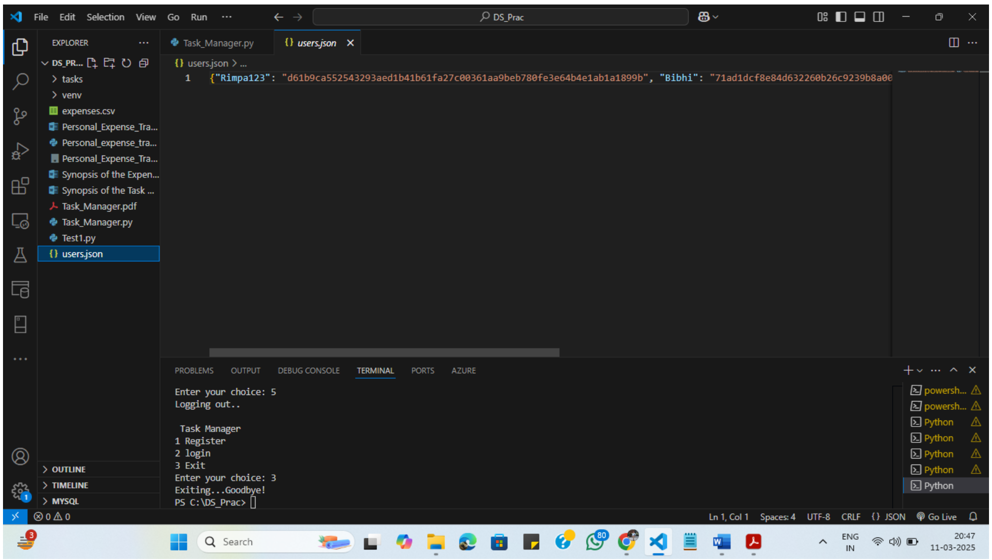
10. Exiting from Menu



```python
class TaskManager:
    def load_task(self):
            return []

    #Save the Task
    def save_task(self,tasks):
        task_file=self.get_task_file()
        with open(task_file,"w") as file:
            json.dump(tasks, file)

    #Add the task
    def add_task(self):
        if not self.current_user:
            print("Please log in first!")
            return

        task_description = input("Enter task description: ")
        tasks = self.load_task()
        task_id = len(tasks)+1
        tasks.append({"id":task_id,"description":task_description,"status":"Pending"})
        self.save_task(tasks)
```

```
Enter your choice: 5
Logging out..

 Task Manager
1 Register
2 login
3 Exit
Enter your choice: 3
Exiting...Goodbye!
PS C:\DS_Prac>
```

11. Users.json file with hashed password Security:



```json
{"Rimpa123": "d61b9ca552543293aed1b41b61fa27c00361aa9beb780fe3e64b4e1ab1a1899b", "Bibhi": "71ad1dcf8e84d632260b26c9239b8a0
```

```
Enter your choice: 5
Logging out..

 Task Manager
1 Register
2 login
3 Exit
Enter your choice: 3
Exiting...Goodbye!
PS C:\DS_Prac>
```