

To Do List Application

AIM: To create a command-line or GUI-based application using Python, allowing users to create, update, and track their to-do lists.

PROGRAM:

```
tasks = []
```

```
while True:
```

```
    print("\nTo-Do List:")
```

```
    print("1. Show Tasks")
```

```
    print("2. Add Task")
```

```
    print("3. Update Task")
```

```
    print("4. Delete Task")
```

```
    print("5. Exit")
```

```
    choice = input("Enter your choice (1-5): ")
```

```
    if choice == '1':
```

```
        if not tasks:
```

```
            print("No tasks yet.")
```

```
        else:
```

```
            for i, task in enumerate(tasks, 1):
```

```
                print(f"{i}. {task}")
```

```
    elif choice == '2':
```

```
        task = input("Enter new task: ")
```

```
        tasks.append(task)
```

```
        print("Task added.")
```

```
elif choice == '3':  
    for i, task in enumerate(tasks, 1):  
        print(f"{i}. {task}")  
    try:  
        num = int(input("Enter task number to update: "))  
        if 1 <= num <= len(tasks):  
            tasks[num - 1] = input("Enter new task: ")  
            print("Task updated.")  
        else:  
            print("Invalid number.")  
    except ValueError:  
        print("Please enter a number.")
```

```
elif choice == '4':  
    for i, task in enumerate(tasks, 1):  
        print(f"{i}. {task}")  
    try:  
        num = int(input("Enter task number to delete: "))  
        if 1 <= num <= len(tasks):  
            tasks.pop(num - 1)  
            print("Task deleted.")  
        else:  
            print("Invalid number.")  
    except ValueError:  
        print("Please enter a number.")
```

```
elif choice == '5':
```

```
    print("Goodbye!")
```

```
    break
```

```
else:
```

```
    print("Please enter a number from 1 to 5.")
```

SAMPLE OUTPUT:

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 1

No tasks yet.

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 2

Enter new task: munikumar

Task added.

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 1

1. munikumar

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 2

Enter new task: muni bhaskar

Task added.

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 3

1. munikumar
2. muni bhaskar

Enter task number to update: 2

Enter new task: muni hi

Task updated.

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 1

1. munikumar
2. muni hi

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 4

1. munikumar
2. muni hi

Enter task number to delete: 2

Task deleted.

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 1

1. munikumar

To-Do List:

1. Show Tasks
2. Add Task
3. Update Task
4. Delete Task
5. Exit

Enter your choice (1-5): 5

Goodbye!

RESULT: Hence, To create a command-line or GUI-based application using Python, allowing users to create, update, and track their to-do lists is successfully completed.