

課程：C#程式設計

WindowsForm 與資料庫作業

1. 繳交期限：12/02 (二) 08:00(第十三週)
2. 作業範圍：C# 程式設計 第一章 ~ 第二十章
3. 請上傳專案程式碼並壓縮成壓縮檔(P123456789.zip)
4. 專案命名格式為:C#_WF_學號(C#_WF_P123456789)
5. 如有疑惑可在 moodle 留言或是寄信詢問
6. 嚴禁抄襲、程式碼雷同、作弊，如發現上述問題雙方以 0 分
7. 共計 200 分(視同兩次作業成績)

題目名稱：CNC 工具機監控儀表板

題目說明：(請繳交程式碼)(檔案名稱：C#_WF_學號)

題目目標：

開發一個 Windows Forms 桌面應用程式，用以模擬智慧工廠的「戰情室」。此應用程式需能：

1. 管理工廠中的所有工具機。
2. 即時監控每台機器的運作狀態（閒置、運轉中、警報）。
3. 對機台進行工單的指派與生產模擬。
4. 將所有生產日誌存入資料庫。
5. 分析特定機台的設備綜合效率，並以儀表板形式呈現。

資料庫設計：

- 資料庫數據請使用附件提供之 csv 檔來建立資料庫
- 涉及「跨資料表」或「多步驟」的資料庫寫入，必須確保資料的一致性。

資料庫一：MachineTools(10%)

說明：儲存工廠中所有工具機的靜態基本資料與即時動態狀態。

欄位名稱	資料型別	備註
MachineID	NVARCHAR(50)	Primary Key，機台的唯一編號，例如 "CNC-001"
Model	NVARCHAR(50)	機台型號，例如 "XYZ-VMC-850"
PurchaseDate	DATE	購入日期
Status	NVARCHAR(20)	機台目前狀態 ("閒置", "運轉中", "警報")
CurrentWorkOrderID	NVARCHAR(50)	Foreign Key，允許空值關聯至 WorkOrders.WorkOrderID，表示目前正在執行的工單

資料庫二：WorkOrders(10%)

說明： 儲存所有待生產或正在生產的工單資訊。

欄位名稱 (Field Name)	資料型別 (Data Type)	約束 / 備註 (Constraints / Notes)
WorkOrderID	NVARCHAR(50)	Primary Key，工單的唯一編號，格式如下 "20251028-001"
ProductName	NVARCHAR(50)	產品名稱
TargetQuantity	INT	預計生產數量
Status	NVARCHAR(20)	工單目前狀態 ("未開始", "進行中", "已完成")
CostTime	INT	表示這張工單預計完成所需的時間(單位:分鐘)

資料庫三：ProductionLog(10%)

說明： 系統的核心資料表，記錄所有機台在生產過程中發生的每一筆事件，包含生產、警報等。

欄位名稱 (Field Name)	資料型別 (Data Type)	約束 / 備註 (Constraints / Notes)
LogID	INT	Primary Key，Identity (自動遞增)紀錄的流水號
MachineID	NVARCHAR(50)	Foreign Key，關聯至 MachineTools.MachineID
WorkOrderID	NVARCHAR(50)	Foreign Key，關聯至 WorkOrders.WorkOrderID
Timestamp	DATETIME	紀錄事件發生的精確時間
LogType	NVARCHAR(20)	紀錄類型 ("開始生產"、"生產"、"發生警報"、"警報解除"、"完成生產")
GoodCount	INT	該次事件當前時間的良品數
BadCount	INT	該次事件當前時間的不良品數

資料庫四：AlarmLog(10%)

說明： 專門記錄警報事件，包含發生與解除時間，以及發生當下的生產進度。

欄位名稱	資料型別	約束/備註
AlarmLogID	INT	Primary Key，Identity
MachineID	NVARCHAR(50)	Foreign Key，關聯至 MachineTools
WorkOrderID	NVARCHAR(50)	Foreign Key，關聯至 WorkOrders
ProductionCountAtAlarm	INT	NULL，記錄警報發生時，該 WorkOrderID 已累計的生產總數 (GoodCount + BadCount)
AlarmStartTime	DATETIME	警報發生時間
AlarmEndTime	DATETIME	NULL (警報解除時間。若為 NULL 表示未解除)
Status	NVARCHAR(20)	警報狀態 ("Active", "Cleared")

功能需求：

一. 核心類別

請建立至少兩個類別：MachineTool 、 WorkOrder，其屬性應對應資料庫欄位。

二. 主視窗 (FormMain) - 監控儀表板

- 介面佈局：
 - a. 左側 (Panel): 使用 ListBox 或 DataGridView 顯示所有機台 (MachineID) 及其狀態。
 - b. 右側 (Panel): 顯示目前選定機台的詳細資訊，以及解除警報的按鈕。
 - c. 下方 (StatusStrip): 顯示資料庫連線狀態。
 - d. 上方 (MenuStrip): 建立欄位工具，有一子項目為儀表板，點擊後能開啟

- 功能：

甲、(10%)啟動初始化與資料載入：

i. 狀態重置：

1. 將系統需先找出 MachineTools 資料表中所有 Status 不為"閒置"的機台，並取得它們正在執行的 CurrentWorkOrderID 列表。
2. 系統必須更新 WorkOrders 資料表，將上述列表 CurrentWorkOrderID 對應到的工單，其 Status 從 "進行中" 重置回 "未開始"。
3. 更新 AlarmLog 資料表，將所有 Status = "Active" 的警報紀錄，其 Status 更新為 "Cleared"，並將 AlarmEndTime 設為當前時間。
4. 更新 MachineTools 資料表，將所有 Status 不為"閒置"的機台，其 Status 更新為 "閒置"，並將 CurrentWorkOrderID 設為 NULL。

ii. 資料載入：視窗啟動時，從 MachineTools 資料表載入所有機台到左側清單中。

乙、(5%)狀態顯示：左側清單應根據機台 Status 顯示不同顏色（例如：運轉中=綠色，警報=紅色，閒置=灰色）。

丙、(5%)詳細資訊：點選左側任一機台，右側應顯示當前機台於 MachineTools 資料表中的全部資料(顯示方式可自行設計)；若狀態為 "警報(未解除)"，需額外查詢 AlarmLog 取得最新的 Active 警報資訊並顯示，同時顯示「解除警報」按鈕。

丁、(20%)工單指派：

i. 右側需有一個 ComboBox，僅載入所有 Status = "未開始" 的工單。

ii. 一個「指派工單」按鈕，按下後：

1. 檢查機台狀態是否為 "閒置"，若不為"閒置"，則需跳出 MessageBox 顯示"無法進行指派"。
2. 更新 MachineTools 資料表，將 CurrentWorkOrderID 設為選中的工單 ID，並將 Status 改為 "運轉中"。
3. 更新 WorkOrders 資料表，將該工單 Status 改為 "進行中"。
4. 新增一筆資料至 ProductionLog 資料表，並依照供單指派內容新增 MachineID、WorkOrderID，並將 LogType 建為"開始生產"。

戊、(40%)即時模擬(時間自行設定)：

i. 在 Tick 事件中：

1. 邏輯迴圈：

找出所有 Status = "運轉中" 的機台，並依序處理每一台：

a. 檢查該機台是否已達「完成生產」的條件：

- 條件成立時：系統執行「完成生產」的所有資料庫更新（更新 WorkOrders、MachineTools、新增 ProductionLog），並結束此機台的本輪 Tick 處理。

b. 如果該機台未達「完成生產」條件：

- 系統透過 Random 隨機決定本次 Tick 是「觸發警報」還是「正常生產」（例如：3% 機率觸發警報，97% 機率正常生產）。

- 觸發警報的狀況：系統執行「模擬警報」的所有資料庫更新（更新 MachineTools Status="警報"、INSERT ProductionLog LogType="發生警報"、INSERT AlarmLog Status="Active" 並記錄資訊）
- 正常生產的狀況：系統執行「模擬生產」的邏輯。具體做法為：設定良率門檻(例 95%)，用 Random 產生 1-100 隨機數。若 randomNumber <= 門檻，則 goodCount=1, badCount=0；否則 goodCount=0, badCount=1。系統將上述 goodCount 與 badCount 結果，新增一筆新紀錄到資料庫 ProductionLog，其 LogType 為 "生產"。

2. 即時刷新:

重新載入左側機台清單，讓 UI 反映最新狀態。

己、(15%)解除警報:

- 由介面布局的 b 點中提到的按鈕，點擊後即可解除警報狀態，並恢復生產。
- 更新資料表 AlarmLog 中找到該機台 Status="Active" 的最新記錄，設 AlarmEndTime=NOW(), Status="Cleared"。
- 資料表 MachineTools 中機台的 Status= "運轉中"。
- ProductionLog 中增加一個 LogType="警報解除"的紀錄。

庚、(5%)例外處理: 所有資料庫操作都必須放在 try...catch 區塊中，若發生錯誤，應在 StatusStrip 或 MessageBox 中顯示錯誤訊息。

三. 生產分析儀表板

- 介面佈局：

- a. ComboBox (cmbMachines): 選擇機台。
- b. Button (btnAnalyze): "開始分析"。
- c. Label (用於顯示結果): lblTotalGood, lblTotalBad, lblYieldRate, lblAlarmCount, lblTotalDowntime。
- d. DataGridView (dgvLongestDowntime): 用於顯示最長停機事件。
- e. DataGridView (dgvCompletedWOs): 用於顯示已完成工單分析。

- 功能：

甲、(60%)生產分析計算：

- i. 按下「開始分析」按鈕後。
- ii. 使用 ADO.NET 撈取 ProductionLog 中，符合「選定機台」的所有日誌，將資料存入 DataSet 或 List<ProductionLog>。
- iii. 使用 ADO.NET 撈取 AlarmLog 中，符合「選定機台」的所有警報紀錄，存入 DataSet 或 List<AlarmLog>。
- iv. 使用 LINQ to DataSet 或 LINQ to Objects 進行計算：
 1. totalGood = 篩選結果中所有 GoodCount 的總和。
 2. totalBad = 篩選結果中所有 BadCount 的總和。
 3. 歷史總體良率: $(\text{double})\text{totalGood} / (\text{totalGood} + \text{totalBad})$ 。
 4. alarmCount = AlarmLog 查詢結果的筆數。
 5. 計算總停機時間 (分鐘):
 - 從 AlarmLog 的查詢結果中，計算每筆警報的持續時間。
 - 若 AlarmEndTime 為 NULL，使用當前時間 (DateTime.Now) 計算持續時間。
 - 將所有持續時間加總得到總分鐘數。
 6. 前三名最長停機事件:
 - 計算 AlarmLog 中每筆記錄的停機持續時間。
 - 使用 LINQ OrderByDescending() 依持續時間降冪排序。
 - 使用 Take(3) 選取前 3 筆記錄。
 - 準備一個包含 AlarmStartTime, AlarmEndTime (或顯示 "進行中"), DurationMinutes 的列表或 DataTable 以供 dgvLongestDowntime 顯示。
 7. 已完成工單分析:
 - 從 ProductionLog 找出該機台所有 LogType == "完成生產" 的 WorkOrderID。
 - 對於每一個已完成的 WorkOrderID:
 - 查詢 ProductionLog 取得該工單的 "開始生產" 時間和 "完成生產" 時間，計算實際完成時間(分)。
 - 查詢 WorkOrders 取得該工單的 ProductName 和 CostTime
 - 查詢 ProductionLog 撈取該 WorkOrderID 在 LogType="生產" 時的所有 GoodCount 和 BadCount 的紀錄，計算該工單良率。

- 準備一個包含 WorkOrderID, ProductName, ActualTimeMinutes, TargetTimeMinutes, TimeDifference, YieldRate 的列表或 DataTable 以供 dgvCompletedWOs 顯示。

乙、顯示結果：

- i. 將計算出的 totalGood, totalBad, YieldRate (格式化為百分比), alarmCount, totalDowntimeMinutes 顯示在對應的 Label 上。
- ii. 將最長停機事件的前 3 名顯示在 dgvLongestDowntime DataGridView 中。
- iii. 將所有已完成工單的分析結果顯示在 dgvCompletedWOs DataGridView 中。