Name : Farhan Ali   
class : BSDS-3A

Artificial Intelligence  
  
instructor name: Rashikh Ali

Program:

To-Do List program

Introduction:

This program is a simple **To-Do List application** made using Python. It allows users to add multiple tasks and keep them organized. Each task can be marked as completed or left as pending. The program runs in a loop, giving users choices through a menu. It helps in managing daily work step by step. The design is easy to use and suitable for beginners. Overall, it is a practical project to learn Python basics.

Code:

def main():

    task\_list = []

    while True:

        print(" My To-Do List")

        print("1. Add Task")

        print("2. Show Tasks")

        print("3. Mark Task as Done")

        print("4. Exit")

        print("=" \* 20)

        user\_choice = input("What would you like to do? Enter your choice: ")

        if user\_choice == '1':

            how\_many = int(input("How many tasks do you want to add: "))

            for task\_num in range(how\_many):

                print(f"Task #{task\_num + 1}:")

                new\_task = input("Enter the task: ")

                while True:

                    try:

                        task\_priority = int(input("Enter priority (1 = High, 2 = Medium, 3 = Low): "))

                        if task\_priority in [1, 2, 3]:

                            break

                        else:

                            print("Please enter 1, 2, or 3 only!")

                    except ValueError:

                        print("Please enter a valid number!")

                task\_dict = {

                    "task": new\_task,

                    "done": False,

                    "priority": task\_priority

                }

                task\_list.append(task\_dict)

                print(" Task added successfully!")

        elif user\_choice == '2':

            if not task\_list:

                print("No tasks found! Add some tasks first.")

            else:

                print(" Your Tasks (sorted by priority) ")

                sorted\_tasks = sorted(task\_list, key=lambda x: x["priority"])

                for idx, current\_task in enumerate(sorted\_tasks):

                    task\_status = " Done" if current\_task["done"] else "○ Not Done"

                    priority\_text = ""

                    if current\_task["priority"] == 1:

                        priority\_text = "HIGH"

                    elif current\_task["priority"] == 2:

                        priority\_text = "MEDIUM"

                    else:

                        priority\_text = "LOW"

                    print(f"{idx + 1}. {current\_task['task']} - {task\_status} - Priority: {priority\_text}")

        elif user\_choice == '3':

            if not task\_list:

                print("No tasks available to mark as done!")

            else:

                print("\nCurrent tasks:")

                sorted\_for\_display = sorted(task\_list, key=lambda x: x["priority"])

                for i, t in enumerate(sorted\_for\_display):

                    status = "done" if t["done"] else "failed"

                    print(f"{i + 1}. {t['task']} [{status}]")

                try:

                    task\_to\_complete = int(input("\nEnter the task number to mark as done: ")) - 1

                    if 0 <= task\_to\_complete < len(task\_list):

                        sorted\_list = sorted(task\_list, key=lambda x: x["priority"])

                        selected\_task = sorted\_list[task\_to\_complete]

                        selected\_task["done"] = True

                        print(f" Task '{selected\_task['task']}' marked as done!")

                    else:

                        print("Oops! Invalid task number. Please try again.")

                except ValueError:

                    print("Please enter a valid number!")

        elif user\_choice == '4':

            print("Thanks for using the To-Do List! Goodbye!")

            break

        else:

            print("Hmm, that's not a valid choice. Please pick 1, 2, 3, or 4.")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

Explaination of the code :

Here’s the explanation in clear step-wise points:

1. The program defines a **main()** function where all the code runs.
2. An empty list called **tasks** is created to store the tasks.
3. A continuous loop displays a menu with four choices:
   * Add Task
   * Show Tasks
   * Mark Task as Done
   * Exit
4. If the user selects **Add Task**, the program asks how many tasks to enter and then saves them in the list with the status “Not Done.”
5. If the user selects **Show Tasks**, the program prints all tasks along with their status, either “Done” or “Not Done.”
6. If the user selects **Mark Task as Done**, the program asks for the task number and updates its status to “Done.”
7. If the user selects **Exit**, the loop ends and the program stops running.
8. If the user enters anything else, the program shows an “Invalid choice” message.
9. The line **if name == "main": main()** makes sure the program starts when it is run directly