第六組

**作業說明：**

學習目標：

1. 前端技術

**HTML**：學習如何使用HTML來構建網頁的結構。

**CSS**：學習如何使用CSS來美化和佈局網頁。

1. 後端技術

**Python**：學習Python的基礎知識，特別是涉及網頁開發的部分。

**Flask**：學習如何使用Flask來構建Web應用。

**SQLite**：學習如何使用SQLite作為數據庫，存儲和檢索應用的數據。

預期效益：

能完成一個有基本功能，如:創建帳號、修改帳號、上架商品、購物車、購買商品、購買紀錄及訂單紀錄的B2C電子商務平台。

**背景資料：**

Python的使用

使用Python的Flask進行網頁開發

使用Python的SQLite進行資料庫的應用

問題的介紹

1. **用戶界面設計**：需要設計美觀、易用的界面來吸引和留住用戶。後端挑戰
2. **數據庫設計與管理**：設計合理的數據庫結構來存儲用戶、商品和訂單等信息，並確保數據的完整性和一致性。
3. **用戶認證與授權**：實現安全可靠的用戶註冊、登錄和權限管理。
4. **業務邏輯實現**：處理訂單創建、支付處理、商品上架和管理等業務邏輯。

相關的知識

1. HTML：用於構建網頁的基本結構。
2. CSS：用於美化和佈局網頁。
3. Python：後端編程語言。
4. Flask/Django：Web框架，用於處理HTTP請求、路由、模板渲染和數據庫交互。
5. SQLite：輕量級關係型數據庫，適合小型應用。

**環境介紹：**

系統需求

|  |  |
| --- | --- |
| 項目 | 建議規格 |
| 作業系統 | Windows 8.1 以上 |
| CPU | Intel Core i5以上 |
| 記憶體 | 8GB以上 |
| 顯示卡 | 內顯即可 |
| 硬碟容量 | 可儲存空間500MB以上 |

使用工具

1. **VSCode**：多功能且輕量級的代碼編輯器，支持眾多擴展和插件。
2. **Git**：分布式版本控制系統，用於追蹤代碼變更和協作開發。
3. **Flask**：Python微框架，用於快速開發Web應用。
4. **Django**：全功能的Python Web框架，適合大型應用。
5. **SQLite**：輕量級數據庫管理系統，適合小型到中型應用。
6. **HTML/CSS**：用於設計和佈局網頁。
7. **DB Browser for SQLite**：圖形界面工具，用於查看和管理SQLite數據庫。

**實作方法：**

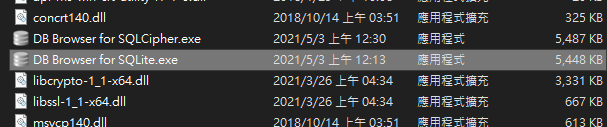
環境檢查、安裝與設定：

安裝SQLite管理工具

先至[官網](http://sqlitebrowser.org/)下載程式，安裝版或免安裝版皆可。



解壓縮後執行主程式：DB Browser for SQLite.exe

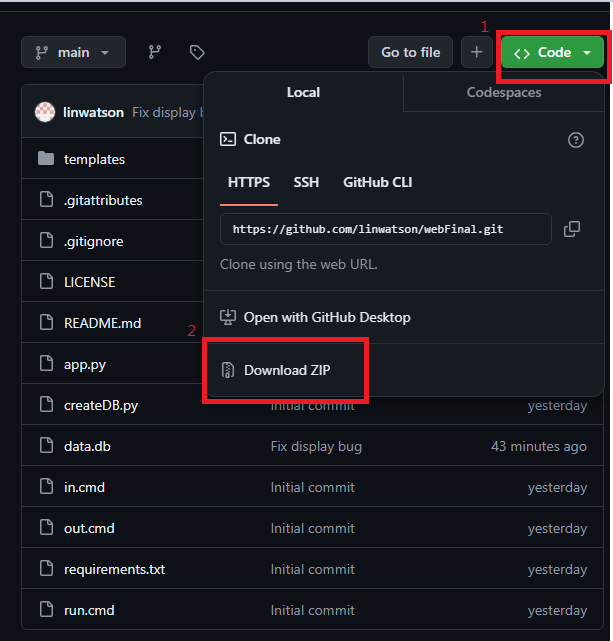


安裝Python 3.12  
先至python[官網](https://www.python.org/downloads/)下載對應的OS版本

後至命令提示字符輸入”python --version”查看當前的python版本

在命令提示字符輸入pip --version查看pip版本

到github專案網址將專案clone下來



將專案解壓縮之後

在命令提示字符下輸入

Cd 專案的絕對路徑

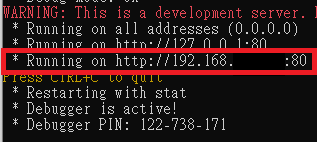
python -m -venv env創建虛擬環境

env\Scripts\activate 進入虛擬環境

pip install -r requirements.txt安裝執行環境所需要的pip包

python createDB.py 建立資料庫

flask run -h 0.0.0.0 -p 80 執行flask程式



將這段https網址 貼在瀏覽器

解決方案步驟

設計項目結構

webFinal/

│

├app.py

├createDB.py

├requirements.txt

├error.log

└─templates/

├cart.html

├editProfile.html

├error.html

├index.html

├login.html

├myAccount.html

├orderHistory.html

├pendingOrders.html

├product.html

├publish.html

├purchaseComplete.html

├register.html

├seller.html

└soldProduct.html

建立數據庫

開發主要功能

用戶認證：實現用戶註冊、登錄和退出功能。

商品管理：實現商品上架、編輯和刪除功能。

購物車：實現添加商品到購物車、查看購物車內容和移除商品功能。

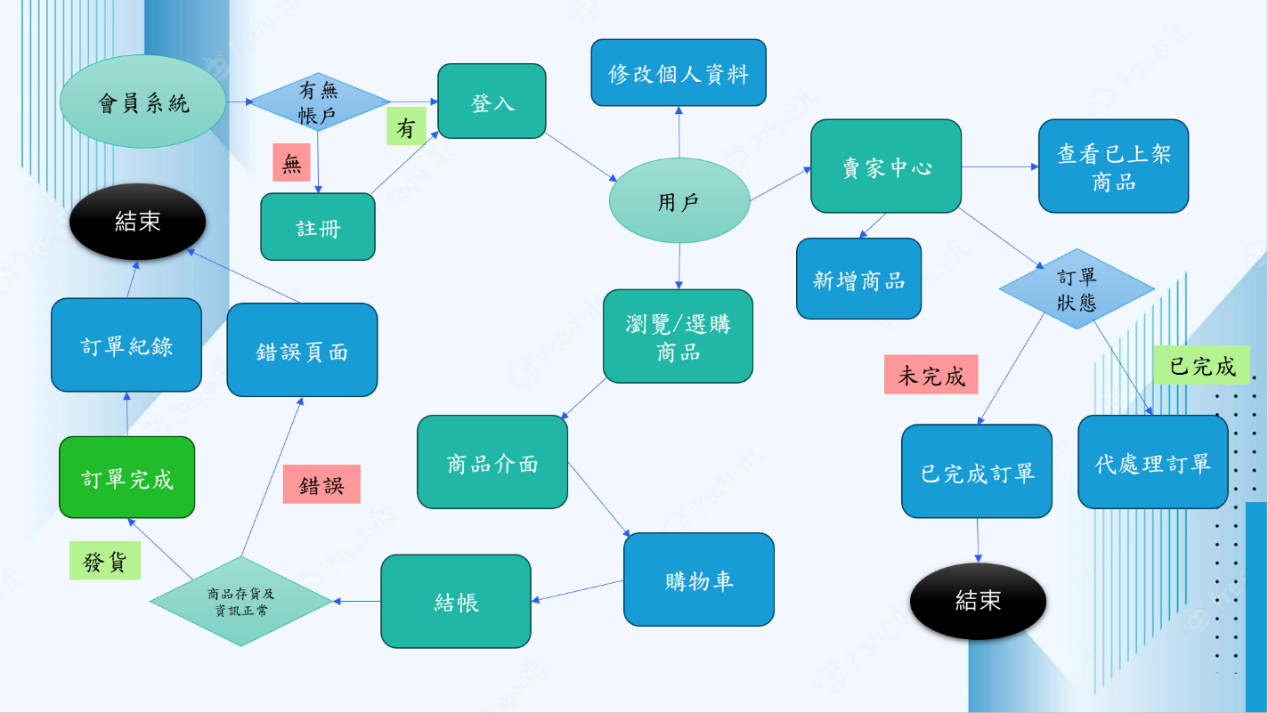
訂單處理：實現創建訂單、查看購買紀錄和訂單詳情功能。

設計前端頁面

設計前端頁面

使用HTML、CSS設計和實現各個網頁。

流程圖



解決方案中所運用之技巧

資料庫:

原本是利用session的方式儲存所有資料 但後來發現關閉硬體後session的資料就不見了，所以利用資料庫去做session的存取

其中包含主鍵與外來鍵的應用在使用主鍵與外來鍵的時候發生FOREIGN KEY constraint failed 造成FOREIGN KEY constraint failed的原因有五種：

1. 引用的外鍵不存在

2. 目標表的主鍵或唯一鍵缺失

3. 外鍵約束錯誤地定義

4. 插入或更新的順序問題

5. 外鍵約束處於禁用狀態

解決方法：

PRAGMA foreign\_keys = ON; 使用外來鍵約束

以及在建立資料表的時候在定義外來鍵的時候加上這行”ON DELETE CASCADE ON UPDATE CASCADE” 意思是 當外來鍵的的參照表被更新或刪除時 自動更新相關的行

頁面的導向問題； 在同樣是post請求的時候 利用給目標加上一個name，當後端python接收到的時候除了判斷request.method外，多判斷一個

name = request.form.get('name’)

頁面的顯示問題：

在選取多個產品的時候 在前端html會有相同product\_id沒辦法重複顯示的問題，解決方法：

在後端python將產品列表打包成一個dict後，利用列表生成式，將dict轉換成可以傳入前端html的格式

**實作心得：**

鮑柏丞：透過構建一個簡單的電子商務平台，可以綜合應用課堂所學的各種知識，提升實戰能力，我覺得參加此課程受益良多。

余芃節：我認為這次的期末作業幫我複習了html、css及sql等程式語言，將過去擁有的網站由php改到Python，以前的我只會使用php 撰寫簡單的網站，我覺得學到用Python做網站是一個很棒的方式，這門課學到了網頁中最重要的架構；遇到困難團隊裡面也有實作能力強的同學把東西做出來，與分組的同學相處很融洽。

林威盛：在這次的分組報告中，我體會到小組合作的重要性。在撰寫程式碼中，遇到了許多的bug，也讓我再次體會到解決問題的重要性，小組合作讓我們能夠集思廣益，互相補充彼此的不足。在這次報告中，我們每個成員都發揮了自己的特長，有的負責程式撰寫，有的負責PPT製作，還有的負責最後的口頭報告。讓每個人都有機會展示自己的能力。

**附註：**

工作分配：

|  |  |  |
| --- | --- | --- |
| 3B117042林威盛 | 3B117027鮑柏丞 | 3A917102余芃節 |
| 程式撰寫、設計、文件製作 | 簡報、文件製作 | 報告 |

文獻參考:

[在 Windows 上使用 Python — Python 3.12.4 說明文件](https://docs.python.org/zh-tw/3/using/windows.html)

UI Snippets([UI-snippets | Checkboxes](https://app.ui-snippets.com/checkboxes/green-check))

W3Schools([W3Schools Online Web Tutorials](https://www.w3schools.com/))

Pchome([PChome 24h購物](https://24h.pchome.com.tw/))

Shopee([shopee.tw](https://shopee.tw/))

8591([8591寶物交易網 - 全台最大遊戲交易平台，安全又優惠](https://www.8591.com.tw/))

Chatgpt([ChatGPT | OpenAI](https://openai.com/chatgpt/))

老師的上課講義