Simple To-Do List

# Technology Stack

• C++ (Object-Oriented Programming)  
• Win32 API (for GUI)

# Overview

A lightweight desktop application to manage daily tasks with a simple graphical interface. Users can add, view, delete, and mark tasks. The project demonstrates Win32 GUI programming combined with clean C++ OOP structure.

# Objective

Show practical implementation of C++ OOP principles in a Windows GUI app, emphasizing separation of concerns and user interactivity.

# Main Features

* • Add Task: Type a task into the edit box and click Add.
* • Task List Display: All tasks appear in a scrollable list box.
* • Toggle Complete: Double-click a task to mark it done/undone.
* • Delete Task: Select a task and click Remove.

# Class Design (OOP Layer)

TaskManager (in TaskManager.h / TaskManager.cpp):  
• Stores tasks in a std::vector of { text, done }.  
• Methods: addTask(), removeTask(), toggleTask(), getTaskCount(), getTask(), isCompleted(), loadFromFile(), saveToFile().  
• File I/O handles persistent storage in tasks.txt.

# GUI Design (main.cpp)

• Edit Box: User enters new tasks.  
• Add Button: Triggers TaskManager::addTask().  
• List Box: Displays tasks with [ ] or [X] prefix.  
• Remove Button: Deletes the selected task.  
• Event Handling:  
 - Add: Reads edit box text, updates model & view.  
 - Remove: Removes selected item.  
 - Double-click: Toggles completion state and refreshes the view.

# Project Files

* • main.cpp — Win32 GUI code with detailed comments.
* • TaskManager.h — Class declaration for task logic.
* • TaskManager.cpp — Implementation of task methods and file I/O.

# Advantages

* • Clear separation of GUI and business logic.
* • Hands-on Win32 API experience.
* • Simple, well-commented code—ideal for demonstrations.
* • Easy to extend and maintain.

# How to Compile (Dev-C++)

* • Create a Windows Application project.
* • Add main.cpp, TaskManager.h, TaskManager.cpp.
* • In Project Options → Parameters → Linker, add -mwindows.
* • Compile & Run—no console window appears.

# Suggestions for Extension

* • Add task priorities or categories.
* • Color-code completed vs. pending tasks.
* • Implement a timer reminder feature.
* • Export/import task lists (CSV or JSON).

# Conclusion

This To-Do List app illustrates core C++ OOP concepts in a real Windows GUI. It's an excellent academic or interview project, showcasing clean architecture and user-centric design.