Develop a basic to-do list application using functions and data structures

Project Overview:

Objective: Develop a simple to-do list application using Python with an emphasis on functions and data structures. Project Overview: Objective: Develop a simple to-do list application using Python with an emphasis on functions and data structures.

Key Components:

1. Functions: You'll be implementing various functions to handle different aspects of the to-do list application. Functions are modular blocks of code that perform specific tasks, making your code more organized and easier to understand. Function to add a task Function to delete a task Function to display the list of tasks Function to mark a task as complete.

2. Data Structures: Utilize appropriate data structures to store and manage the to-do list. A common choice would be a list or a dictionary, but you can explore other options based on your creativity and understanding.

# To-Do List Application

def add\_task(tasks, task\_name):

"""Add a new task to the list."""

task = {"task": task\_name, "completed": False}

tasks.append(task)

print(f"Task '{task\_name}' added successfully.")

def delete\_task(tasks, task\_index):

"""Delete a task from the list."""

if 0 <= task\_index < len(tasks):

removed\_task = tasks.pop(task\_index)

print(f"Task '{removed\_task['task']}' deleted successfully.")

else:

print("Invalid task index.")

def display\_tasks(tasks):

"""Display all tasks in the list."""

if not tasks:

print("No tasks in the list.")

return

print("\nTo-Do List:")

for index, task in enumerate(tasks, 1):

status = "Complete" if task["completed"] else "Incomplete"

print(f"{index}. {task['task']} - {status}")

def mark\_task\_complete(tasks, task\_index):

"""Mark a task as complete."""

if 0 <= task\_index < len(tasks):

tasks[task\_index]["completed"] = True

print(f"Task '{tasks[task\_index]['task']}' marked as complete.")

else:

print("Invalid task index.")

def main():

tasks = []

while True:

print("\n--- To-Do List Menu ---")

print("1. Add Task")

print("2. Delete Task")

print("3. Display Tasks")

print("4. Mark Task as Complete")

print("5. Exit")

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

if choice == "1":

task\_name = input("Enter the task name: ")

add\_task(tasks, task\_name)

elif choice == "2":

display\_tasks(tasks)

try:

task\_index = int(input("Enter the task index to delete: ")) - 1

delete\_task(tasks, task\_index)

except ValueError:

print("Invalid input. Please enter a number.")

elif choice == "3":

display\_tasks(tasks)

elif choice == "4":

display\_tasks(tasks)

try:

task\_index = int(input("Enter the task index to mark complete: ")) - 1

mark\_task\_complete(tasks, task\_index)

except ValueError:

print("Invalid input. Please enter a number.")

elif choice == "5":

print("Exiting the application.")

break

else:

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

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

main()

OUTPUT Snippet:

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated