**使用技術與工具**

1. **圖形界面設計 (GUI)**：
   * 使用 tkinter 套件構建 GUI，並輔助以 ttk 提供更現代化的視覺樣式。
   * 包括 Canvas 與滾動條的組合用來呈現動態內容。
2. **資料處理**：
   * 整合資料庫操作 (透過 WordDatas 類別)。
   * 包括單字的新增、查詢、修改與刪除。
3. **功能性頁籤 (Tabs)**：
   * Notebook 用於分頁設計，實現查詢頁面與已學習單字頁面的功能切換。
4. **事件處理**：
   * 使用 bind 綁定滾輪滾動事件及清單點擊事件。
   * 透過按鈕回調實現資料查詢與互動操作。
5. **視窗管理**：
   * 使用 Toplevel 創建模態窗口，處理更詳細的資料檢視與編輯功

**使用 TTK 開發翻譯工具心得分享**

在學習程式設計的過程中，我嘗試結合 Python 的 tkinter 與 ttk 工具，開發了一個簡單實用的翻譯小工具。這是一個關於 GUI 開發、資料庫應用，以及如何提升使用者體驗的實踐案例。

**專案概述**

此工具的目的是幫助使用者快速查詢英文單字及其翻譯，並能將學習過的單字儲存起來供後續複習。主要功能包含：

1. 查詢單字與片語翻譯。
2. 動態呈現單字的詳細資訊，例如詞性、例句、翻譯。
3. 已學習單字的管理功能，包括新增、修改與刪除。

**技術挑戰與實現**

**1. GUI 設計**

透過 tkinter，我設計了一個直觀的介面。Canvas 與滾動條的結合讓資料顯示更靈活，而 ttk 的 Notebook 分頁使得界面整潔有序。

**2. 資料管理**

利用 SQLite 整理學習資料，並設計了資料表來保存學習記錄。為了讓操作方便，我設計了封裝的查詢與更新函數，確保資料處理的正確性與穩定性。

**3. 使用者互動**

為了提升工具的友好性，我實現了以下功能：

* 即時查詢與錯誤提示。
* 點擊歷史記錄即自動查詢。
* 使用模態窗口集中處理編輯與刪除操作。

**心得體會**

這次的開發過程讓我對 GUI實作的設計與資料庫操作有了更深入的理解。特別是在處理資料呈現的過程。包含Canvas的彈性與滾動條的搭配、pack( )和grid( )方法的使用差異。資料庫操作及資料處理方面，則是使用到『氣泡排序演算法』處理資料，讓前端介面鋪陳資料時，有更好的處理資料效率。

未來，我希望能進一步改進此工具，例如增加語音朗讀功能、整合 API 翻譯文章功能或是隨機單字小考功能，讓學習者有更完整的工具支援。