

# Qt 行程管理系統—技術實作說明

## 一、專案概述

本專案為使用C++ 與 Qt Widgets 所實作之桌面行程管理系統，具備任務新增、分類、提醒、國定假日自動產生與資料永久儲存等功能。系統採事件導向設計，並以資料結構為核心，動態重建使用者介面。

## 二、核心資料結構設計（Task Model）

系統以 Task 結構作為主要資料模型，包含標題、日期、時間、分類、顏色、完成狀態、提醒狀態、優先權與群組資訊。所有任務皆儲存於 QVector<Task> 中，UI 僅作為資料的視覺呈現。

## 三、國定假日自動產生機制

程式啟動時會依照系統當前年份，自動建立國定假日任務。透過 Holiday 結構描述節日資料，並於加入前檢查是否已存在，以避免重複插入。

## 四、使用者介面與資料同步

系統透過 rebuildTree() 函式，根據 Task 資料動態重建 QTreeWidget 內容，並依是否為優先任務分類顯示，確保 UI 與資料狀態同步。

## 五、提醒系統設計

使用 QTimer 每 30 秒檢查任務狀態，若任務時間已到且尚未提醒，則透過 QMessageBox 顯示提醒，並即時更新 reminded 狀態避免重複提醒。

## 六、資料永久儲存（JSON）

系統使用 QJsonDocument 將任務資料序列化儲存至 tasks.json，並於程式啟動時自動讀取還原，確保資料跨執行保存。

## 七、任務刪除與完成控制

任務刪除使用 C++ erase-remove idiom 實作，完成勾選則加入任務順序判斷，避免未完成前置任務即完成後續任務。

## 八、技術總結

本專案整合 Qt UI、資料模型設計、事件處理、時間控制與 JSON 序列化，具備中高階桌面應用程式開發水準，適合作為課程專題或作品展示。