Day 4 Task

Create a Python program for managing a to-do list with the following functionalities:

1. **To-Do List Initialization:**
    - Initialize an empty list to represent the to-do list.

2. **Help Message:**
    - Display a help message to the user explaining the available commands:
        - **add** -> add a task to the to-do list.
        - **complete** -> mark a task as complete.
        - **view all** -> view the current tasks in the to-do list.
        - **view complete ->** view all the completed tasks in the to-do list.
        - **delete ->** Delete the to-do list and take it to the bin if it's not permanent.
        - **view incomplete ->** view all the incomplete tasks in the to-do list.
        - **view bin ->** view all the tasks that are deleted and are not currently in bin.
        - **restore ->** restore the deleted task from the bin.
        - **clear bin ->** delete all the to-dos that are presented in the bin.
        - **help** -> display all the help message.
        - **exit** -> exit the program.

3. **Task Addition:**
    - Implement the ability to add a task to the to-do list. When the user enters the **add** command, prompt them to enter the task description. Ensure that the same task cannot be added twice. When the task is added, the default status of the task will be incomplete.

4. **Task Completion:**

- Implement the ability to mark a task as complete. When the user enters the **complete** command, prompt them to enter the task number they want to mark as complete. Display an appropriate message if the task is not found. Ones the task is completed set the status of the task to completed.

5. **All To-Do List View:**
    - Implement the **view all** command to display the tasks in the to-do list. Include information about whether each task is complete or not.

6. **Completed To-Do List View:**
    - Implement the **view complete** command to display the completed task in the to-do list.

7. **Delete To-Do:**
    - Implement the **delete** command to delete the to-do list. While deleting task user can delete by the unique id of each task. Ones the user choose the task to be delete, then they should be asked whether they want to delete it permanently or not. If user wants to delete the to-do permanently then it should delete permanently from the list. Otherwise. The task should be deleted temporarily and should be move to the bin so, that later if user wish to restore the to-do they can do so.

8. **In completed To-Do List View:**
    - Implement the **view incomplete** command to display all the incomplete task in the to-do list.

9. **Invalid Options Entered:**
    - Display the invalid message. After then show them again the all the available options.

10. View Bin:

- Implement the **view bin** command that will display all the deleted task and are presented in the bin.

11. Restore Deleted To-Do:

- Implement the **restore** command that will help user restore the task they have previously deleted.

12. Clear Bin Command:

- Implement the clear bin command that will delete all the to-do's permanently from the bin.

13. Help Command:

- Display all the help messages.

14. Exit Confirmation:

- When the user enters **exit**, prompt them with "Are you sure you want to exit?" If the user inputs "yes," the program should stop. If the user inputs anything else, the program should continue.

NOTE: Please attempt to complete this task by analyzing the requirements and building your own logic. Avoid using external assistance, including chat GPT, to enhance your problem-solving skills. If you encounter challenges, discuss them with your supervisor or fellow interns. Feel free to research specific topics independently. Do not copy the entire question and seek answers from external sources like chat GPT. Thank You.