

## 組員

系級	學號	姓名
工海系大二	B08505007	黃仟如
工海系大二	B08505024	劉虹伶
工海系大二	B08505029	黃郁潔
財金系大四	B06703112	黃品臻
電機系大四	B03901193	姜宇謙

## 主題

本次指定題的主題為高鐵訂票系統，我們參考高鐵網站的訂票流程與計費方式，功能涵蓋訂票、退票、修改訂位人數、查詢時刻表、查詢優惠車次、查詢訂位狀態。票種涵蓋一般全票、商務艙、大學生優惠票、來回票以及早鳥票。

## 系統架構及功能簡介

此系統包含的功能有查詢當日時刻表、查詢優惠車次、訂票、退票/修改訂位人數、查詢訂位狀態的功能，並藉由使用者與終端機互動的方式，提供簡單明瞭的操作過程。

以下為功能介紹:使用者進入系統可執行三大功能:訂票、退票/修改人數、查詢。

(1)如果選擇的是訂票的功能，則會引導使用者輸入搭乘日期/起訖站名 /搭乘時間，輸入完成後系統會顯示五筆在搭乘時間以後的車次，內容包含起訖站名、發車時間以及到站時間。使用者可以輸入車次以及艙等，再決定訂票的數量(如果所選艙等為商務艙，則不提供大學生票的選擇)。上述步驟完成後，系統會依照所選車次以及艙等分配座位，並列印出所有票務資訊，包含搭乘時間、起訖站名與發車抵達時間、車次編號、票種/數量、繳款金額(依照所訂購之艙等或票種計算)、座位號碼以及訂位代碼。待使用者輸入使用者 ID，即完成所有訂票手續。

(2)根據台灣高鐵網站退票以及修改人數的規定，退票即是取消所有訂位資料，且如果訂購的是來回票不可單獨退掉單向行程；修改人數即是減少訂購人數，如果訂購的是來回票，會自動取消來回雙向的行程。如果選擇的是退票的功能，使用者需要輸入使用者 ID 以及訂位代碼，即可完成退票手續。如果選擇的是修改人數的更能則是需要輸入使用者 ID 需要輸入使用者 ID 以及訂位代碼，訂位代碼以及欲減少人數之人數，系統會顯示剩餘人數的票務資訊，包含搭乘時間、起訖站名與發車抵達時間、車次編號、修改後之票種/數量以及修改後的座位號碼。

(3)查詢功能主要為查詢訂位紀錄以及訂位代號，查詢使用的狀況分為二種，分別為有使用者 ID 和訂位代號的情況下，查詢詳細的訂位紀錄以及在缺少訂位代

號的情況下，藉由使用者 ID 和起訖站名、日期、車次號碼，查詢訂位代號以及訂位資訊。此外，查詢還有查詢當日時刻表，查詢當日優惠車次的功能。

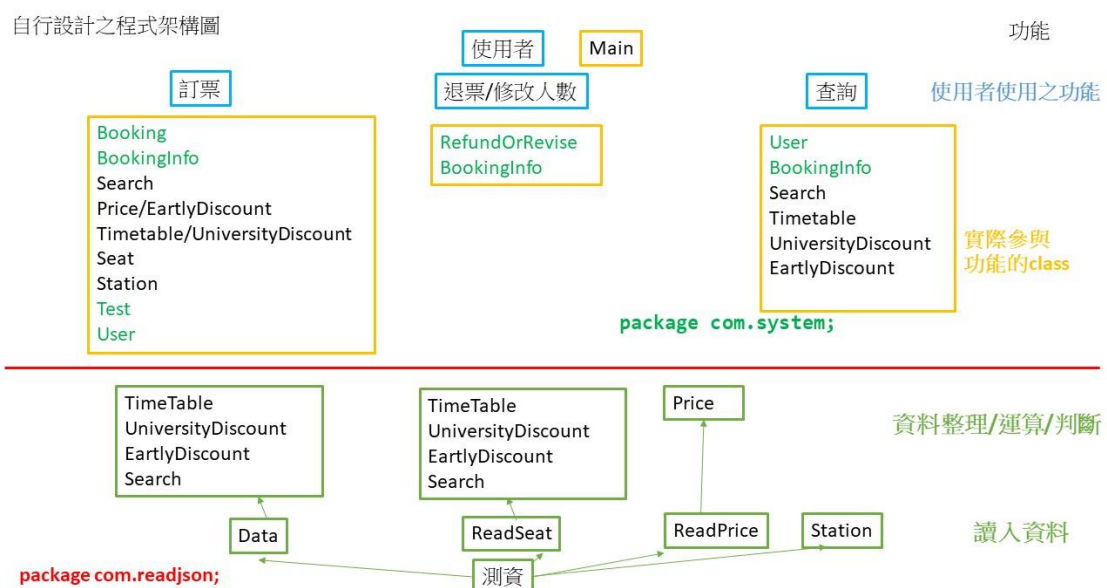
## 使用到的套件

使用 org.json 套件處理 json 檔案的匯入，並將所需的資料庫以 json.jar 匯入程式中自創的 lib 資料夾中

## 流程圖

下圖為自行設計之程式流程圖：

主要是想要藉由自己設計的流程圖展現參與在各個功能中的 class 分別有哪些，以及使用者和測資分別是透過系統中的哪些 Class 進入系統。



藍色所代表的是以使用者視角可以看到之外部程式的功能，例如三大主要功能：訂票、退票/修改人數以及查詢。在功能下方橘色範圍內的則是代表著參與各功能的 Class，不同的功能也會使用同一個 Class 的內容在其中運算以及更正。在紅色線下方則是將外部資料(助教提供之測資)讀入程式的過程：讀入的過程有兩階段，運用 org.json 讀入程式，將如入資料選取所適合的資料在分配進到不同 Class 中有系統地整理，再在 Class 中進行判斷以及運算。

訂票、退票/修改人數以及查詢為三個獨立的功能，在系統不關掉重置的情況下，建議先進行訂票的動作，待系統有紀錄之後，再進行查詢或者退票修改。

## Package 規劃(Optional)

可簡單附上 package 的樹狀圖, 或是繪製 class diagram

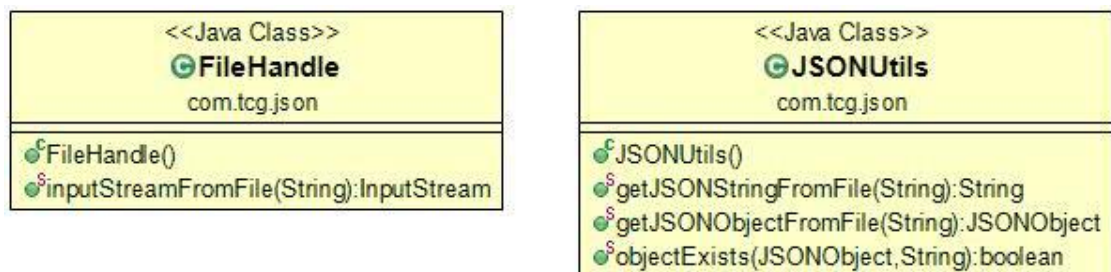
在我們的高鐵訂票系統中總共有 4 個 package，分別為 package com.system、package com.readjson、package com.exception、package com.tcg.json。

(1) package com.exception 管理的是所有可能產生的 exception，分別有 BookingException、ClassException、DataException、LeftSeatException、

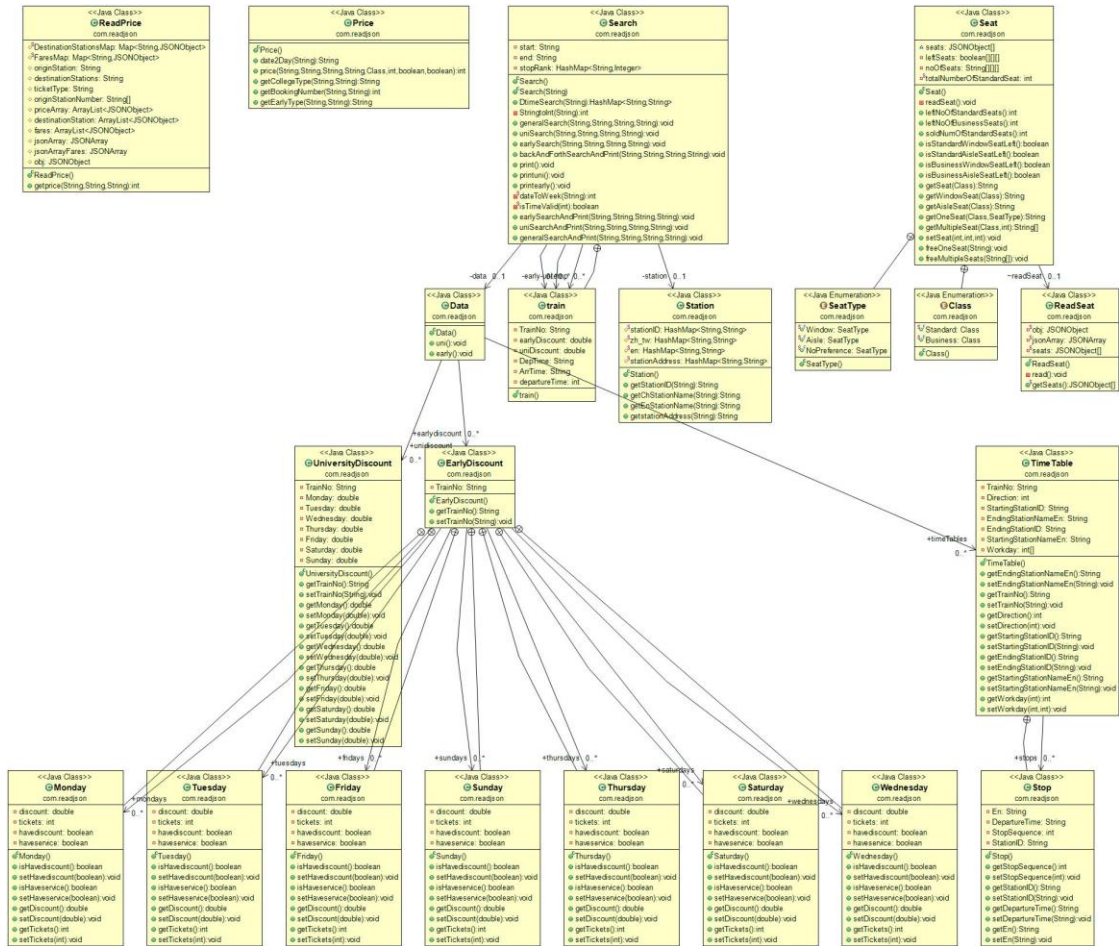
NoOfTicketException、OrderNoException、OrderInfoException、PriceException、PrefException、ReviseException、TimeException、StationNameException、UserException)，為了使所有 Class 版面清楚、容易辨識，將所有例外處理放在同一個 package，在需要時引入 class 使用。



(2) package `com.tcg.json` 則是管理處理與 `org.json` 套件相關的 Class，作為讀入資料之用。

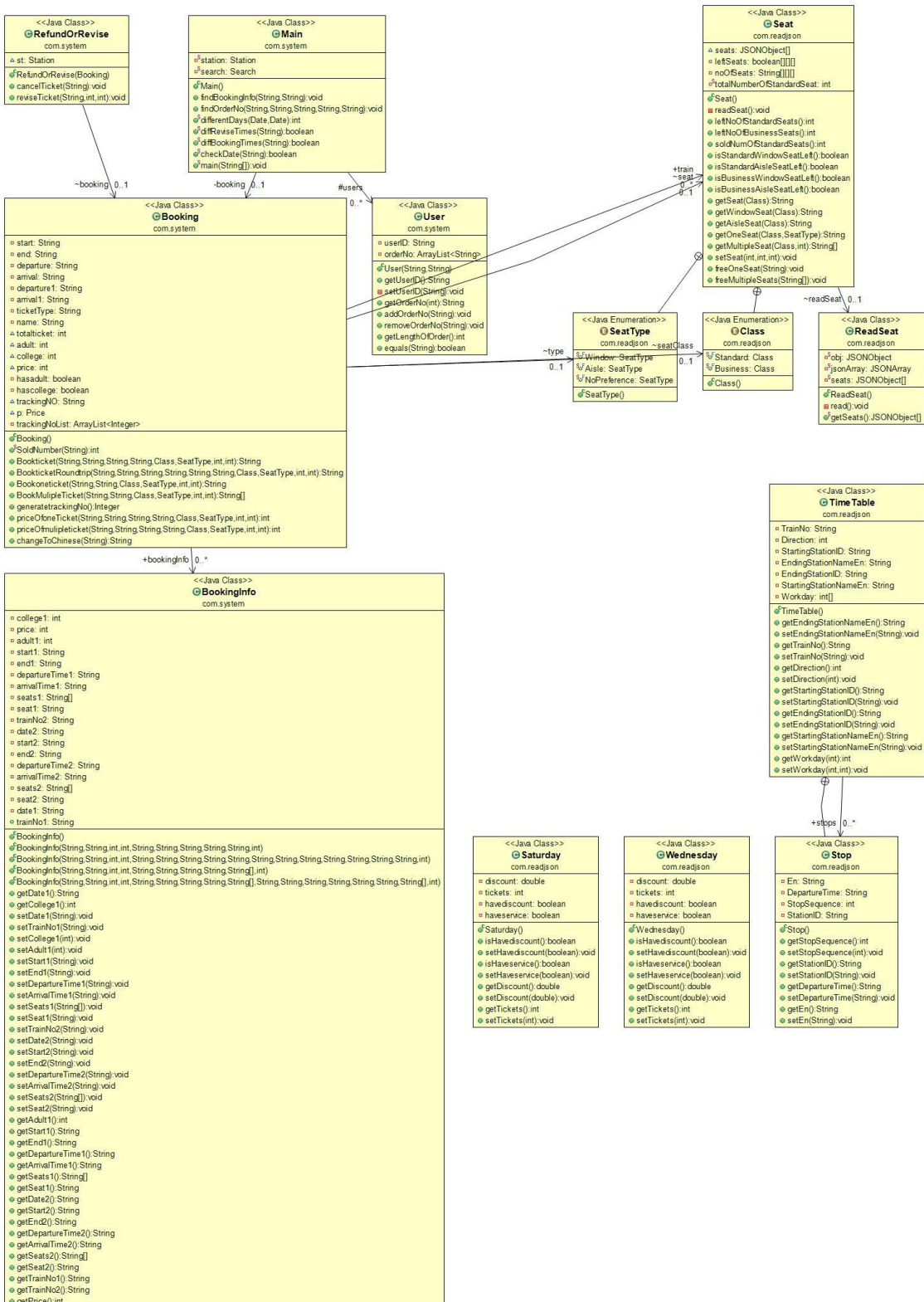


(3) package `com.readjson` 管理的是由 `.json` 檔讀入資料的 Class，以及執行跟 `.json` 檔相關的判斷以及運算。其中 `Data`、`EarlyDiscount`、`UniversityDiscount`、`TimeTable` 處理的多是跟列車班次以及優惠相關；`ReadSeat`、`Seat` 處理的是讀入列車座位資料，以及將處理分配、釋放座位、判斷座位數量；`ReadPrice`、`Price`、`Station` 處理的則是票價計算相關的內容。



(4) package com. system 則是搭配高鐵系統所需要的功能，組合其他各個 Class 資料，進行判斷以及輸出。包含 Boooking、BookingInfo、RefundOrRevise、Main、test、User，多數跟存取訂票、退票、修改、使用者相關的內容。





## Design Pattern(Optional)

(1) 在 Search 中使用了自行定義的 comparator 達到 Sorting 的目的。因為在使用 Search 的功能時，會需要將班次結果依照發車時間按順序排列，自行定義的 comparator 可以自動將 Class 中的特定變數進行排列。如果不使用這項功

能，會需要自己實作一個 Quick sort 或是 Merge Sort，才能達到  $O(n \log n)$  的演算效率，也有較高的可能性會因為使用 Quick sort 或是 Merge Sort 出現錯誤，程式也會較為複雜。

- (2) 在 Seat 中，使用一個三維陣列管理列車座位的狀態，用 boolean 函數代表其是否尚未被訂走。如果是用 class 加上兩層 inner class 會使要尋找某一位置時，尋找變數的過程較為麻煩。使用陣列讓程式能夠以多層迴圈就能執行判斷。在選擇釋放座位時，使用傳入欲釋放的座位的方式釋放指定的座位，並且改動迴圈中的 boolean 達到釋放座位的目的，這個方法使得釋放座位的過程(用在之後的退票)，變得較為容易，且不容易錯誤，或者不知道系統自動釋放的座位是哪一個。
- (3) 在 Booking 中，將訂票的動作分解成訂一張票以及訂多張票，搭配了 Seat 中的分配一個位置以及分配多個位置。這個方法使得處理來回票，較為容易，因為將來回票視為執行兩次的訂票動作，而回程票的訂購則是執行起訖站對調後的訂票。
- (4) Exception 詳盡的設計也是確保程式能夠正確運行的重要環節，因為我們的程式並未使用資料庫的功能，因此程式不能隨意地被結束，才能將我們的訂票紀錄都保存下來，以供其他功能使用。因此，exception 的設計需要考慮到非常多不同的可能性，像是輸入格式的例外，因為時間無法訂購的例外，票種與數量的各種例外都，所以我們加入了十多個 Exception 以確保程式可以順暢的運作，在遇到例外問題程式不會報錯結束，而是被例外處理準確地接住。因此我們在設計使用者與終端機互動之功能中，加入了返回的功能，在遇到例外時以返回功能使得程式可以繼續運作。

## 分工表

姓名	學號	占比	工作細項
黃仟如	B08505007	30%	com.exception: ClassException DateException LeftSeatException NoOfTicketException OrderInfoException OrderNoException PrefException PriceException ReviseException StationNameException TimeException UserException com.readjson: ReadSeat Seat Station com.system: User Main
劉虹伶	B08505024	15%	Data、search、timeTable
黃郁潔	B08505029	17.5%	Price PriceException ReadPrice ReadStation
黃品臻	B06703112	20%	Data TimeTable Booking BookingInfo RefundOrRevise BookingException
姜宇謙	B03901193	17.5%	search, data, timeTable

## 心得(optional)

姜宇謙：

很開心能有機會和組員們合作這個 project，大家都很認真，也很願意學習。從一開始有點不知從何下手，我們逐步分析目標，到把大目標分成小目標，再慢慢組合成最後的結果，我覺得學到最多的就是把大問題 modularize 成小問題後，每個小問題的 solution 要怎麼互相結合。

舉例來說，search 的部分，就必須要先有 parser，data 讀得進來，才能做各種不同條件的 search。我們在 parser 的部分花了不少時間，也感謝教授在星期一的視訊特別給我們的指導。我在 search 的 constructor 中用了 hashmap 來記錄每個 stop 的順序，這樣 input 起迄站，就可以判斷是南下還是北上，再去 search timetable 中符合的班次。也要做一些 search 的功能讓 booking 可以用到。很可惜這次時間不太夠，希望之後能有機會把 Data base / UI 也做出來。

劉虹伶：

指定題的作業量非常龐大，json 檔的數量跟數據多到讓人眼花撩亂，擔心自己能力不足拖累大家，也因此學期初就讓我感到相當的焦慮。在撰寫程式的最前期遇上了很多困難，從讀取 json 檔的資訊就遇上了諸多障礙，還用了一個小時和老師請教這個問題，後來的幾次實際經驗後慢慢學習，培養了很多耐心，也發現打程式這件事情除了能力以外，更需要無限的耐心去完成，常常在電腦前一坐就是三到六小時。感覺經過這一次 project 後自己的自學能力提升了不少。在一開始迷惑時，直屬學長問過我一個問題，他說：「你有辦法查出你需要知道的東西嗎？」這句話聽來饒口，但給了我很大的啟發。時間因素之下，最後還是來不及完成很想要的爬蟲和做出漂漂的 GUI 介面，暑假會來自己嘗試看看。

很開心能和大家一起做事情，團隊都為了完成這份專題付出很多心力，在很多時候覺得被各個組員罩，不知道自己有沒有做好該做的事情。在討論時，我常常會當說話特別多的、讓討論流程接續下去的那一方，也會擔心自己是不是太自作主張的做了多餘的事情，但還是有慢慢地完成了自選與指定的兩個 project，很感謝團隊成員的包容和很大很大的心力。

黃仟如：

指定題真的是非常複雜也很多需要注意的小細節，從一開始讀進 json 檔就遇到了不少問題而卡關許久。而在我們能順利讀進 json 檔之後我接下了整理 seat.json 的工作，沒想到真是個地獄，整天在三層迴圈內轉來轉去，轉得我暈頭轉向。但做出來後真的蠻有成就感的，尤其被說 class 設計得很完整、method 很好用更是開心。後來還做了查詢訂單資訊的 method 和整個 main method，要把所有 class 的資訊好好統整起來使用也是很大的挑戰，中間也遇到了不少麻煩，而且因為我們沒時間再去做 GUI 的部分，整個跟終端機互動的流程設計起來也很繁雜很累，稍有個失誤在跑程式的時候可能就會因為沒抓到 Exception 而使程式停止，測試的訂單資料就會全部不見。在整個設計的過程中自己查了很多相關資料，除了別人部落格上整理好的、Stack Overflow 上別人的討論串之外，也看了很多份 Java SE Document，很多東西真的找不到解決辦法的時候仔細看 Document 竟然真的就找到問題而且自己就解決了，教授在課堂中教過最讓我覺得受用的大概就是看 Document 的能力了吧！雖然這次礙於時間無法將 GUI 和資料庫實現，但之後有機會也希望能夠做做看！

黃郁潔：

在指定題中，因為設計上要使用的東西較為複雜，所以在設計時遇到的問題也較自選題多。在疫情開始前我們就開始了討論，然而對第一次做專題的我們，下手時就相當茫然，於是我們便開始分析架構，希望能從中找出最適合下手的地方。然而在我們正式開始寫程式前，疫情就爆發了，使我們各自回到了家中，而我們的一切工作，也就變得完全的線上化，線上開會、工作分配、互助合作，希望在防疫生活中也能順利地完成專題。但首先，匯入檔案就使我們很大的一個關卡，因為沒有相關經驗，在網路上搜尋無果（有資料，但我們嘗試過後並沒有成功），所以還在提問時間中向老師詢問如何操作檔案，在老師的耐心講解後，我們終於有了對操作的初步認識，才能順利地展開接下來的專題製作。因為先前就有將要用的功能，大致該有的樣子討論出來，所以也順利的找到了可以最先編程的地方，大家就各自認領了一部分，開始了作業。大致上我們遇到的問題也都能夠在合作討論中解決，但其實在過程中我也發現了很多我們身為新手，在一開始可能不太會注意到的小細節。個人在進行編碼時，因為思路是一脈相承的，所以當獲取一份資料，便可以以最理想的方式設計是否要回傳、回傳型態等等，還是只是將資料儲存在這個 Class 中，但在分工之後，其實有時候我們不容易知道，需要資料的人最終決定要以什麼方式使用，而在不想輕易的更動目前所完成部分的情況下（這也是因為若是我自己動了一部分，而其他組員有人沒發現，去更動自己已經有的檔案，接下來可能會出現很多錯誤，而這時候我們又尋找到底錯誤出在哪裡，會消耗更多精神），所以可能就要大費周章的，寫了一個長長的方法，只為了得到想要的資料。因此我們可以發現，若是能夠在更早的時候，就先確認從上游到下游的所有連鎖關係，將彼此所需要的型態、資訊樣式等等，都先明確地列出，就可以有效的避免這樣的情形，雖然可能多花費了一些時間，但這也是我在專題中學到的珍貴經驗。

黃品臻：

在寫這次指定題的過程是我覺得是相當挑戰的，題目本身就有相當的難度，因為需要考慮到的使用狀況非常複雜，而且先前完全沒有人合作一起寫程式的經驗，再加上在這遇到疫情的時候，不能見面討論也為報告添加了一些難度。從看懂測資然後正確的讀進資料就花上我們許多時間，如何準確的更新每個 Class 的進度讓所有組員都知道最新版本也讓我們苦惱很久。甚至連 class 要怎麼命名讓大家可以統一並且看懂也需要互相磨合，真的很感謝組員透過頻繁的開會討論，一步步找到適合彼此的工作模式，也可以在程式遇到困難時互相 cover，找到解決的方法。我覺得這次的團體作業中學到最多的是自學的能力，如何在各大網站找到自己可以套用的資料，再快速吸收內化，並且應用出來，雖然過程可以說是非常痛苦，常常在電腦前面一待就是一整天，但是越到後期越發現自己看參考網路上的 code 越來越輕鬆就可以讀懂。此外我覺得指定題還訓練了自己分析並且拆解問題的能力。我一開始是參與讀入班次資料的部分，這部分因為比較少需要跟其他 class 要資料因此思考方式可以較為直觀。但是當我開始負責訂票退票修改的部分，因為需要考慮的狀況變多，且需要清楚的知道要跟哪個 Class 要資料會比較容易，常常在電腦前面瘋狂默念起訖站對調就是回程的訂票，但是在一大堆相似名稱的變數面前，還是會出現給錯資料的問題。真的很感謝每個組員之間的互相協助和提醒才能完成這次作業。