

資料庫系統 Final Project III

主題：租車管理系統

組別：G06

組員：50915112 廖子科

51015107 廖玟嫻

51015130 林敬雅

中華民國 114 年 6 月 13 日

目錄

壹、 簡介.....	1
貳、 應用情境與使用案例.....	2
參、 系統需求說明.....	3
肆、 概念層模型(完整性限制).....	4
伍、 ER Diagram 及說明	11
5.1 ER Diagram	11
5.2 ER Diagram – Relationship	12
5.3 ER Diagram – Entity in SQL	14
5.4 ER Diagram – Relationship in SQL.....	27
5.5 ER Diagram – 合併 Table.....	35
陸、 VIEW / GRANT	42
1. guest_user	42
2. customer_user	46
3. staff_admin.....	49
柒、 完整 SQL 統整.....	52
DDL (schema).....	52
DML (record).....	55
DCL / VIEW (External Level).....	59
Transaction.....	65
捌、 系統結果.....	66
玖、 組員分工與心得.....	83
50915112 廖子科.....	83
51015107 廖玟嫻.....	84
51015130 林敬雅.....	85
壹拾、 參考文獻.....	86

壹、簡介

隨著共享經濟的興起，越來越多人選擇租車作為短期出行的解決方案。對於平時無需長期擁有汽車的人來說，購買一輛汽車不僅需要支付數十萬元，還需負擔定期稅金、保險費、保養維修費等長期開銷。因此，當有用車需求時，租車服務成為更為經濟的選擇。

租車管理系統主要為顧客提供線上租車服務。顧客可以透過系統瀏覽可租借車輛的清單，並根據自身需求選擇適合的車輛，提前進行線上預約。當租賃日到來時，顧客只需攜帶有效身分證件與駕照正本，辦理保險與租賃契約，即可順利取車，避免因現場車輛短缺而無法提供租車的情況發生。

此外，租車公司員工可透過系統管理車輛，以及租賃訂單的管理，進一步提升顧客體驗與服務品質。

貳、應用情境與使用案例

難得的連假，許多人都會想和家人或朋友一起出門透透氣，來一趟說走就走的小旅行。但並不是每個人都有自己的車，再加上不同旅伴、目的地與行程安排，常常會需要有更有彈性、能在異地取車還車的交通方式。這時候，透過租車平台預約一台合適的車輛，就成為了最方便又靈活的選擇。

以下為使用案例說明：

顧客 A	打開本系統，選擇在 4/3 早上十點在台中分店取車，4/6 下午四點在高雄分店還車。
系統	根據所選取的時間查詢可用車輛，回傳車型、租金與可選擇保險。
顧客 A	透過本系統預約了一台 Toyota Altis，選擇加購全險，獲得訂單編號。
系統	透過所選取的車型、時間、保險方案，生成租賃訂單。
顧客 A	選擇用 LINE Pay 進行付款。
系統	新增一筆對應的付款紀錄，將付款方式設定為 linepay。
員工	在後台查看顧客訂單，確認保險文件與車輛狀態，於 4/3 完成交車流程。
顧客 A	在現場出示駕照與身分證後取車。
系統	更新訂單狀態為 active。
顧客 A	4/6 晚上八點在高雄分店還車。
系統	系統判定逾期，計算額外罰金。
員工	檢查車輛確認功能無異常，確認車輛歸還。
系統	更新訂單狀態為 completed。

參、系統需求說明

針對租車管理系統，須具備以下功能性需求：

1. 會員註冊與登入

會員可以使用 Google 帳號進行註冊與登入(由後端程式串接 Google OAuth 2.0)，系統會檢查資料庫中是否已有該 Gmail。如果沒有，顧客將被導向註冊頁面，並需完成電話驗證確保未來聯絡方式，未來也可藉由電話號碼來擴充減少惡意訂單的功能。

2. 車輛瀏覽與選擇

顧客可以在系統中瀏覽所有可租借的車輛，並根據車種、燃料類型、期望租金範圍等條件進行篩選。每輛車會顯示詳細資訊，包括每日租金、逾期租金、車輛狀態、可乘坐人數等，讓顧客可以根據需求選擇適合的車輛。

3. 租車預約與個人管理功能

顧客可以選擇租借車輛並進行線上預約，選擇取車和還車的時間與地點，顧客可以查看自己的租賃訂單，包括租賃的車輛、租期、取車地點與還車地點。系統應提供取消訂單的功能，但需遵循時間限制(如:最慢於 5 天前能到系統修改訂單)。

4. 付款管理

顧客可以在租賃過程中選擇不同的付款方式，如現金、信用卡等。且需提供付款歷史記錄查詢功能，讓顧客可以查看過去的付款紀錄和交易明細。

5. 保險選擇

顧客可以提前了解租車公司提供的各種保險方案，每項保險方案會有詳細的說明，顧客可在租車時選擇附加保險。

6. 車輛與訂單狀態管理

租車公司員工可以查看並更新每輛車的狀態，也可以對租賃訂單進行處理。

7. 安全性功能

每位顧客只能看自己的租賃訂單，不能查看其他顧客的訂單資訊，保護顧客的隱私。

肆、概念層模型(完整性限制)

1. 會員資料表(member Table)

欄位名稱	資料型態	是否可為空	Domain	說明
<u>google_id</u>	String	否	Google OAuth 取得顧客識別碼， 為 21~22 碼數字字串。 正規表達式: <code>^[0-9]{21,22}\$</code> 例:106455916943881880515 型態長度最多為 22	*PK (Google ID)
gmail	String	否	符合電子郵件格式: 帳號@網域。使用者名稱(@前)長度需為 6~30 個字元，僅能包含: 英文字母、數字、句點(.)、底線(_)、百分比(%)、加號(+)、減號(-)。網域名稱(@後)需為有效網域格式，例: gmail.com 、 yahoo.com.tw 等。 正規表達式: <code>^[a-zA-Z0-9._%+-]{6,30}@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\$</code> 例:user_name@example.com 型態長度最多為 50	UK 會員信箱
phone	String	否	符合行動電話號碼，共 10 位數字字串。 正規表達式: <code>^09\d{8}\\$</code> 。 例: 0918202667 型態長度最多為 10	UK 會員電話

2. 地點資料表(location Table)

欄位名稱	資料型態	是否可為空	Domain	說明
<u>loc_id</u>	String	否	4 個大寫英文字組成。 正規表達式: <code>^[A-Z]{4}\$</code> 例: YUNH 型態長度最多為 4	*PK 分店代號
loc_name	String	否	2~5 個中英文字+「店」字所組成 正規表達式: <code>^-[一-龜 A-Za-z]{2,5}店\$</code> 例: 雲虎旗鑑店 型態長度最多為 6	分店名稱
city	String	否	必須為國內的縣市中文名。 正規表達式: <code>^(臺北市 新北市 桃園市 臺中市 臺南市 高雄市 基隆市 新竹市 嘉義市 新竹縣 苗栗縣 彰化縣 南投縣 雲林縣 嘉義縣 屏東縣 宜蘭縣 花蓮縣 臺東縣 澎湖縣 金門縣 連江縣)\$</code> 例:雲林縣 型態長度最多為 3	縣市
district	String	否	1~3 字鄉鎮名+[鄉/鎮/市/區] 正規表達式: <code>^-[一-龜]{1,3}(鄉 鎮 市 區)\$</code> 例:阿里山鄉、東區 型態長度最多為 4	鄉鎮市區
address	String	否	符合國內街道規定。 <code>^([一-龜]{1,3}(村 里))?</code> <code>([0-9]{1,3}鄰)?</code> <code>[一-龜]+(路 街 大道)</code> <code>(一二三四五六七八九十){1,3}段)?</code> <code>([0-9]+巷)?([0-9]+弄)?</code> <code>([0-9]{1,4}之)?[0-9]{1,4}號</code> <code>([0-9]{1,3}樓(之[0-9]+)?)?</code> <code>(^-[一-龜 A-Za-z0-9]+室)?\$</code>	街道地址

			例:忠孝東路四段 200 號 11 樓 型態長度最多為 30	
--	--	--	-----------------------------------	--

3. 保險方案資料表(insurance Table)

欄位名稱	資料型態	是否可為空	Domain	說明
<u>insurance_id</u>	String	否	7 個大寫英文、數字組成 正規表達式: ^A-Z0-9]{7}\$ 例:Y25S001 型態長度最多為 7	*PK 保險代碼
ins_name	String	否	3~15 個中英字或括號 正規表達式: ^-[A-Za-z()]{3,15}\$ 例:第三人責任升級 型態長度最多為 15	UK 保險名稱
coverage	String	否	1~250 個中文字或特定符號 正規表達式: ^-[A-Za-z0-9(),。、():][1,250]\$ 例:「」 型態長度最多為 250	保險項目
ins_fee	Integer	否	正整數(>=0)	每日保險費用， 0 元為免費方案

4. 車型資料表(Model Table)

欄位名稱	資料型態	是否可為空	Domain	說明
<u>model_id</u>	String	否	7位大寫英文、數字組成 正規表達式: <code>^[A-Z0-9]{7}\$</code> 例: MITGD01 型態長度最多為 7	*PK 車型編碼
brand	String	否	1~20 個大小寫英文組成 正規表達式: <code>^[A-Za-z]{1,20}\$</code> 例: Volkswagen 型態長度最多為 20	汽車品牌
model_name	String	否	1~20 個大小寫英文、數字、空白組成 正規表達式: <code>^[A-Za-z0-9 -]{1,20}\$</code> 例: Grand Lancer 型態長度最多為 20	車型名稱
car_type	String	否	限定字串: Compact、Sedan、 SUV、MPV 型態長度最多為 7	車輛類型
fuel_type	String	否	限定字串: Gasoline、Electric、 Hybrid 型態長度最多為 8	燃油類型
engine_cc	Integer	否	非負整數($>=0$)	引擎排氣量(cc)
transmission	Integer	否	必須為 0 或 1	變速箱類型 0 表示手排 1 表示自排
image_url	String	否	1~50 個英文字、數字、底線組成， 且 img/開頭、.jpg 或.png 結尾 正規表達式: <code>^img\[A-Za-z0-9\]+\.jpg\$</code> 例: img/CMC_JSPACE.jpg 型態長度最多為 50	車型圖片位置

5. 汽車資料表(Car Table)

欄位名稱	資料型態	是否可為空	Domain	說明
vin	String	否	<p>符合 VIN 編碼標準 ISO 4030(可兼容 ISO3779): WMI 全球製造商代碼(第 1~3 碼)+VDS 車輛特徵碼(4~8 碼)+核對號碼(第 9 碼)+生產年份 (第 10 碼)+生產代號(第 11 碼)+VIS 車輛識別碼(12~17 碼)。共 17 碼字串。</p> <p>正規表示式:</p> $^{\text{[A-HJ-NPR-Za-hj-npr-z0-9]}\{8\}[\text{0-9X}][\text{A-HJ-NPR-Za-hj-npr-z0-9]}\{2\}[\text{0-9]}\{6\$}$ <p>型態長度最多為 17</p>	*PK 車輛識別碼
plate_number	String	否	<p>符合國內租賃車牌規定</p> <p>正規表示式:</p> $^{\text{R[A-Z]}\{2\}-[\text{0-35-9]}\{4\$}$ <p>型態長度最多為 8</p>	UK 車牌號碼
daily_fee	Integer	否	大於 0 正整數(>0)	每日租金
late_fee	Integer	否	大於 0 正整數(>0)	每小時逾期罰金
year_made	Integer	否	大於等於 1980 正整數 (>=1980)	製造年份
seat_num	Integer	否	大於 0 且小於 16 正整數(>0 且 <16)	可乘坐人數
color	String	否	<p>1~4 中文字+1「色」字</p> <p>正規表示式:</p> $^{\text{[-顛]}\{1,4\}\text{色\$}}$ <p>型態長度最多為 5</p>	車輛顏色
mileage	INT(6)	否	大於 0 正整數(>0)	里程數
car_status	String	否	限定字串: available、maintenance、disable 型態長度最多為 12	車輛狀態 預設為 available
notes	String	是	任意字串	車輛備註
model_id	String	否	參照 Model Table 中的 model_id	FK，車型
loc_id	String	否	參照 Location Table 中的 loc_id	FK，地點

6. 租賃訂單資料表(Rental Table)

欄位名稱	資料型態	是否可為空	Domain	說明
<u>google_id</u>	Integer	否	參照 Member Table 中的 google_id	*PK (與 created_at) FK , google id
<u>created_at</u>	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒	*PK (與 google_id) 訂單建立時間
start_date	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒	租賃開始時間
end_date	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒	預計結束時間
actual_pickup	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒 NULL 表示尚未取車	實際取車時間
actual_return	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒 NULL 表示尚未歸還	實際還車時間
rental_status	String	否	限定字串: pending 、 active 、 completed 、 cancelled 、 reject 型態長度最多為 9	租賃訂單狀態
amount	Integer	否	正整數(>0)	付款金額
penalty_fee	Integer	否	正整數(>=0) 預設為 0	總罰金 (逾期還車與錯誤地點還車的合計)
payment_date	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期	付款時間

			小時(24 小時制):分:秒	
method	String	否	限定字串：cash、credit、linepay 型態長度最多為 8	付款方式
car_vin	String	否	參照 Car Table 中的 car_vin	FK，汽車 id
pickup_loc	String	否	參照 Location Table 中的 loc_id	FK，取車地點
drop_loc	String	否	參照 Location Table 中的 loc_id	FK，預計還車地點
actual_drop_loc	String	是	參照 Location Table 中的 loc_id NULL 表示尚未歸還	FK，實際還車地點
insurance_id	String	否	參照 Insurance Table 中的 insurance_id	FK，保險代碼

伍、ER Diagram 及說明

5.1 ER Diagram

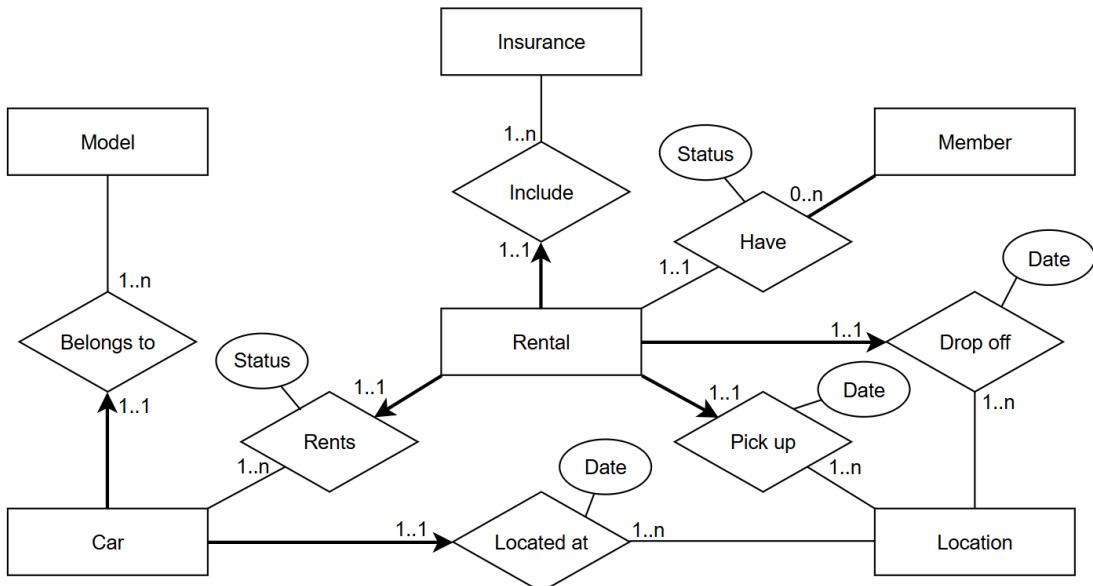


圖 1、詳細 ERD

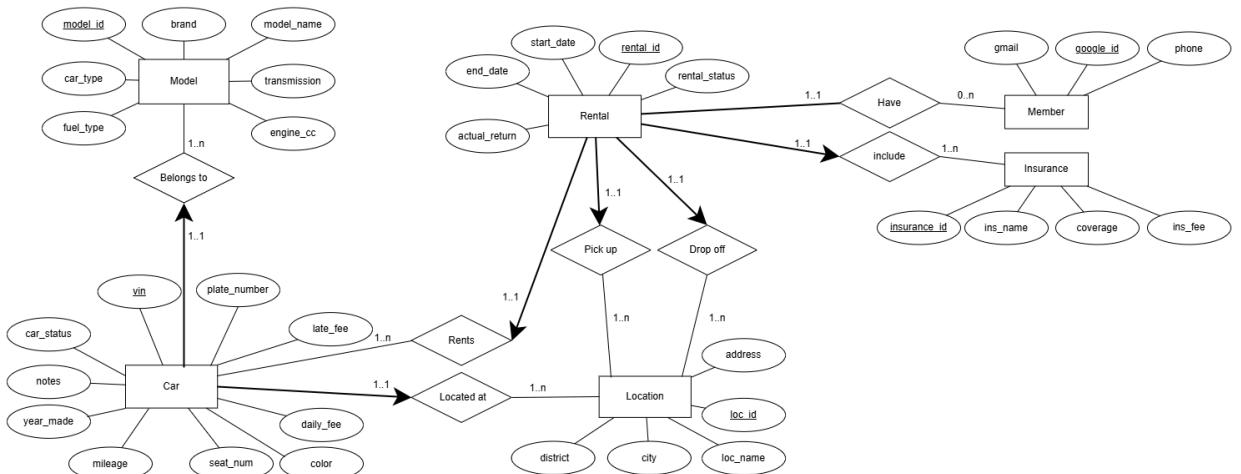
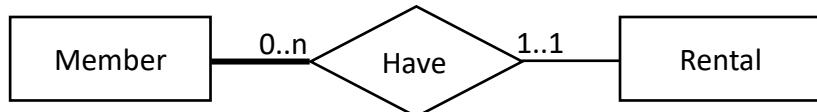


圖 2、完整 ERD

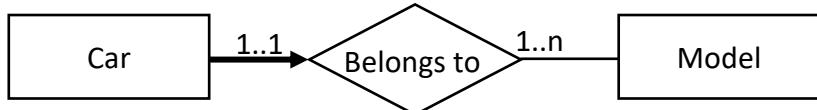
5.2 ER Diagram – Relationship

以下為各實體之間的關聯

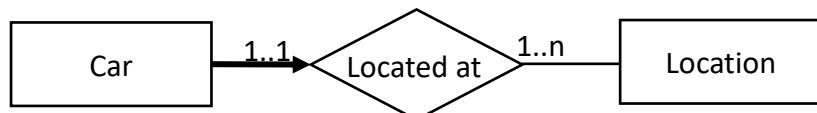
- Member (會員) 實體和 Rental (租賃訂單) 實體之間存在一對多的「擁有」關係：
 - 一位會員可以有多筆租賃訂單
 - 每筆租賃訂單只會對應到一位會員



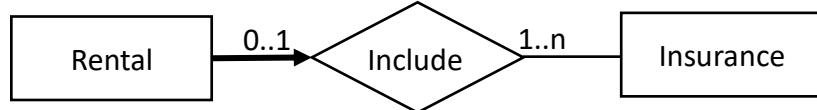
- Car (汽車) 實體和 Model(車型) 實體之間存在多對一的「屬於」關係：
 - 一個 Model 可以有多輛 Car
 - 每輛 Car 只屬於一個 Model



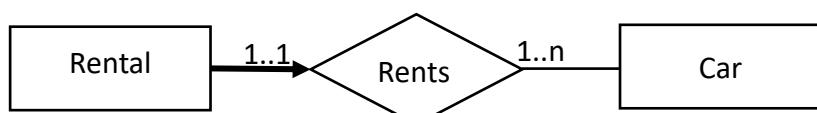
- Car (汽車) 實體和 Location (地點) 實體之間存在多對一的「位於」關係：
 - 一輛車只會停一個地點
 - 多輛車停放的地點可以是同一個地點



- Rental (租賃訂單) 實體和 Insurance (保險方案) 實體之間存在多對一的「包含」關係：
 - 每一筆租賃只能包含一個保險方案
 - 一個保險方案可以對應到多筆租賃訂單

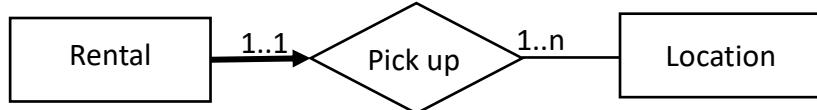


- Rental (租賃訂單) 實體和 Car(汽車) 實體之間存在多對一的「租用」關係：
 - 每筆租賃訂單只能租用一輛汽車
 - 一輛汽車可以被多筆租賃訂單租用



6. Rental (租賃訂單) 實體和 Location(地點) 實體之間存在多對一的「取車位置」關係：

- 每筆租賃訂單只能租用指定一個取車的地點
- 一個地點可以是多筆租賃訂單的取車地點



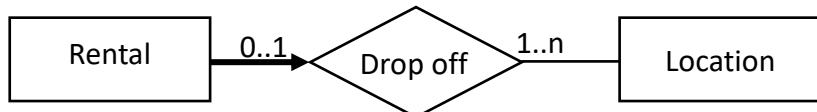
7. Rental (租賃訂單) 實體和 Location(地點) 實體之間存在多對一的「還車位置」關係：

- 每筆租賃訂單只能租用指定一個還車的地點
- 一個地點可以是多筆租賃訂單的還車地點



8. Rental (租賃訂單) 實體和 Location(地點) 實體之間存在多對一的「實際還車位置」關係：

- 每筆租賃訂單只能有一個/沒有實際還車的地點
- 一個地點可以是多筆租賃訂單的實際還車地點



5.3 ER Diagram – Entity in SQL

在本章節中會提及 record(資料)，對應之 DML 請參考第陸章節。

1. Member 實體

Member			
欄位名稱	資料型態	是否可為空	Domain
<u>google_id</u>	String	否	<p>Google OAuth 取得顧客識別碼，為 21~22 碼數字字串。 正規表達式: <code>^[0-9]{21,22}\$</code> 例:106455916943881880515 型態長度最多為 22</p>
gmail	String	否	<p>符合電子郵件格式: 帳號@網域。 使用者名稱(@前)長度需為 6~30 個字元，僅能包含: 英文字母、數字、句點(.)、底線(_)、百分比(%)、加號(+)、減號(-)。網域名稱(@後)需為有效網域格式， 例:gmail.com、yahoo.com.tw 等。 正規表達式: <code>^[a-zA-Z0-9._%+-]{6,30}@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}\$</code> 例:user_name@example.com 型態長度最多為 50</p>
phone	String	否	<p>符合行動電話號碼，共 10 位數字字串。 正規表達式: <code>^09\d{8}\\$</code>。 例: 0918202667 型態長度最多為 10</p>

DDL:

```
CREATE TABLE member(
    google_id VARCHAR(22) NOT NULL PRIMARY KEY,
    gmail VARCHAR(50) NOT NULL UNIQUE KEY,
    phone VARCHAR(10) NOT NULL UNIQUE KEY,
    CHECK (google_id REGEXP '^[0-9]{21,22}$'),
    CHECK (gmail REGEXP '^[a-zA-Z0-9._%+-]{6,30}@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'),
    CHECK (phone REGEXP '^09[0-9]{8}$')
);
```

Record (共 15 筆):

google_id	gmail	phone
879012345678901234567	quincy_liu@gmail.com	0978999000
657890123456789012345	kulpia@gmail.com	0978985123
111996782304424104647	gene_st@gmail.com	0914238500
192703829478239801723	Wen_Che@gmail.com	0932535148
215784230947120938765	asd_tw@gmail.com	0931164655
342057390258473869423	kevin_jj@gmail.com	0923969027
102345678901234567890	jane_liu@gmail.com	0999000111
213456789012345678901	kevin_liu@yahoo.com	0912333444
324567890123456789012	lisa_wang@nfu.edu.tw	0923444555
435678901234567890123	mike_smith@wistron.com	0934555666
546789012345678901234	nancy_johnson@foxconn.com	0945666777
567890123456789012345	ethan.lu@yahoo.com	0944555666
678901234567890123456	fiona.lee@nfu.edu.tw	0955666777
789012345678901234567	george.lin@gmail.com	0966777888
123456789012345678901	alice_wu@gmail.com	0987123456

2. Location 實體

Location			
欄位名稱	資料型態	是否可為空	Domain
<u>loc_id</u>	String	否	4 個大寫英文字組成。 正規表達式: <code>^[A-Z]{4}\$</code> 例: YUNH 型態長度最多為 4
loc_name	String	否	2~5 個中英文字 + 「店」字所組成 正規表達式: <code>^-[A-Za-z]{2,5}店\$</code> 例: 雲虎旗鑑店 型態長度最多為 6
city	String	否	必須為國內的縣市中文名。 正規表達式: <code>^(臺北市 新北市 桃園市 臺中市 臺南市 高雄市 基隆市 新竹市 嘉義市 新竹縣 苗栗縣 彰化縣 南投縣 雲林縣 嘉義縣 屏東縣 宜蘭縣 花蓮縣 臺東縣 澎湖縣 金門縣 連江縣)\$</code> 例: 雲林縣 型態長度最多為 3
district	String	否	1~3 字鄉鎮名+[鄉/鎮/市/區] 正規表達式: <code>^-[A-Za-z]{1,3}(鄉 鎮 市 區)\$</code> 例: 阿里山鄉、東區 型態長度最多為 4
address	String	否	符合國內街道規定。 <code>^([一-龜]{1,3}(村 里))?</code> <code>([0-9]{1,3}鄰)?</code> <code>[一-龜]+(路 街 大道)</code> <code>([一二三四五六七八九十]{1,3}段)?</code> <code>([0-9]+巷)?([0-9]+弄)?</code> <code>([0-9]{1,4}之)?([0-9]{1,4}號)</code> <code>([0-9]{1,3}樓(之[0-9]+)?)?</code> <code>([一-龜 A-Za-z0-9]+室)?\$</code> 例: 忠孝東路四段 200 號 11 樓

			型態長度最多為 30
--	--	--	------------

DDL:

```
CREATE TABLE location (
    loc_id VARCHAR(4) NOT NULL PRIMARY KEY CHECK (loc_id REGEXP '^[A-Z]{4}$'),
    loc_name VARCHAR(6) NOT NULL CHECK (loc_name REGEXP '^[-顎 A-Za-z]{2,5}店$'),
    city VARCHAR(3) NOT NULL CHECK (city REGEXP '^([臺北市|新北市|桃園市|臺中市|臺南市|高雄市|基隆市|新竹市|嘉義市|新竹縣|苗栗縣|彰化縣|南投縣|雲林縣|嘉義縣|屏東縣|宜蘭縣|花蓮縣|臺東縣|澎湖縣|金門縣|連江縣]$'),
    district VARCHAR(4) NOT NULL CHECK(district REGEXP '^[-顎]{1,3}(區|市|鎮|鄉)$'),
    address VARCHAR(50) NOT NULL,
    CHECK (address REGEXP '^([-顎]{1,3}(村|里))?(0-9){1,3}鄰)?[-顎]+(路|街|大道)([一二三四五五六七八九|十]{1,3}段)?(0-9+巷)?(0-9+弄)?(0-9{1,4}之)?0-9{1,4}號(0-9{1,3}樓(之0-9+)?)([-顎 A-Za-z0-9]+室)?$')
);
```

Record (共 15 筆):

loc_id	loc_name	city	district	address
YUNH	雲林縣虎尾店	雲林縣	虎尾鎮	文化路 64 號
YUNL	雲林縣斗六店	雲林縣	阿里山鄉	大學路三段 123 號
CHIE	嘉義市東區店	嘉義市	東區	學府路 300 號
CHIC	嘉義縣民雄店	嘉義縣	民雄鄉	神農路 145 號
YUNN	雲林縣斗南店	雲林縣	斗南鎮	大業路 106 號
TCWF	臺中市霧峰店	臺中市	霧峰區	吉峰東路 168 號
TCNT	臺中市南屯店	臺中市	南屯區	大墩路 200 號
TPXY	臺北市信義店	臺北市	信義區	松山路 100 號
NTPC	新北市板橋店	新北市	板橋區	文化路二段 50 號
KHLG	高雄市苓雅店	高雄市	苓雅區	和平一路 300 號
KHCZ	高雄市前鎮店	高雄市	前鎮區	中華五路 88 號
TNED	臺南市東區店	臺南市	東區	大學路 100 號
TNNB	臺南市北區店	臺南市	北區	成功路 200 號
HSIN	新竹市東區店	新竹市	東區	光復路一段 150 號
TYCL	桃園市中壢店	桃園市	中壢區	中正路 300 號

3. Insurance 實體

Insurance			
欄位名稱	資料型態	是否可為空	Domain
insurance_id	String	否	7 個大寫英文、數字組成 正規表達式: <code>^[A-Z0-9]{7}\$</code> 例:Y25S001 型態長度最多為 7
ins_name	String	否	3~15 個中英字或括號 正規表達式: <code>^[-龜A-Za-z\(\)\(\)]{3,15}\$</code> 例:第三人責任升級 型態長度最多為 15
coverage	String	否	1~250 個中文字或特定符號 正規表達式: <code>^[-龜A-Za-z0-9\(\),。、()[:「」]]{1,250}\$</code> 型態長度最多為 250
ins_fee	Integer	否	正整數($>=0$)

DDL:

```
CREATE TABLE insurance(
    insurance_id VARCHAR(7) NOT NULL PRIMARY KEY CHECK (insurance_id REGEXP
'^[A-Z0-9]{7}$'),
    ins_name VARCHAR(20) NOT NULL UNIQUE KEY CHECK (ins_name REGEXP '^[一
-龥 A-Za-z\\(\\) ( ) ]{3,15}$'),
    coverage VARCHAR(250) NOT NULL CHECK (coverage REGEXP '^-龥 A-Za-z0-
9\\(\\) , 。、( )::「_」 ]{1,250}$'),
    ins_fee INT NOT NULL CHECK (ins_fee >= 0)
);
```

Record (共 10 筆):

insurance_id	ins_name	car_id	ins_fee
Y25S001	基本保險方案	第三人責任保險：每人傷害上限 200 萬、每事故傷害上限 400 萬、財損上限 50 萬。駕駛人保險 100 萬、乘客每人 100 萬（超載除外）。	0
Y25S002	第三人責任升級	每人傷害上限提升至 500 萬、每事故傷害 1000 萬、財損上限 200 萬。其餘條件比照基本方案。	200
Y25S003	第三人責任與車體損害升級版	每人傷害上限提升至 500 萬、每事故傷害 1000 萬、財損上限 200 萬。其餘規定比照基本保險。	400
Y25S004	豪華第三人責任險	每人傷害上限 800 萬、每事故傷害 1500 萬、財損上限 300 萬，乘客保險每人 200 萬。不含車體損害。	600
Y25X001	全方位保險方案	涵蓋第三人責任升級版保障，並新增車體損失險（含碰撞、自撞、翻車、他人肇逃）、竊盜險。自負額每次事故 5000 元。	800
Y25X002	高階全險(免自負額)	等同全方位保險方案，另提供不限事故次數之免自負額保障。亦納入每日車輛停駛補償保險。	1400
Y25E001	節能保險方案	提供基本保險保障，專為電動車設計，另含電池損害與火災保險（限電動車適用）。	100
Y25X003	超級全險(海外人士專用)	提供全方位保險保障，另納入外籍駕駛法律援助保險與語言協助。	1600
Y25Z001	商務菁英保險方案	含高階全險保障，另增加行程延誤補償與駕駛更換服務，每日最高補償 3000 元。	1800
Y25Z002	至尊全險方案	含所有前述保險保障，另提供不限次數道路救援、全天候代步車服務、全額車損理賠（免自負額）。	2500

4. Model 實體

Model			
欄位名稱	資料型態	是否可為空	Domain
<u>model_id</u>	String	否	7位大寫英文、數字組成 正規表達式: <code>^[A-Z0-9]{7}\$</code> 例: MITGD01 型態長度最多為 7
brand	String	否	1~20 個大小寫英文組成 正規表達式: <code>^[A-Za-z]{1,20}\$</code> 例: Volkswagen 型態長度最多為 20
model_name	String	否	1~20 個大小寫英文、數字、空白組成 正規表達式: <code>^[A-Za-z0-9 -]{1,20}\$</code> 例: Grand Lancer 型態長度最多為 20
car_type	String	否	限定字串: Compact、Sedan、SUV、MPV 型態長度最多為 7
fuel_type	String	否	限定字串: Gasoline、Electric、Hybrid 型態長度最多為 8
engine_cc	Integer	否	非負整數(≥ 0)
transmission	Integer	否	必須為 0 或 1
image_url	String	否	1~50 個英文字、數字、底線組成，且 img/ 開頭、.jpg 或.png 結尾 正規表達式: <code>^img\[A-Za-z0-9 _]+\.\jpg\$</code> 例: img/CMC_JSPACE.jpg 型態長度最多為 50

DDL:

```
CREATE TABLE model(
    model_id VARCHAR(7) NOT NULL PRIMARY KEY CHECK (model_id REGEXP '^[A-Z0-9]{7}$'),
    brand VARCHAR(20) NOT NULL CHECK (brand REGEXP '^[A-Za-z]{1,20}$'),
    model_name VARCHAR(20) NOT NULL,
    car_type VARCHAR(7) NOT NULL CHECK (car_type IN ('Compact', 'Sedan', 'SUV', 'MPV')),
    fuel_type VARCHAR(8) NOT NULL CHECK (fuel_type IN ('Gasoline', 'Electric', 'Hybrid')),
    engine_cc INT NOT NULL CHECK (engine_cc >= 0),
    transmission INT NOT NULL CHECK (transmission IN (0, 1)),
    image_url VARCHAR(50) NOT NULL CHECK (image_url REGEXP '^img\\/[A-Za-z0-9_]+\\.\\(jpg|png\\)$')
);
```

Record (共 10 筆):

model_id	brand	model_name	car_type	fuel_type	engine_cc	transmission	image_url
TOTA012	TOYOTA	Altis 12	Compact	Hybrid	1800	1	img/Altis12.jpg
HONO019	HONDA	Odyssey 2019	MPV	Gasoline	2500	1	img/Odyssey12.jpg
TES3001	TESLA	Model 3	Sedan	Electric	0	1	img/model3.jpg
TOTR022	TOYOTA	RAV4 2022	SUV	Hybrid	2500	1	img/rav4_2022.jpg
MITG001	MITSUBISHI	Grand Lancer	Compact	Gasoline	1500	1	img/glancer.jpg
FODF018	FORD	Focus 2018	Compact	Gasoline	1600	1	img/focus_2018.jpg
HYDT020	HYUNDAI	Tucson 2020	SUV	Gasoline	2000	1	img/tucson_2020.jpg
NISL021	NISSAN	Leaf 2021	Compact	Electric	0	1	img/leaf_2021.jpg
KIAS022	KIA	Seltos 2022	SUV	Gasoline	1600	1	img/seltos_2022.jpg
CMCJ001	CMC	JSPACE	MPV	Gasoline	1500	0	img/CMC_JSPACE.jpg

5. Car 實體

Car			
欄位名稱	資料型態	是否可為空	Domain
vin	String	否	<p>符合 VIN 編碼標準 ISO 4030(可兼容 ISO3779): WMI 全球製造商代碼(第 1~3 碼)+VDS 車輛特徵碼(4~8 碼)+核對號碼(第 9 碼)+生產年份 (第 10 碼)+生產代號(第 11 碼)+VIS 車輛識別碼(12~17 碼)。共 17 碼字串。</p> <p>正規表示式:</p> <pre>^[A-HJ-NPR-Za-hj-npr-z0-9]{8}[0-9X][A-HJ-NPR-Za-hj-npr-z0-9]{2}[0-9]{6}\$</pre> <p>型態長度最多為 17</p>
plate_number	String	否	<p>符合國內租賃車牌規定</p> <p>正規表示式:</p> <pre>^R[A-Z]{2}-[0-35-9]{4}\$</pre> <p>型態長度最多為 8</p>
daily_fee	Integer	否	大於 0 正整數(>0)
late_fee	Integer	否	大於 0 正整數(>0)
year_made	Integer	否	大於等於 1980 正整數 (≥ 1980)
seat_num	Integer	否	大於 0 且小於 16 正整數(>0 且 < 16)
color	String	否	<p>1~4 中文字+1「色」字</p> <p>正規表示式:</p> <pre>^-[纈]{1,4}色\$</pre> <p>型態長度最多為 5</p>
mileage	INT(6)	否	大於 0 正整數(>0)
car_status	String	否	<p>限定字串: available、maintenance、disable</p> <p>型態長度最多為 12</p>
notes	String	是	任意字串

DDL:

```
CREATE TABLE car (
    vin VARCHAR(17) NOT NULL PRIMARY KEY CHECK (vin REGEXP '^[A-HJ-NPR-Za-hj-npr-z0-9]{8}[0-9X][A-HJ-NPR-Za-hj-npr-z0-9]{2}[0-9]{6}$'),
    plate_number VARCHAR(8) NOT NULL UNIQUE KEY CHECK (plate_number REGEXP '^R[A-Z]{2}-[0-35-9]{4}$'),
    daily_fee INT NOT NULL CHECK (daily_fee > 0),
    late_fee INT NOT NULL CHECK (late_fee > 0),
    year_made INT NOT NULL CHECK (year_made >= 1980),
    seat_num INT NOT NULL CHECK (seat_num > 0 and seat_num < 16),
    color VARCHAR(5) NOT NULL CHECK (color REGEXP '^[一-龥]{1,4}色$'),
    mileage INT NOT NULL CHECK (mileage > 0),
    car_status VARCHAR(12) NOT NULL DEFAULT 'available' CHECK (car_status IN ('available', 'maintenance', 'disable')),
    notes TEXT
);
```

Record (共 12 筆):

第 1~6 筆:

vin		plate_number	daily_fee	late_fee	year_made
4T1BE46KX7U123456		RAA-7777	1800	200	2020
seat_num	color	mileage	notes		car_status
5	白色	15000	可以隨心所欲		available
5	白色	16500	可以隨心所欲		available
7	黑色	12000	座椅舒適，適合家庭出遊		available
5	銀色	8000	行駛安靜		available
5	白色	7500	可以隨心所欲 SUV		available
5	白色	9200	可以隨心所欲 SUV		maintenance

第 7~12 筆:

vin	plate_number	daily_fee	late_fee	year_made
JMBGR31L6JU123456	RAF-5678	1700	200	2019
1FADP3F2XJL123456	RBC-1239	1800	200	2018
KM8K6CA43LU123456	RBD-8567	2000	250	2020
1N4A11APXJC123899	RBB-6789	1800	200	2021
1N4A11APXJC123456	RBD-1112	1800	200	2021
KNDJ23AU7L1216211	RBZ-2355	1800	200	2022
seat_num	color	mileage	notes	car_status
5	黑色	15000	內裝整潔，冷氣超強	available
5	銀色	18000	油耗低	maintenance
5	黑色	5000	空間寬敞	available
5	綠色	3000	行駛安靜	available
5	綠色	3000	行駛安靜	disabled
5	藍色	4000	小巧靈活，適合城市駕駛	disabled

6. Rental 實體

Rental			
欄位名稱	資料型態	是否可為空	Domain
<u>google_id</u>	Integer	否	Google OAuth 取得使用者識別碼，為 21~22 碼數字字串。 正規表達式: ^[0-9]{21,22}\$ 例:106455916943881880515 型態長度最多為 22
<u>created_at</u>	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
actual_return	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒 NULL 表示尚未歸還
rental_status	String	否	限定字串: pending、active、completed、cancelled 型態長度最多為 9
amount	Integer	否	正整數(>0)
payment_date	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
method	String	否	限定字串：cash、credit、linepay 型態長度最多為 8

DDL:

```
CREATE TABLE rental (
    google_id VARCHAR(22) NOT NULL CHECK (google_id REGEXP '^[0-9]{21,22}$'),
    created_at DATETIME NOT NULL,
    actual_return DATETIME,
    rental_status VARCHAR(9) NOT NULL CHECK (rental_status IN ('pending', 'active',
    'completed', 'cancelled', 'reject')),
```

```

amount INT NOT NULL CHECK (amount >= 0),
payment_date DATETIME,
method VARCHAR(8) NOT NULL CHECK (method IN ('cash', 'credit', 'linepay')),
PRIMARY KEY (google_id,created_at)
);

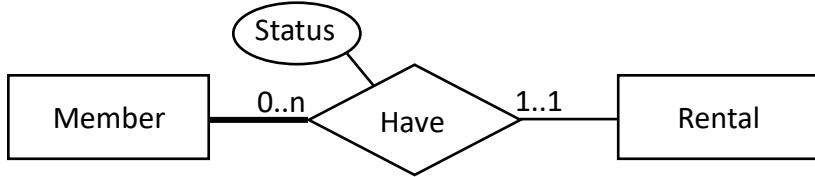
```

Record (共 11 筆):

google_id	created_at	actual_return	amount	payment_date	method
879012345678901234567	2025-06-01 13:09:21	NULL	8000	NULL	cash
879012345678901234567	2025-05-25 13:25:08	NULL	6400	NULL	cash
657890123456789012345	2025-05-01 13:39:19	NULL	12400	NULL	cash
192703829478239801723	2025-06-14 18:59:20	2025-06-20 00:30:00	5200	2025-06-15 19:25:00	linepay
111996782304424104647	2025-07-08 19:22:26	NULL	4400	2025-07-08 21:52:34	credit
102345678901234567890	2025-06-15 11:44:58	2025-06-27 17:55:00	10000	2025-06-23 09:20:00	cash
213456789012345678901	2025-06-01 19:45:39	NULL	5200	NULL	cash
324567890123456789012	2025-06-01 12:22:28	NULL	7800	NULL	cash
435678901234567890123	2025-06-03 09:25:35	2025-06-10 19:08:00	10800	2025-06-07 19:45:30	cash
213456789012345678901	2025-06-12 08:11:43	NULL	3800	NULL	cash
213456789012345678901	2025-06-02 13:13:33	NULL	18800	NULL	cash

5.4 ER Diagram – Relationship in SQL

1. Member ↔ Rental (會員預約租車訂單)



DDL:

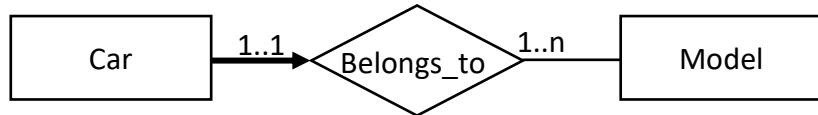
```

CREATE TABLE Have (
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    status VARCHAR(9) NOT NULL CHECK (status IN
        ('pending', 'active', 'completed', 'cancelled', 'reject')),
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id, created_at)
        ON DELETE CASCADE,
    FOREIGN KEY (google_id) REFERENCES Member (google_id)
        ON DELETE NO ACTION,
    PRIMARY KEY(google_id,created_at)
);
    
```

Record (共 11 筆):

google_id	created_at	Status
879012345678901234567	2025-06-01 13:09:21	reject
879012345678901234567	2025-05-25 13:25:08	cancelled
657890123456789012345	2025-05-01 13:39:19	active
192703829478239801723	2025-06-14 18:59:20	completed
111996782304424104647	2025-07-08 19:22:26	active
102345678901234567890	2025-06-15 11:44:58	completed
213456789012345678901	2025-06-01 19:45:39	cancelled
324567890123456789012	2025-06-01 12:22:28	cancelled
435678901234567890123	2025-06-03 09:25:35	completed
213456789012345678901	2025-06-12 08:11:43	active
213456789012345678901	2025-06-02 13:13:33	active

2. Car ↔ Model (車輛所屬的車型)



DDL:

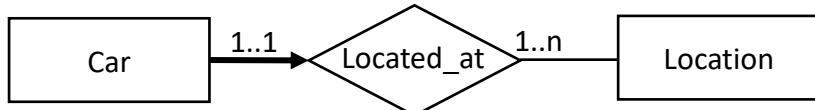
```

CREATE TABLE Belongs_to (
    CarVin VARCHAR(17) NOT NULL, -- role
    ModelId VARCHAR(7) NOT NULL, -- role
    PRIMARY KEY (CarVin),
    FOREIGN KEY (CarVin) REFERENCES Car(vin)
        ON DELETE NO ACTION,
    FOREIGN KEY (ModelId) REFERENCES Model(model_id)
        ON DELETE NO ACTION
);
  
```

Record (共 15 筆):

CarVin	ModelId
4T1BE46KX7U123456	TOTA012
4T1BE46KXPX123789	TOTA012
5FNRL38647B654321	HONO019
5YJ3E1EA7KF123456	TES3001
4T3HV36KXPX456781	TOTR022
4T3HV36KXPX456782	TOTR022
JMBGR31L6JU123456	MITG001
1FADP3F2XJL123456	FODF018
KM8K6CA43LU123456	HYDT020
1N4A11APXJC123899	NISL021
1N4A11APXJC123456	NISL021
KNDJ23AU7L1216211	KIAS022
KNDJ23AU7L1234567	KIAS022
CMBJ1234567890123	CMCJ001
CMBJ1234567890124	CMCJ001

3. Car ↔ Location (汽車目前停放地點)



DDL:

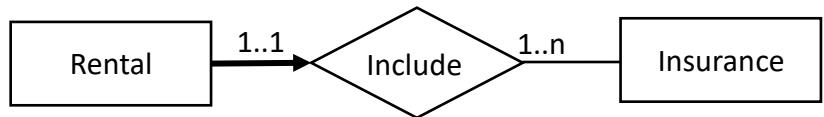
```

CREATE TABLE Located_at (
    CarVin VARCHAR(17) NOT NULL,
    LocationId VARCHAR(4),
    PRIMARY KEY (CarVin),
    FOREIGN KEY (CarVin) REFERENCES Car(vin)
        ON DELETE NO ACTION,
    FOREIGN KEY (LocationId) REFERENCES Location(loc_id)
        ON DELETE SET NULL
);
  
```

Record (共 15 筆):

CarVin	LocationId
4T1BE46KX7U123456	YUNH
4T1BE46KXPX123789	YUNL
5FNRL38647B654321	YUNH
5YJ3E1EA7KF123456	KHLG
4T3HV36KXPX456781	YUNH
4T3HV36KXPX456782	CHIE
JMBGR31L6JU123456	YUNH
1FADP3F2XJL123456	TPXY
KM8K6CA43LU123456	TCWF
1N4A11APXJC123899	TCNT
1N4A11APXJC123456	TCNT
KNDJ23AU7L1216211	CHIC
KNDJ23AU7L1234567	TPXY
CMBJ1234567890123	TCWF
CMBJ1234567890124	YUNN

4. Rental ↔ Insurance (訂單選用保險)



DDL:

```

CREATE TABLE Include (
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    InsuranceId VARCHAR(7) NOT NULL,
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id, created_at)
        ON DELETE CASCADE,
    FOREIGN KEY (InsuranceId) REFERENCES Insurance (insurance_id)
        ON DELETE NO ACTION,
    PRIMARY KEY(google_id, created_at)
);
  
```

Record (共 11 筆):

google_id	created_at	InsuranceId
879012345678901234567	2025-06-01 13:09:21	Y25S002
879012345678901234567	2025-05-25 13:25:08	Y25X002
657890123456789012345	2025-05-01 13:39:19	Y25X002
192703829478239801723	2025-06-14 18:59:20	Y25E001
111996782304424104647	2025-07-08 19:22:26	Y25S003
102345678901234567890	2025-06-15 11:44:58	Y25S001
213456789012345678901	2025-06-01 19:45:39	Y25S004
324567890123456789012	2025-06-01 12:22:28	Y25S003
435678901234567890123	2025-06-03 09:25:35	Y25Z001
213456789012345678901	2025-06-12 08:11:43	Y25X003
213456789012345678901	2025-06-02 13:13:33	Y25Z002

5. Rental ↔ Car (訂單選用車輛)



DDL:

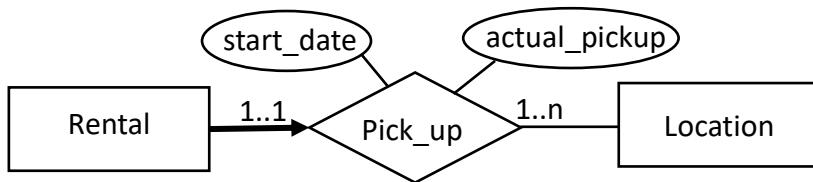
```

CREATE TABLE Rents (
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    CarVin VARCHAR(17) NOT NULL,
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id, created_at)
        ON DELETE CASCADE,
    FOREIGN KEY (CarVin) REFERENCES Car (vin)
        ON DELETE NO ACTION,
    PRIMARY KEY(google_id, created_at)
);
  
```

Record (共 11 筆):

google_id	created_at	CarVin
879012345678901234567	2025-06-01 13:09:21	879012345678901234567
879012345678901234567	2025-05-25 13:25:08	879012345678901234567
657890123456789012345	2025-05-01 13:39:19	657890123456789012345
192703829478239801723	2025-06-14 18:59:20	192703829478239801723
111996782304424104647	2025-07-08 19:22:26	111996782304424104647
102345678901234567890	2025-06-15 11:44:58	102345678901234567890
213456789012345678901	2025-06-01 19:45:39	213456789012345678901
324567890123456789012	2025-06-01 12:22:28	324567890123456789012
435678901234567890123	2025-06-03 09:25:35	435678901234567890123
213456789012345678901	2025-06-12 08:11:43	213456789012345678901
213456789012345678901	2025-06-02 13:13:33	213456789012345678901

6. Rental ↔ Location(租用時間和地點、實際取車時間)



DDL:

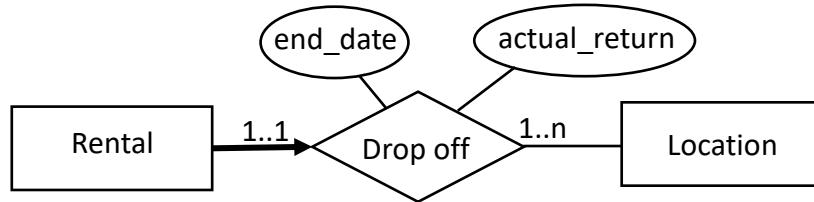
```

CREATE TABLE Pick_up(
    start_date DATETIME,          -- attribute
    actual_pickup DATETIME,       -- attribute
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    LocationId VARCHAR(4) NOT NULL,
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id, created_at)
        ON DELETE CASCADE,
    FOREIGN KEY (LocationId) REFERENCES Location (loc_id)
        ON DELETE NO ACTION,
    PRIMARY KEY(google_id,created_at)
);
    
```

Record (共 11 筆):

google_id	created_at	LocationId	start_date	actual_pickup
879012345678901234567	2025-06-01 13:09:21	YUNH	2025-06-02 08:00:00	NULL
879012345678901234567	2025-05-25 13:25:08	YUNH	2025-06-13 08:00:00	NULL
657890123456789012345	2025-05-01 13:39:19	YUNN	2025-07-14 11:05:00	NULL
192703829478239801723	2025-06-14 18:59:20	KHLG	2025-06-18 13:10:00	2025-06-18 13:05:20
111996782304424104647	2025-07-08 19:22:26	TCNT	2025-07-11 21:30:00	NULL
102345678901234567890	2025-06-15 11:44:58	TCWF	2025-06-23 09:30:00	2025-06-23 09:32:00
213456789012345678901	2025-06-01 19:45:39	TCWF	2025-07-12 07:30:00	NULL
324567890123456789012	2025-06-01 12:22:28	KHLG	2025-06-02 06:03:00	NULL
435678901234567890123	2025-06-03 09:25:35	YUNH	2025-06-07 20:00:00	2025-06-07 20:00:00
213456789012345678901	2025-06-12 08:11:43	YUNH	2025-06-19 05:00:00	NULL
213456789012345678901	2025-06-02 13:13:33	YUNH	2025-07-15 07:03:00	NULL

7. Rental ↔ Location(歸還時間和地點、實際歸還時間)



DDL:

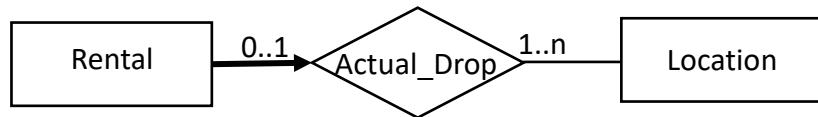
```

CREATE TABLE Drop_off (
    end_date DATETIME,          -- attribute
    actual_return DATETIME,      -- attribute
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    LocationId VARCHAR(4) NOT NULL,
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id,
    created_at)
    ON DELETE CASCADE,
    FOREIGN KEY (LocationId) REFERENCES Location (loc_id)
    ON DELETE NO ACTION,
    PRIMARY KEY(google_id,created_at)
);
  
```

Record (共 11 筆):

google_id	created_at	LocationId	EndTime	actual_return
879012345678901234567	2025-06-01 13:09:21	YUNH	2025-06-05 17:00:00	NULL
879012345678901234567	2025-05-25 13:25:08	YUNH	2025-06-14 17:00:00	NULL
657890123456789012345	2025-05-01 13:39:19	YUNN	2025-07-17 23:45:00	NULL
192703829478239801723	2025-06-14 18:59:20	KHLG	2025-06-20 00:40:00	2025-06-20 00:30:00
111996782304424104647	2025-07-08 19:22:26	TCNT	2025-07-13 21:00:00	NULL
102345678901234567890	2025-06-15 11:44:58	CHIC	2025-06-27 18:00:00	2025-06-27 17:55:00
213456789012345678901	2025-06-01 19:45:39	TCWF	2025-07-13 20:00:00	NULL
324567890123456789012	2025-06-01 12:22:28	YUNH	2025-06-04 19:05:00	NULL
435678901234567890123	2025-06-03 09:25:35	YUNL	2025-06-10 19:05:00	2025-06-10 19:08:00
213456789012345678901	2025-06-12 08:11:43	YUNH	2025-06-19 23:00:00	NULL
213456789012345678901	2025-06-02 13:13:33	TCNT	2025-07-18 16:06:00	NULL

8. Rental ↔ Location (實際歸還地點)



DDL:

```
CREATE TABLE Actual_Drop(
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    LocationId VARCHAR(4) NOT NULL,
    FOREIGN KEY (google_id, created_at) REFERENCES Rental (google_id,
    created_at)
    ON DELETE CASCADE,
    FOREIGN KEY (LocationId) REFERENCES Location (loc_id)
    ON DELETE NO ACTION,
    PRIMARY KEY(google_id,created_at)
);
```

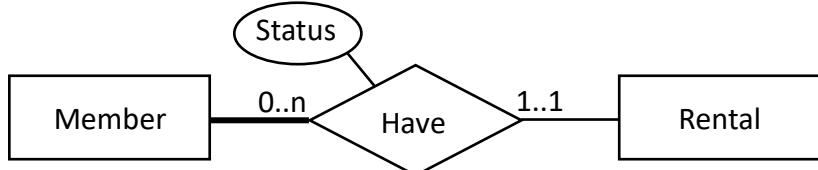
Record (共 3 筆):

google_id	created_at	LocationId
192703829478239801723	2025-06-14 18:59:20	KHLG
102345678901234567890	2025-06-15 11:44:58	CHIC
435678901234567890123	2025-06-03 09:25:35	YUNL

5.5 ER Diagram – 合併 Table

針對於 Rental Table 與多個 relation 有 1..1 的關係，因此合併 Rental Table。其中第六項之 Rental & <Return>，為訂單實際歸還之地點，之間關係為 0..1，為了減少表數，故一起合併，Record 若未關聯到實際還車地點則用 Null 表示，在現實生活中符合邏輯，訂單未還沒到歸還階段，實際歸還地點會是空值。

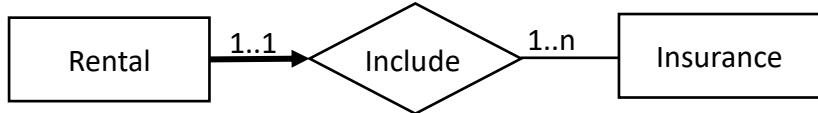
1. Rental & <Have>



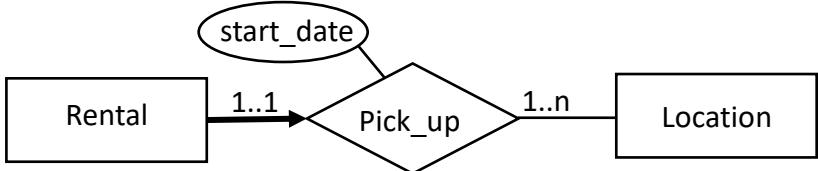
2. Rental & <Rents>



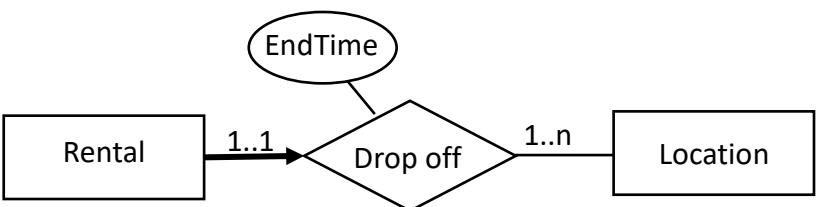
3. Rental & <Include>



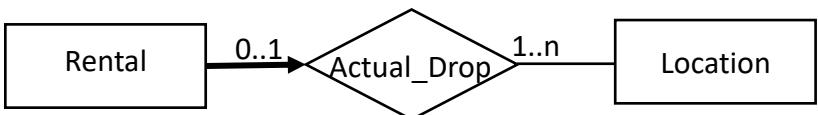
4. Rental & <Pick_up>



5. Rental & <Drop_off>



6. Rental & <Actual_Drop>



合併後 schema:

欄位名稱	資料型態	是否可為空	Domain
google_id	Integer	否	參照 Member Table 中的 google_id
created_at	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
start_date	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
end_date	DATETIME	否	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
actual_pickup	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒 NULL 表示尚未取車
actual_return	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒 NULL 表示尚未歸還
rental_status	String	否	限定字串: pending、active、completed、cancelled 型態長度最多為 9
amount	Integer	否	正整數(>0)
payment_date	DATETIME	是	YYYY/MM/DD HH:MM:SS 西元年/月份/日期 小時(24 小時制):分:秒
method	String	否	限定字串：cash、credit、linepay 型態長度最多為 8
car_vin	String	否	參照 Car Table 中的 car_vin

pickup_loc	String	否	參照 Location Table 中的 loc_id
drop_loc	String	否	參照 Location Table 中的 loc_id
actual_drop_loc	String	是	參照 Location Table 中的 loc_id NULL 表示尚未歸還
insurance_id	String	否	參照 Insurance Table 中的 insurance_id

合併後的 DDL:

```

CREATE TABLE rental (
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    start_date DATETIME NOT NULL,
    end_date DATETIME NOT NULL,
    actual_pickup DATETIME,
    actual_return DATETIME,
    rental_status VARCHAR(9) NOT NULL CHECK (rental_status IN ('pending',
    'active', 'completed', 'cancelled', 'reject')),
    amount INT NOT NULL CHECK (amount >= 0),
    payment_date DATETIME,
    method VARCHAR(8) NOT NULL CHECK (method IN ('cash', 'credit', 'linepay')),
    car_vin VARCHAR(17) NOT NULL,
    pickup_loc VARCHAR(4) NOT NULL,
    drop_loc VARCHAR(4) NOT NULL,
    actual_drop_loc VARCHAR(4),
    insurance_id VARCHAR(7) NOT NULL,
    CONSTRAINT fk_google_id FOREIGN KEY (google_id) REFERENCES
    member(google_id)
        ON DELETE CASCADE,
    CONSTRAINT fk_car_vin FOREIGN KEY (car_vin) REFERENCES car(vin)
        ON DELETE NO ACTION,
    CONSTRAINT fk_pickup_loc FOREIGN KEY (pickup_loc) REFERENCES
    location(loc_id) ON DELETE NO ACTION,
    CONSTRAINT fk_drop_loc FOREIGN KEY (drop_loc) REFERENCES location(loc_id)
        ON DELETE NO ACTION,
    CONSTRAINT fk_actual_drop_loc FOREIGN KEY (actual_drop_loc) REFERENCES
    location(loc_id) ON DELETE SET NULL,
    CONSTRAINT fk_insurance_id FOREIGN KEY (insurance_id) REFERENCES
    insurance(insurance_id) ON DELETE NO ACTION,
    PRIMARY KEY (google_id,created_at)
);

```

合併後 Record(共 11 筆):

第 1~9 筆:

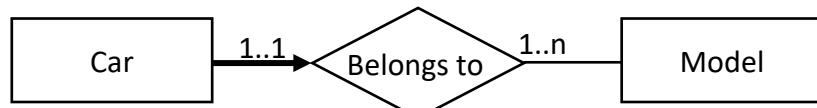
google_id	created_at	start_date	end_date	actual_pickup	actual_return	rental_status	
87901234567 8901234567	2025-06-01 13:09:21	2025-06-02 08:00:00	2025-06-05 17:00:00	NULL	NULL	reject	
87901234567 8901234567	2025-05-25 13:25:08	2025-06-13 08:00:00	2025-06-14 17:00:00	NULL	NULL	cancelled	
65789012345 6789012345	2025-05-01 13:39:19	2025-07-14 11:05:00	2025-07-17 23:45:00	NULL	NULL	active	
19270382947 8239801723	2025-06-14 18:59:20	2025-06-18 13:10:00	2025-06-20 00:40:00	2025-06-18 13:05:20	2025-06-20 00:30:00	completed	
11199678230 4424104647	2025-07-08 19:22:26	2025-07-11 21:30:00	2025-07-13 21:00:00	NULL	NULL	active	
10234567890 1234567890	2025-06-15 11:44:58	2025-06-23 09:30:00	2025-06-27 18:00:00	2025-06-23 09:32:00	2025-06-27 17:55:00	completed	
21345678901 2345678901	2025-06-01 19:45:39	2025-07-12 07:30:00	2025-07-13 20:00:00	NULL	NULL	cancelled	
32456789012 3456789012	2025-06-01 12:22:28	2025-06-02 06:03:00	2025-06-04 19:05:00	NULL	NULL	cancelled	
amount	payment_date	method	car_vin	pickup_loc	drop_loc	actual_drop_loc	insurance_id
8000	NULL	cash	4T1BE46KX 7U123456	YUNH	YUNH	NULL	Y25S002
6400	NULL	cash	4T1BE46KX 7U123456	YUNH	YUNH	NULL	Y25X002
12400	NULL	cash	CMBJ12345 67890124	YUNN	YUNN	NULL	Y25X002
5200	2025-06-15 19:25:00	linepay	5YJ3E1EA7K F123456	KHLG	KHLG	KHLG	Y25E001
4400	2025-07-08 21:52:34	credit	1N4A11APX JC123899	TCNT	TCNT	NULL	Y25S003
10000	2025-06-23 09:20:00	cash	KM8K6CA4 3LU123456	TCWF	CHIC	CHIC	Y25S001
5200	NULL	cash	KM8K6CA4 3LU123456	TCWF	TCWF	NULL	Y25S004
7800	NULL	cash	4T3HV36KX PX456781	KHLG	YUNH	NULL	Y25S003

第 9~11 筆：

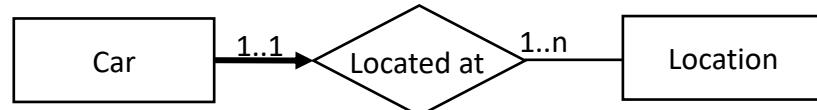
google_id	created_at	start_date	end_date	actual_pickup	actual_return	rental_status
43567890123	2025-06-03 09:25:35	2025-06-07 20:00:00	2025-06-10 19:05:00	2025-06-07 20:00:00	2025-06-10 19:08:00	completed
21345678901	2025-06-12 08:11:43	2025-06-19 05:00:00	2025-06-19 23:00:00	NULL	NULL	
21345678901	2025-06-02 13:13:33	2025-07-15 07:03:00	2025-07-18 16:06:00	NULL	NULL	active
amount	payment_date	method	car_vin	pickup_loc	drop_loc	actual_drop_loc
10800	2025-06-07 19:45:30	cash	4T1BE46KX 7U123456	YUNH	YUNL	YUNL
3800	NULL	cash	5FNRL3864 7B654321	YUNH	YUNH	NULL
18800	NULL	cash	5FNRL3864 7B654321	YUNH	TCNT	NULL
						insurance_id

針對於 Car Table 與 <Belongs_to> 有 1..1 的關係以及與 <Located at> 有 1..1 的關係，因此將 <Belongs_to> 和 <Located at> 合併到 Car Table

- Car & <Belongs to>



- Car & <Located at>



合併後的 DDL:

```

CREATE TABLE car (
    vin VARCHAR(17) NOT NULL PRIMARY KEY CHECK (vin REGEXP '^[A-HJ-NPR-Za-hj-npr-z0-9]{8}[0-9X][A-HJ-NPR-Za-hj-npr-z0-9]{2}[0-9]{6}$'),
    plate_number VARCHAR(8) NOT NULL UNIQUE KEY CHECK (plate_number
    REGEXP '^R[A-Z]{2}-[0-35-9]{4}$'),
    daily_fee INT NOT NULL CHECK (daily_fee > 0),
    late_fee INT NOT NULL CHECK (late_fee > 0),
    year_made INT NOT NULL CHECK (year_made >= 1980),
    seat_num INT NOT NULL CHECK (seat_num > 0 and seat_num < 16),
)
    
```

```

color VARCHAR(5) NOT NULL CHECK (color REGEXP '^[\u2e80-\u9fff]{1,4}色$'),
mileage INT NOT NULL CHECK (mileage > 0),
car_status VARCHAR(12) NOT NULL DEFAULT 'available' CHECK (car_status IN
('available', 'maintenance', 'disable')),
notes TEXT,
model_id VARCHAR(7) NOT NULL,
loc_id VARCHAR(4),
FOREIGN KEY (model_id) REFERENCES model(model_id)
    ON DELETE NO ACTION,
FOREIGN KEY (loc_id) REFERENCES location(loc_id)
    ON DELETE SET NULL
);

```

合併後 Record(共 15 筆):

第 1~10 筆:

vin	plate_number	model_id	daily_fee	late_fee	year_made
4T1BE46KX7U123456	RAA-7777	TOTA012	1800	200	2020
4T1BE46KXPX123789	RAA-8888	TOTA012	1800	200	2020
5FNRL38647B654321	RAR-2385	HONO019	2200	200	2020
5YJ3E1EA7KF123456	REK-3556	TES3001	2500	300	2021
4T3HV36KXPX456781	RBC-8811	TOTR022	2200	300	2022
4T3HV36KXPX456782	RBC-9922	TOTR022	2200	300	2022
JMBGR31L6JU123456	RAF-5678	MITG001	1700	200	2019
1FADP3F2XJL123456	RBC-1239	FODF018	1800	200	2018
KM8K6CA43LU123456	RBD-8567	HYDT020	2000	250	2020
1N4A11APXJC123899	RBB-6789	NISL021	1800	200	2021

loc_id	seat_num	color	mileage	notes	car_status
YUNH	5	白色	15000	可以隨心所欲	available
YUNL	5	白色	16500	可以隨心所欲	available
YUNH	7	黑色	12000	座椅舒適，適合家庭出遊	available
KHLG	5	銀色	8000	行駛安靜	available
YUNH	5	白色	7500	可以隨心所欲 SUV	available
CHIE	5	白色	9200	可以隨心所欲 SUV	maintenance
YUNH	5	黑色	15000	內裝整潔，冷氣超強	available
TPXY	5	銀色	18000	油耗低	maintenance
TCWF	5	黑色	5000	空間寬敞	available
TCNT	5	綠色	3000	行駛安靜	available

第 11~15 筆：

vin	plate_number	model_id	daily_fee	late_fee	year_made
1N4A11APXJC123456	RBD-1112	NISL021	1800	200	2021
KNDJ23AU7L1216211	RBZ-2355	KIAS022	1800	200	2022
KNDJ23AU7L1234567	RBS-8875	KIAS022	1800	200	2022
CMBJ1234567890123	RAS-1235	CMCJ001	1700	200	2019
CMBJ1234567890124	RAS-5678	CMCJ001	1700	200	2019
loc_id	seat_num	color	mileage	notes	car_status
TCNT	5	綠色	3000	行駛安靜	disable
CHIC	5	藍色	4000	小巧靈活，適合城市駕駛	disable
TPXY	5	藍色	4000	小巧靈活，適合城市駕駛	disable
TCWF	8	白色	8000	空間大，適合家庭出遊	available
YUNN	8	白色	5500	空間大，適合家庭出遊	available

陸、VIEW / GRANT

本系統區分三種不同的使用者群組，分別為：

- guest_user :為非會員(未註冊)之使用者
- customer_user :為會員(已註冊)之使用者
- staff_admin :為租車公司員工(管理員)

本章節將依序說明各使用者群組的 VIEW 及操作權限(GRANT)，以劃分不同角色在系統中的權限範圍，VIEW 導向會是以提供後端程式開發使用。

1. guest_user

1.所有 google_id 的 VIEW。用於註冊時確認 Google 帳號是否已註冊。

```
CREATE VIEW GU_member_google_view AS
SELECT google_id FROM member;
GRANT SELECT ON car_rental_db.GU_member_google_view TO 'guest_user'@'localhost';
```

SELECT VIEW 結果：

```
MariaDB [car_rental_db]> SELECT * FROM GU_member_google_view;
+-----+
| google_id |
+-----+
| 213456789012345678901 |
| 111996782304424104647 |
| 324567890123456789012 |
| 342057390258473869423 |
| 215784230947120938765 |
| 192703829478239801723 |
| 435678901234567890123 |
| 567890123456789012345 |
| 546789012345678901234 |
| 678901234567890123456 |
| 789012345678901234567 |
| 657890123456789012345 |
| 879012345678901234567 |
| 123456789012345678901 |
| 102345678901234567890 |
+-----+
15 rows in set (0.002 sec)
```

2.查詢所有可租用車輛的 VIEW。 (不限定時間)

```
CREATE VIEW GU_cars_view AS
SELECT
    c.vin,
    m.brand,
    m.model_name,
    m.car_type,
    m.fuel_type,
    m.engine_cc,
    m.transmission,
    m.image_url,
    c.daily_fee,
```

```

c.late_fee,
c.seat_num,
c.notes,
l.loc_name

FROM car c
JOIN model m ON c.model_id = m.model_id
JOIN location l ON c.loc_id = l.loc_id
WHERE c.car_status = 'available';
GRANT SELECT ON car_rental_db.GU_cars_view TO 'guest_user'@'localhost';

```

SELECT VIEW 結果:

vin	brand	model_name	car_type	fuel_type	engine_cc	transmission	image_url	daily_fee	late_fee	seat_num	notes	loc_name
CMBJ1234567890123	CMC	JSPACE	MPV	Gasoline	1500	0	img/CNC_JSPACE.jpg	1700	200	8	空間大，適合家庭出遊	臺中市霧峰店
CMBJ1234567890124	CNC	JSPACE	MPV	Gasoline	1500	0	img/CNC_JSPACE.jpg	1700	200	8	空間大，適合家庭出遊	臺中市霧峰店
CM9999999999999999	BMW	5 Series	Sedan	Gasoline	2000	1	img/bmw_5series.jpg	2200	250	5	空間舒適，適合家庭旅遊	臺中市霧峰店
LM9999999999999999	HYUNDAI	Tucson	SUV	Gasoline	2000	1	img/tucson_2020.jpg	2000	250	5	空間舒適	臺中市霧峰店
JMBGR3116JU123456	MITSUBISHI	Grand Lancer	Compact	Gasoline	1500	1	img/glancer.jpg	1700	200	5	內裝整潔，冷氣超強	臺中市霧峰店
IN4A11APXJC123899	NISSAN	Leaf 2021	Compact	Electric	0	1	img/leaf_2021.jpg	1800	200	5	行動方便	臺中市霧峰店
GT2SE1234567890123	TOYOTA	Altis 12	Compact	Electric	0	1	img/altis12.jpg	1800	200	5	可上路	臺中市霧峰店
HT1BE16K07U123456	TOYOTA	Altis 12	Compact	Hybrid	1800	1	img/Altis12.jpg	1800	200	5	可以隨心所欲	臺中市霧峰店
HT1BE16K07U123789	TOYOTA	RAV4 2022	SUV	Hybrid	1800	1	img/rav4_2022.jpg	2200	300	5	可以隨心所欲SUV	臺中市霧峰店
HT3AV39K0DPA456781	TOYOTA	RAV4 2022	SUV	Hybrid	2500	1	img/rav4_2022.jpg	2200	300	5	可以隨心所欲	臺中市霧峰店

3. 查詢所有車輛的已被租用時間並篩選租用歸還時間是大於目前時間的 VIEW。用於客戶查詢該車輛可租用時間。

```

CREATE VIEW GU_rental_schedule_view AS
SELECT
    r.car_vin as vin,
    r.start_date,
    r.end_date
FROM rental r
JOIN car c ON r.car_vin = c.vin
WHERE r.rental_status IN ('pending', 'active')
    AND r.end_date > NOW()
ORDER BY r.car_vin, r.start_date;
GRANT SELECT ON car_rental_db.GU_rental_schedule_view TO 'guest_user'@'localhost';

```

SELECT VIEW 結果

vin	start_date	end_date
1N4A11APXJC123899	2025-07-11 21:30:00	2025-07-13 21:00:00
5FNRL38647B654321	2025-06-19 05:00:00	2025-06-19 23:00:00
5FNRL38647B654321	2025-07-15 07:03:00	2025-07-18 16:06:00
CMBJ1234567890124	2025-07-14 11:05:00	2025-07-17 23:45:00

應用於開發時下查詢條件，進而知道該車輛哪些時段不可租用。

```

MariaDB [car_rental_db]> SELECT * FROM GU_rental_schedule_view WHERE vin = '5FNRL38647B654321';
+-----+-----+-----+
| vin   | start_date      | end_date       |
+-----+-----+-----+
| 5FNRL38647B654321 | 2025-06-19 05:00:00 | 2025-06-19 23:00:00 |
| 5FNRL38647B654321 | 2025-07-15 07:03:00 | 2025-07-18 16:06:00 |
+-----+-----+-----+
2 rows in set (0.001 sec)

```

4. 將前兩個 VIEW 做 JOIN，顯示所有租用車輛的租用時間。用於客戶查詢指定時段有哪些車輛可租用(包含車輛資訊)。

```

CREATE VIEW GU_car_with_rented_time_view AS
SELECT
    c.vin,
    c.brand,
    c.seat_num,
    c.model_name,
    c.daily_fee,
    c.car_type,
    c.fuel_type,
    c.loc_name,
    engine_cc,
    transmission,
    image_url,
    t.start_date,
    t.end_date
FROM GU_cars_view as c
LEFT JOIN GU_rental_schedule_view t ON c.vin = t.vin;
GRANT SELECT ON car_rental_db.GU_car_with_rented_time_view TO 'guest_user'@'localhost';

```

SELECT VIEW 結果

vin	brand	seat_num	model_name	daily_fee	car_type	fuel_type	loc_name	engine_cc	transmission	image_url	start_date	end_date
CMBJ1234567890123	CNC	8	JSPACE	1780	MPV	Gasoline	臺中市霧峰店	1500	0	img/CNC_JSPACE.jpg	NULL	NULL
CMBJ1234567890124	CNC	8	JSPACE	1780	MPV	Gasoline	臺林縣斗六店	1500	0	img/CNC_JSPACE.jpg	2025-07-14 11:05:00	2025-07-17 23:45:00
SFNRL386478654321	HONDA	7	Odyssey 2019	2280	MPV	Gasoline	臺林縣虎尾店	2500	1	img/Odyssey12.jpg	2025-07-15 07:03:00	2025-07-18 16:06:00
SFNRL386478654322	HONDA	7	Odyssey 2019	2280	MPV	Gasoline	臺林縣虎尾店	2500	1	img/Odyssey12.jpg	2025-06-19 05:00:00	2025-06-19 23:00:00
1N4A111PAKX123456	NISSAN	5	Leaf 2021	2280	MPV	Gasoline	臺中市南屯店	2000	1	img/leaf_2021.jpg	NULL	NULL
1N4A111PAKX123456	NISSAN	5	Leaf 2021	1780	Compact	Gasoline	臺中市南屯店	1500	1	img/glacier.jpg	NULL	NULL
SVJ3811ATHF123456	TESLA	5	Model 3	2580	Sedan	Electric	高市旗津市臺灣店	0	1	img/model3.jpg	2025-07-11 21:30:00	2025-07-13 21:00:00
1T1B646K0XV123456	TOYOTA	5	Altis 12	1880	Compact	Hybrid	臺中市南屯六店	1800	1	img/Altis12.jpg	NULL	NULL
1T1B646K0XV123789	TOYOTA	5	Altis 12	1880	Compact	Hybrid	臺林縣虎尾店	1800	1	img/Altis12.jpg	NULL	NULL
4T3H36KXDX456781	TOYOTA	5	RAV4 2022	2280	SUV	Hybrid	臺林縣虎尾店	2500	1	img/raw4_2022.jpg	NULL	NULL

應用於開發時下查詢條件，進而知道該車輛哪些時段不可租用。

```

mariadb [car_rental_db]> SELECT *
    FROM GU_car_with_rented_time_view
   WHERE car_type = 'MPV'
     AND fuel_type = 'Gasoline'
     AND start_date IS NULL OR end_date IS NULL
     OR (
        start_date <= '2025-07-16 18:00:00' AND end_date >= '2025-07-15 18:00:00'
     );
2 rows in set (0.00 sec)

```

vin	brand	seat_num	model_name	daily_fee	car_type	fuel_type	loc_name	engine_cc	transmission	image_url	start_date	end_date
CMBJ1234567890123	CNC	8	JSPACE	1780	MPV	Gasoline	臺中市霧峰店	1500	0	img/CNC_JSPACE.jpg	NULL	NULL
SFNRL386478654321	HONDA	7	Odyssey 2019	2280	MPV	Gasoline	臺林縣虎尾店	2500	1	img/Odyssey12.jpg	2025-06-19 05:00:00	2025-06-19 23:00:00

123 · 租車

篩選條件

選擇查詢點
臺林縣虎尾店

選擇日期：
□ 今天使用
2025-06-14 - 2025-06-16

租車時間： 上午 09:00 下午 06:00

選擇類型：
 不限 小型車 中大型房車 SUV MPV

車輛狀態：
正常

每日租金範圍：
0 - 6000

OK 取消

5. 熱門租用車型 Top 5

```
CREATE VIEW GU_top5_models_view AS
SELECT
    m.brand,
    m.model_name,
    m.fuel_type,
    COUNT(*) AS rental_count
FROM rental r
JOIN car c ON r.car_vin = c.vin
JOIN model m ON c.model_id = m.model_id
GROUP BY m.model_id
ORDER BY rental_count DESC
LIMIT 5;

GRANT SELECT ON car_rental_db.GU_top5_models_view TO 'guest_user'@'localhost';
```

SELECT VIEW 結果

```
MariaDB [car_rental_db]> SELECT * FROM GU_top5_models_view;
+-----+-----+-----+-----+
| brand | model_name | fuel_type | rental_count |
+-----+-----+-----+-----+
| TOYOTA | Altis 12 | Hybrid | 3 |
| HYUNDAI | Tucson 2029 | Gasoline | 2 |
| HONDA | Odyssey 2019 | Gasoline | 2 |
| NISSAN | Leaf 2021 | Electric | 1 |
| CMC | JSPACE | Gasoline | 1 |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)
```

6. 各汽車品牌提供車輛數

```
CREATE VIEW view_brand_car_count AS
SELECT m.brand, COUNT(*) AS car_count
FROM car c
JOIN model m ON c.model_id = m.model_id
GROUP BY m.brand;

GRANT SELECT ON car_rental_db.view_brand_car_count TO 'guest_user'@'localhost';
```

SELECT VIEW 結果

```
MariaDB [car_rental_db]> SELECT * FROM view_brand_car_count;
+-----+-----+
| brand | car_count |
+-----+-----+
| CMC | 2 |
| FORD | 1 |
| HONDA | 1 |
| HYUNDAI | 1 |
| KIA | 2 |
| MITSUBISHI | 1 |
| NISSAN | 2 |
| TESLA | 1 |
| TOYOTA | 4 |
+-----+-----+
9 rows in set (0.001 sec)
```

其餘 VIEW 權限外，guest_user 還具備以下操作權限，可針對 location , insurance Table (分店地點、保險方案)進行查詢，以及能新增會員資料到 member Table (註冊會員)。

```
GRANT INSERT ON car_rental_db.member TO 'guest_user'@'localhost';
GRANT SELECT ON car_rental_db.location TO 'guest_user'@'localhost';
GRANT SELECT ON car_rental_db.insurance TO 'guest_user'@'localhost';
```

2. customer_user

1. 查詢會員 google_id, phone 的 VIEW。用於會員查詢並修改綁定的電話號碼。

```
CREATE VIEW CM_member_phone_VIEW AS
SELECT google_id, phone FROM member;
GRANT SELECT,UPDATE ON car_rental_db.CM_member_phone_VIEW TO
'customer_user'@'localhost';
```

SELECT VIEW 結果

```
MariaDB [car_rental_db]> SELECT * FROM CM_member_phone_VIEW;
+-----+-----+
| google_id | phone |
+-----+-----+
| 213456789012345678901 | 0912333444 |
| 111996782304424104647 | 0914238500 |
| 324567890123456789012 | 0923444555 |
| 342057390258473869423 | 0923969027 |
| 21578423994712938765 | 0931164655 |
| 192703829478239801723 | 0932535148 |
| 435678901234567890123 | 09345555666 |
| 567890123456789012345 | 09445555666 |
| 546789012345678901234 | 09456667777 |
| 678901234567890123456 | 09556667777 |
| 789012345678901234567 | 0966777888 |
| 657890123456789012345 | 0978985123 |
| 879012345678901234567 | 0978999000 |
| 123456789012345678901 | 0987123456 |
| 102345678901234567890 | 0999000111 |
+-----+-----+
15 rows in set (0.001 sec)
```

應用於開發時下更新。

```
MariaDB [car_rental_db]> UPDATE CM_member_phone_VIEW SET phone='0922222222' WHERE google_id='213456789012345678901';
Query OK, 1 row affected (0.005 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [car_rental_db]> SELECT * FROM CM_member_phone_VIEW;
+-----+-----+
| google_id | phone |
+-----+-----+
| 111996782304424104647 | 0914238500 |
| 213456789012345678901 | 0922222222 |
| 324567890123456789012 | 0923444555 |
| 342057390258473869423 | 0923969027 |
| 21578423994712938765 | 0931164655 |
| 192703829478239801723 | 0932535148 |
| 435678901234567890123 | 09345555666 |
| 567890123456789012345 | 09445555666 |
| 546789012345678901234 | 09456667777 |
| 678901234567890123456 | 09556667777 |
| 789012345678901234567 | 0966777888 |
| 657890123456789012345 | 0978985123 |
| 879012345678901234567 | 0978999000 |
| 123456789012345678901 | 0987123456 |
| 102345678901234567890 | 0999000111 |
+-----+-----+
```

2. 查詢簡略租賃紀錄的 VIEW，提供會員快速查看歷史租賃紀錄。

```
CREATE VIEW CM_rental_basic_view AS
SELECT
    r.google_id,
    r.start_date,
    r.end_date,
    r.rental_status,
    r.amount,
    c.vin
FROM rental r
JOIN car c ON r.car_vin = c.vin;
GRANT SELECT ON car_rental_db.CM_rental_full_view TO 'customer_user'@'localhost';
```

SELECT VIEW 結果

MariaDB [car_rental_db]> SELECT * FROM CM_rental_basic_view;							
google_id	start_date	end_date	rental_status	amount	vin		
102345678901234567890	2025-06-23 09:30:00	2025-06-27 18:00:00	completed	10000	KM8K6CA43LU123456		
111996782304424104647	2025-07-11 21:30:00	2025-07-13 21:00:00	active	4400	1N4A11APXJC123899		
192703829478239801723	2025-06-18 13:10:00	2025-06-20 00:40:00	completed	5200	5YJ3E1AE7KF123456		
213456789012345678901	2025-07-12 07:30:00	2025-07-13 20:00:00	cancelled	5200	KM8K6CA43LU123456		
213456789012345678901	2025-07-15 07:03:00	2025-07-18 16:06:00	active	18800	5FNRL38647B654321		
213456789012345678901	2025-06-19 05:00:00	2025-06-19 23:00:00	active	3800	5FNRL38647B654321		
32456789012345678901	2025-06-02 06:03:00	2025-06-04 19:05:00	cancelled	7800	4T3HV36KXPX456781		
435678901234567890123	2025-06-07 20:00:00	2025-06-10 19:05:00	completed	10800	4T1BE46KX7U123456		
657890123456789012345	2025-07-14 11:05:00	2025-07-17 23:45:00	active	12400	CMBJ1234567890124		
879012345678901234567	2025-06-13 08:00:00	2025-06-14 17:00:00	cancelled	6400	4T1BE46KX7U123456		
879012345678901234567	2025-06-02 08:00:00	2025-06-05 17:00:00	reject	8000	4T1BE46KX7U123456		

- 3.查詢租賃明細的 VIEW，提供會員點進明細時查詢租賃紀錄。

```
CREATE VIEW CM_rental_full_view AS
```

SELECT

r.google_id,
r.created_at,
r.start_date,
r.end_date,
r.rental_status,
r.amount,
r.method,
r.actual_drop_loc,
r.actual_pickup,
m.brand,
m.model_name,
m.car_type,
m.fuel_type,
m.engine_cc,
m.transmission,
m.image_url,
c.daily_fee,
c.seat_num

FROM rental r

JOIN car c ON r.car_vin = c.vin

JOIN model m ON c.model_id = m.model_id;

```
GRANT SELECT ON car_rental_db.CM_rental_full_view TO 'customer_user'@'localhost';
```

SELECT VIEW 結果

應用於開發時查詢。

```
MariaDB [car_rental_db]> SELECT * FROM CM_rental_full_view WHERE google_id = '879012345678901234567';
```

google_id	created_at	start_date	end_date	rental_status	amount	method	actual_drop_loc	actual_pickup	brand	model_name	car_type	fuel_type	engine_cc	transmission	image_url	daily_fee	seat_num
879012345678901234567	2025-05-25 13:25:08	2025-06-13 08:00:00	2025-06-14 17:00:00	cancelled	6480	cash	NULL	NULL	TOYOTA	Altis 12	Compact	Hybrid	1800	1	img/Altis12.jpg	1880	5
879012345678901234567	2025-06-01 13:09:21	2025-06-02 08:00:00	2025-06-05 17:00:00	reject	8080	cash	NULL	NULL	TOYOTA	Altis 12	Compact	Hybrid	1800	1	img/Altis12.jpg	1880	5

2 rows in set (0.001 sec)

4.針對會員尚未結案的訂單，能對部分欄位更新操作。如:保險方案、取消訂單。

```
CREATE VIEW CM_rental_updatable_view AS  
SELECT  
    google_id,  
    created_at,  
    rental_status,  
    insurance_id  
FROM rental;  
  
GRANT UPDATE ON car_rental_db.CM_rental_updatable_view TO 'customer_user'@'localhost';
```

SELECT VIEW 結果

```
MariaDB [car_rental_db]> SELECT * FROM CM_rental_updatable_view;
```

google_id	created_at	rental_status	insurance_id
102345678901234567890	2025-06-15 11:44:58	completed	Y255001
111996782304424104647	2025-07-08 19:22:26	active	Y255003
19278329478239801723	2025-06-14 18:59:20	completed	Y25E001
213456789012345678901	2025-06-01 19:45:39	cancelled	Y255004
213456789012345678901	2025-06-02 13:13:33	active	Y257002
213456789012345678901	2025-06-12 08:11:43	active	Y25X003
324567890123456789012	2025-06-01 12:22:28	cancelled	Y255003
435678901234567890123	2025-06-03 09:25:35	completed	Y25Z001
657890123456789012345	2025-05-01 13:39:19	active	Y25X002
879012345678901234567	2025-05-25 13:25:08	cancelled	Y25X002
879012345678901234567	2025-06-01 13:09:21	reject	Y255002

11 rows in set (0.001 sec)

其餘 VIEW 權限外，customer_user 還具備以下操作權限，可針對 location , insurance Table(分店地點、保險方案)進行查詢，以及能新增會員資料到 member Table (註冊會員)。

```
GRANT SELECT ON car_rental_db.location TO 'customer_user'@'localhost';  
GRANT SELECT ON car_rental_db.insurance TO 'customer_user'@'localhost';  
GRANT INSERT ON car_rental_db.rental TO 'customer_user'@'localhost';
```

然而為了確保在註冊後不會失去基本查詢功能，customer_user 也能使用 guest_user 所提供的車輛與租賃資訊查詢介面。

```
GRANT SELECT ON car_rental_db.GU_cars_view TO 'customer_user'@'localhost';  
GRANT SELECT ON car_rental_db.GU_rental_schedule_view TO 'customer_user'@'localhost';  
GRANT SELECT ON car_rental_db.GU_car_with_rented_time_view TO 'customer_user'@'localhost';  
GRANT SELECT ON car_rental_db.view_brand_car_count TO 'customer_user'@'localhost';  
GRANT SELECT ON car_rental_db.GU_top5_models_view TO 'customer_user'@'localhost';
```

3. staff_admin

為租車系統中的後台管理員角色，需要高層級的資料檢視與管理能力。

1. 快速瀏覽租賃紀錄(列表)的 VIEW，僅作查帳或報表列出。

```
CREATE VIEW AD_rental_summary_view AS
SELECT
    r.google_id,
    r.created_at,
    r.start_date,
    r.end_date,
    r.rental_status,
    r.amount,
    r.car_vin
FROM rental r;
```

```
GRANT SELECT ON car_rental_db.AD_rental_summary_view TO 'staff_user'@'localhost';
```

SELECT VIEW 結果

google_id	created_at	start_date	end_date	rental_status	amount	car_vin
102345678901234567890	2025-06-15 11:44:58	2025-06-23 09:00:00	2025-06-27 18:00:00	completed	10000	KMBK6CA43LU123456
11199678230424104647	2025-07-01 19:22:26	2025-07-11 21:30:00	2025-07-13 21:00:00	active	4400	1N4A11APXJC123899
192703829478239801723	2025-06-14 18:59:20	2025-06-18 13:10:00	2025-06-20 00:40:00	completed	5200	5Y3E1EA7KF123456
213456789012345678901	2025-06-01 19:45:39	2025-07-12 07:30:00	2025-07-13 20:00:00	cancelled	5200	KMBK6CA43LU123456
213456789012345678901	2025-06-02 13:13:33	2025-07-15 07:03:00	2025-07-18 16:06:00	active	18800	5FNR138647B654321
213456789012345678901	2025-06-12 08:11:43	2025-06-19 05:00:00	2025-06-19 23:00:00	active	3800	5FNR138647B654321
324567890123456789012	2025-06-01 12:22:28	2025-06-02 06:03:00	2025-06-04 19:05:00	cancelled	7800	4T3H46KX7U123456
435678901234567890123	2025-06-03 09:25:35	2025-06-07 20:00:00	2025-06-10 19:05:00	completed	10800	4T1BE46KX7U123456
657890123456789012345	2025-05-01 13:39:19	2025-07-14 11:05:00	2025-07-17 23:45:00	active	12400	CMBJ1234567890124
879012345678901234567	2025-05-25 13:25:08	2025-06-13 08:00:00	2025-06-14 17:00:00	cancelled	6400	4T1BE46KX7U123456
879012345678901234567	2025-06-01 13:09:21	2025-06-02 08:00:00	2025-06-05 17:00:00	reject	8000	4T1BE46KX7U123456

2. 查詢每筆狀態訂單筆數

```
CREATE VIEW AD_rental_status_count_view AS
SELECT
    CASE
        WHEN r.rental_status IN ('canceled', 'rejected') THEN 'canceled'
        WHEN r.rental_status = 'active' AND NOW() BETWEEN r.start_date AND
r.end_date THEN 'active_ongoing'
        WHEN r.rental_status = 'active' AND NOW() < r.start_date THEN
'active_waiting'
        WHEN r.rental_status = 'completed' THEN 'completed'
        ELSE 'other'
    END AS rental_status_group,
    COUNT(*) AS total_orders
FROM rental r
GROUP BY rental_status_group
```

```

UNION ALL
SELECT
    'TOTAL' AS rental_status_group,
    COUNT(*) AS total_orders
FROM rental;
GRANT SELECT ON car_rental_db.AD_rental_summary_view TO 'staff_user'@'localhost';

```

SELECT VIEW 結果

```

MariaDB [car_rental_db]> SELECT * FROM AD_rental_status_count_view;
+-----+-----+
| rental_status_group | total_orders |
+-----+-----+
| active_waiting      |        4 |
| completed           |        3 |
| other                |        4 |
| TOTAL               |       11 |
+-----+-----+
4 rows in set (0.001 sec)

```

應用於開發介面(示意圖)



3.點選某筆訂單後顯示詳細內容的 VIEW，給客服或租賃部門調查用。此 VIEW 還需要搭配二階段查詢保險名稱、費用以及取車還車地點店名查詢，以避免多張表進行 JOIN 效率低問題。

```

CREATE VIEW AD_rental_detail_view AS
SELECT
    r.google_id,
    r.created_at,
    r.start_date,
    r.end_date,
    r.actual_return,
    r.rental_status,
    r.amount,
    r.method,

```

```
r.pickup_loc,  
r.drop_loc,  
r.insurance_id,  
c.vin,  
m.model_name,  
m.brand,  
c.daily_fee,  
c.late_fee  
FROM rental r  
JOIN car c ON r.car_vin = c.vin  
JOIN model AS m ON c.model_id = m.model_id;  
GRANT SELECT ON car_rental_db.AD_rental_detail_view TO 'staff_admin'@'localhost';
```

SELECT VIEW 結果

name	db	SELECT * FROM AD_rental_detail_view;
1023356789012345678901	2025-06-23 11:45:08	2025-06-23 09:30:00 2025-06-27 17:00:00 completed 16000 cash TOWF CHIC Y255801 N0886CAH3LU123456 Tucson 2020 HYUNDAI 2000 250
11196789012345678901	2025-07-08 19:22:26	2025-07-01 21:30:00 2025-07-01 21:00:00 active 4000 cash Linepay TONT Y255803 1N4A11ACJ1C3123899 Leaf 2021 NISSAN 1800 260
21345789012345678901	2025-06-19 19:45:59	2025-06-19 19:45:59 2025-06-20 06:30:00 completed 5200 cash TOWF CHIC Y255804 N0886CAH3LU123456 Tucson 2020 HYUNDAI 2000 250
21345789012345678901	2025-06-02 13:13:33	2025-07-05 07:03:00 2025-07-05 18:00:00 cancelled 5200 cash TOWF CHIC Y255804 N0886CAH3LU123456 Tucson 2020 HYUNDAI 2000 250
2350789012345678901	2025-06-02 17:22:00	2025-06-02 06:00:00 2025-06-02 19:05:00 completed 7600 cash YUNH Y255803 GT3MV900X00056781 RAV4 2022 TOYOTA 2000 380
2350789012345678901	2025-06-02 17:22:00	2025-06-02 06:00:00 2025-06-02 19:05:00 cancelled 7600 cash YUNH Y255803 GT3MV900X00056781 RAV4 2022 TOYOTA 2000 380
435678901234567890123	2025-06-03 09:25:35	2025-06-07 20:00:00 2025-06-10 19:00:00 completed 16000 cash YUNH Y255801 N1B1E6G0U70123456 Altis 12 TOYOTA 1800 260
7890123456789012345678901	2025-06-13 08:00:00	2025-06-14 17:00:00 2025-06-14 17:00:00 cancelled 6400 cash YUNH Y255802 N1B1E6G0U70123456 Altis 12 TOYOTA 1800 260
7890123456789012345678901	2025-06-01 13:09:21	2025-06-02 08:00:00 2025-06-05 17:00:00 reject 8000 cash YUNH Y255802 N1B1E6G0U70123456 Altis 12 TOYOTA 1800 260

其餘 VIEW 權限外，賦予 staff_admin 對其多個資料表(location、insurance、model、car)的新增、查詢、修改與刪除權限，對 member 資料表僅能查詢、rental 租用資料表僅查詢和更新。而在本次報告中，為了管理彈性與測試需求，開放刪除權限。然而在實務應用上，資料刪除可能導致資訊遺失，因此未來系統可考慮導入軟刪除機制，使刪除操作僅標記資料為"已刪除"，而非從資料庫中永久移除。如此就可在不開放角色刪除權限的情況下，也能達成管理員所需的資料刪除(隱藏)，同時也能保留資料備查與還原的彈性。

```
GRANT SELECT ON car_rental_db.member TO 'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.location TO 'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.insurance TO 'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.model TO 'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.car TO 'staff_admin'@'localhost';
GRANT SELECT, UPDATE ON car_rental_db.rental TO 'staff_admin'@'localhost';
```

柒、完整 SQL 統整

DDL (schema)

```
CREATE DATABASE car_rental_db;
USE car_rental_db;

-- Create member Table
CREATE TABLE member(
    google_id VARCHAR(22) NOT NULL PRIMARY KEY,
    gmail VARCHAR(50) NOT NULL UNIQUE KEY,
    phone VARCHAR(10) NOT NULL UNIQUE KEY,
    CHECK (google_id REGEXP '^[0-9]{21,22}$'),
    CHECK (gmail REGEXP '^[a-zA-Z0-9._%+-]{6,30}@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$'),
    CHECK (phone REGEXP '^09[0-9]{8}$')
);

-- Create location
CREATE TABLE location (
    loc_id VARCHAR(4) NOT NULL PRIMARY KEY CHECK (loc_id REGEXP '^[A-Z]{4}$'),
    loc_name VARCHAR(6) NOT NULL CHECK (loc_name REGEXP '^[一-龜 A-Za-z]{2,5}店$'),
    city VARCHAR(3) NOT NULL CHECK (city REGEXP '^(臺北市|新北市|桃園市|臺中市|臺南市|高雄市|基隆市|新竹市|嘉義市|新竹縣|苗栗縣|彰化縣|南投縣|雲林縣|嘉義縣|屏東縣|宜蘭縣|花蓮縣|臺東縣|澎湖縣|金門縣|連江縣)$'),
    district VARCHAR(4) NOT NULL CHECK(district REGEXP '^[一-龜]{1,3}(區|市|鎮|鄉)$'),
    address VARCHAR(50) NOT NULL,
    CHECK (address REGEXP '^([-龜]{1,3}(村|里))?(0-9){1,3}鄰)?[-龜]+(路|街|大道)(一二三四五六七八九十){1,3}段)?(0-9)+巷)?(0-9)+弄)?(0-9){1,4}之)?0-9){1,4}號)?(0-9){1,3}樓)?(0-9)+)?)[-龜 A-Za-z0-9]+室)?$')
);

-- Create insurance Table
CREATE TABLE insurance(
    insurance_id VARCHAR(7) NOT NULL PRIMARY KEY CHECK (insurance_id REGEXP '^[A-Z0-9]{7}$'),
    ins_name VARCHAR(20) NOT NULL UNIQUE KEY CHECK (ins_name REGEXP '^[一-龜 A-Za-z\\(\\) ]{3,15}$'),
    coverage VARCHAR(250) NOT NULL CHECK (coverage REGEXP '^[一-龜 A-Za-z0-9\\(\\) , 。 、 ( ) : : 「 」 ]{1,250}$')
);
```

```

    ins_fee INT NOT NULL CHECK (ins_fee >= 0)
);

-- Create model Table
CREATE TABLE model(
    model_id VARCHAR(7) NOT NULL PRIMARY KEY CHECK (model_id REGEXP '^[A-Z0-
9]{7}$'),
    brand VARCHAR(20) NOT NULL CHECK (brand REGEXP '^[A-Za-z]{1,20}$'),
    model_name VARCHAR(20) NOT NULL CHECK (model_id REGEXP '^[A-Za-z0-9-
]{1,20}$'),
    car_type VARCHAR(7) NOT NULL CHECK (car_type IN ('Compact', 'Sedan', 'SUV',
'MPV')),
    fuel_type VARCHAR(8) NOT NULL CHECK (fuel_type IN ('Gasoline', 'Electric', 'Hybrid')),
    engine_cc INT NOT NULL CHECK (engine_cc >= 0),
    transmission INT NOT NULL CHECK (transmission IN (0, 1)),
    image_url VARCHAR(50) NOT NULL CHECK (image_url REGEXP '^img\\/[A-Za-z0-
9_]+\\.\\(jpg|png\\)$')
);

-- Create car Table
CREATE TABLE car (
    vin VARCHAR(17) NOT NULL PRIMARY KEY CHECK (vin REGEXP '^[A-HJ-NPR-Za-hj-npr-
z0-9]{8}[0-9X][A-HJ-NPR-Za-hj-npr-z0-9]{2}[0-9]{6}$'),
    plate_number VARCHAR(8) NOT NULL UNIQUE KEY CHECK (plate_number REGEXP
'^R[A-Z]{2}-[0-35-9]{4}$'),
    daily_fee INT NOT NULL CHECK (daily_fee > 0),
    late_fee INT NOT NULL CHECK (late_fee > 0),
    year_made INT NOT NULL CHECK (year_made >= 1980),
    seat_num INT NOT NULL CHECK (seat_num > 0 and seat_num < 16),
    color VARCHAR(5) NOT NULL CHECK (color REGEXP '^[一-龥]{1,4}色$'),
    mileage INT NOT NULL CHECK (mileage > 0),
    car_status VARCHAR(12) NOT NULL DEFAULT 'available' CHECK (car_status IN
('available', 'maintenance', 'disable')),
    notes TEXT,
    model_id VARCHAR(7) NOT NULL,
    loc_id VARCHAR(4),
    FOREIGN KEY (model_id) REFERENCES model(model_id)
        ON DELETE NO ACTION,

```

```

    FOREIGN KEY (loc_id) REFERENCES location(loc_id)
    ON DELETE SET NULL
);

-- Create rental Table
CREATE TABLE rental (
    google_id VARCHAR(22) NOT NULL,
    created_at DATETIME NOT NULL,
    start_date DATETIME NOT NULL,
    end_date DATETIME NOT NULL,
    actual_pickup DATETIME,
    actual_return DATETIME,
    rental_status VARCHAR(9) NOT NULL CHECK (rental_status IN ('pending', 'active',
'completed', 'cancelled', 'reject')),
    amount INT NOT NULL CHECK (amount >= 0),
    payment_date DATETIME,
    method VARCHAR(8) NOT NULL CHECK (method IN ('cash', 'credit', 'linepay')),
    car_vin VARCHAR(17) NOT NULL,
    pickup_loc VARCHAR(4) NOT NULL,
    drop_loc VARCHAR(4) NOT NULL,
    actual_drop_loc VARCHAR(4),
    insurance_id VARCHAR(7) NOT NULL,
    CONSTRAINT fk_google_id FOREIGN KEY (google_id) REFERENCES member(google_id)
        ON DELETE CASCADE,
    CONSTRAINT fk_car_vin FOREIGN KEY (car_vin) REFERENCES car(vin)
        ON DELETE NO ACTION,
    CONSTRAINT fk_pickup_loc FOREIGN KEY (pickup_loc) REFERENCES location(loc_id)
        ON DELETE NO ACTION,
    CONSTRAINT fk_drop_loc FOREIGN KEY (drop_loc) REFERENCES location(loc_id)
        ON DELETE NO ACTION,
    CONSTRAINT fk_actual_drop_loc FOREIGN KEY (actual_drop_loc) REFERENCES
location(loc_id)
        ON DELETE SET NULL,
    CONSTRAINT fk_insurance_id FOREIGN KEY (insurance_id) REFERENCES
insurance(insurance_id)
        ON DELETE NO ACTION,
    PRIMARY KEY (google_id,created_at)
);

```

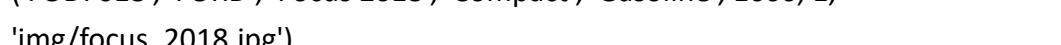
DML (record)

```
USE car_rental_db;

-- INSERT member Table
INSERT INTO member (google_id, gmail, phone) VALUES
('879012345678901234567', 'quincy_liu@gmail.com', '0978999000'),
('657890123456789012345', 'kulzia@gmail.com', '0978985123'),
('111996782304424104647', 'gene_st@gmail.com', '0914238500'),
('192703829478239801723', 'Wen_Che@gmail.com', '0932535148'),
('215784230947120938765', 'asd_tw@gmail.com', '0931164655'),
('342057390258473869423', 'kevin_jj@gmail.com', '0923969027'),
('102345678901234567890', 'jane_liu@gmail.com', '0999000111'),
('213456789012345678901', 'kevin_liu@yahoo.com', '0912333444'),
('324567890123456789012', 'lisa_wang@nfu.edu.tw', '0923444555'),
('435678901234567890123', 'mike_smith@wistron.com', '0934555666'),
('546789012345678901234', 'nancy_johnson@foxconn.com', '0945666777'),
('567890123456789012345', 'ethan.lu@yahoo.com', '0944555666'),
('678901234567890123456', 'fiona.lee@nfu.edu.tw', '0955666777'),
('789012345678901234567', 'george.lin@gmail.com', '0966777888'),
('123456789012345678901', 'alice_wu@gmail.com', '0987123456');

-- INSERT location Data
INSERT INTO location (loc_id, loc_name, city, district, address) VALUES
('YUNH', '雲林縣虎尾店', '雲林縣', '虎尾鎮', '文化路 64 號'),
('YUNL', '雲林縣斗六店', '雲林縣', '阿里山鄉', '大學路三段 123 號'),
('CHIE', '嘉義市東區店', '嘉義市', '東區', '學府路 300 號'),
('CHIC', '嘉義縣民雄店', '嘉義縣', '民雄鄉', '神農路 145 號'),
('YUNN', '雲林縣斗南店', '雲林縣', '斗南鎮', '大業路 106 號'),
('TCWF', '臺中市霧峰店', '臺中市', '霧峰區', '吉峰東路 168 號'),
('TCNT', '臺中市南屯店', '臺中市', '南屯區', '大墩路 200 號'),
('TPXY', '臺北市信義店', '臺北市', '信義區', '松山路 100 號'),
('NTPC', '新北市板橋店', '新北市', '板橋區', '文化路二段 50 號'),
('KHLG', '高雄市苓雅店', '高雄市', '苓雅區', '和平一路 300 號'),
('KHCZ', '高雄市前鎮店', '高雄市', '前鎮區', '中華五路 88 號'),
('TNED', '臺南市東區店', '臺南市', '東區', '大學路 100 號'),
('TNNB', '臺南市北區店', '臺南市', '北區', '成功路 200 號'),
('HSIN', '新竹市東區店', '新竹市', '東區', '光復路一段 150 號'),
('TYCL', '桃園市中壢店', '桃園市', '中壢區', '中正路 300 號');
```

```
-- INSERT insurance Data
INSERT INTO insurance (insurance_id, ins_name, coverage, ins_fee) VALUES
('Y25S001', '基本保險方案', '第三人責任保險：每人傷害上限 200 萬、每事故傷害上限 400 萬、財損上限 50 萬。駕駛人保險 100 萬、乘客每人 100 萬（超載除外）。', 0),
('Y25S002', '第三人責任升級', '每人傷害上限提升至 500 萬、每事故傷害 1000 萬、財損上限 200 萬。其餘條件比照基本方案。', 200),
('Y25S003', '第三人責任與車體損害升級版', '每人傷害上限提升至 500 萬、每事故傷害 1000 萬、財損上限 200 萬。其餘規定比照基本保險。', 400),
('Y25S004', '豪華第三人責任險', '每人傷害上限 800 萬、每事故傷害 1500 萬、財損上限 300 萬，乘客保險每人 200 萬。不含車體損害。', 600),
('Y25X001', '全方位保險方案', '涵蓋第三人責任升級版保障，並新增車體損失險（含碰撞、自撞、翻車、他人肇逃）、竊盜險。自負額每次事故 5000 元。', 800),
('Y25X002', '高階全險(免自負額)', '等同全方位保險方案，另提供不限事故次數之免自負額保障。亦納入每日車輛停駛補償保險。', 1400),
('Y25E001', '節能保險方案', '提供基本保險保障，專為電動車設計，另含電池損害與火災保險（限電動車適用）。', 100),
('Y25X003', '超級全險（海外人士專用）', '提供全方位保險保障，另納入外籍駕駛法律援助保險與語言協助。', 1600),
('Y25Z001', '商務菁英保險方案', '含高階全險保障，另增加行程延誤補償與駕駛更換服務，每日最高補償 3000 元。', 1800),
('Y25Z002', '至尊全險方案', '含所有前述保險保障，另提供不限次數道路救援、全天候代步車服務、全額車損理賠（免自負額）。', 2500);
```

```
-- INSERT model Data
INSERT INTO model (model_id, brand, model_name, car_type, fuel_type,
engine_cc, transmission, image_url) VALUES
('TOTA012', 'TOYOTA', 'Altis 12', 'Compact', 'Hybrid', 1800, 1, 'img/Altis12.jpg'),
('HONO019', 'HONDA', 'Odyssey 2019', 'MPV', 'Gasoline', 2500, 1,
),
('TES3001', 'TESLA', 'Model 3', 'Sedan', 'Electric', 0, 1, 'img/model3.jpg'),
('TOTR022', 'TOYOTA', 'RAV4 2022', 'SUV', 'Hybrid', 2500, 1, 'img/rav4_2022.jpg'),
('MITG001', 'MITSUBISHI', 'Grand Lancer', 'Compact', 'Gasoline', 1500, 1,
),
('FODF018', 'FORD', 'Focus 2018', 'Compact', 'Gasoline', 1600, 1,
),
```

```

('HYDT020', 'HYUNDAI', 'Tucson 2020', 'SUV', 'Gasoline', 2000, 1,
'img/tucson_2020.jpg'),
('NISL021', 'NISSAN', 'Leaf 2021', 'Compact', 'Electric', 0, 1, 'img/leaf_2021.jpg'),
('KIAS022', 'KIA', 'Seltos 2022', 'SUV', 'Gasoline', 1600, 1, 'img/seltos_2022.jpg'),
('CMCJ001', 'CMC', 'JSPACE', 'MPV', 'Gasoline', 1500, 0, 'img/CMC_JSPACE.jpg');

-- INSERT car Data
INSERT INTO car (vin, plate_number, model_id, daily_fee, late_fee, year_made,
loc_id, seat_num, color, mileage, notes, car_status) VALUES
('4T1BE46KX7U123456', 'RAA-7777', 'TOT012', 1800, 200, 2020, 'YUNH', 5, '白色',
15000, '可以隨心所欲', 'available'), -- Altis12
('4T1BE46KXPX123789', 'RAA-8888', 'TOT012', 1800, 200, 2020, 'YUNL', 5, '白色',
16500, '可以隨心所欲', 'available'), -- Altis12
('5FNRL38647B654321', 'RAR-2385', 'HONO019', 2200, 200, 2020, 'YUNH', 7, '黑色',
12000, '座椅舒適，適合家庭出遊', 'available'), -- Odyssey 2019
('5YJ3E1EA7KF123456', 'REK-3556', 'TES3001', 2500, 300, 2021, 'KHLG', 5, '銀色',
8000, '行駛安靜', 'available'), -- TESLA
('4T3HV36KXPX456781', 'RBC-8811', 'TOTR022', 2200, 300, 2022, 'YUNH', 5, '白色',
7500, '可以隨心所欲 SUV', 'available'), -- RAV4 2022
('4T3HV36KXPX456782', 'RBC-9922', 'TOTR022', 2200, 300, 2022, 'CHIE', 5, '白色',
9200, '可以隨心所欲 SUV', 'maintenance'), -- RAV4 2022
('JMBGR31L6JU123456', 'RAF-5678', 'MITG001', 1700, 200, 2019, 'YUNH', 5, '黑色',
15000, '內裝整潔，冷氣超強', 'available'), -- Grand Lancer
('1FADP3F2XJL123456', 'RBC-1239', 'FODF018', 1800, 200, 2018, 'TPXY', 5, '銀色',
18000, '油耗低', 'maintenance'), -- Focus 2018
('KM8K6CA43LU123456', 'RBD-8567', 'HYDT020', 2000, 250, 2020, 'TCWF', 5, '黑色',
5000, '空間寬敞', 'available'), -- Tucson 2020
('1N4A11APXJC123899', 'RBB-6789', 'NISL021', 1800, 200, 2021, 'TCNT', 5, '綠色',
3000, '行駛安靜', 'available'), -- Leaf 2021
('1N4A11APXJC123456', 'RBD-1112', 'NISL021', 1800, 200, 2021, 'TCNT', 5, '綠色',
3000, '行駛安靜', 'disable'), -- Leaf 2021
('KNDJ23AU7L1216211', 'RBZ-2355', 'KIAS022', 1800, 200, 2022, 'CHIC', 5, '藍色',
4000, '小巧靈活，適合城市駕駛', 'disable'), -- Seltos 2022
('KNDJ23AU7L1234567', 'RBS-8875', 'KIAS022', 1800, 200, 2022, 'TPXY', 5, '藍色',
4000, '小巧靈活，適合城市駕駛', 'disable'), -- Seltos 2022
('CMBJ1234567890123', 'RAS-1235', 'CMCJ001', 1700, 200, 2019, 'TCWF', 8, '白色',
8000, '空間大，適合家庭出遊', 'available'), -- CMC JSPACE

```

```

('CMBJ1234567890124', 'RAS-5678', 'CMCJ001', 1700, 200, 2019, 'YUNN', 8, '白色',
5500, '空間大，適合家庭出遊', 'available'); -- CMC JSPACE

-- INSERT rental Data
INSERT INTO rental (created_at, google_id, start_date, end_date, actual_return,
rental_status,amount, payment_date, method, car_vin,
pickup_loc,drop_loc,insurance_id,actual_pickup,actual_drop_loc) VALUES
('2025-06-01 13:09:21', '879012345678901234567', '2025-06-02 08:00:00', '2025-
06-05 17:00:00', NULL, 'reject', 8000, NULL, 'cash', '4T1BE46KX7U123456', 'YUNH',
'YUNH', 'Y25S002',NULL,NULL),
('2025-05-25 13:25:08', '879012345678901234567', '2025-06-13 08:00:00', '2025-
06-14 17:00:00', NULL, 'cancelled', 6400, NULL, 'cash', '4T1BE46KX7U123456',
'YUNH', 'YUNH', 'Y25X002',NULL,NULL),
('2025-05-01 13:39:19', '657890123456789012345', '2025-07-14 11:05:00', '2025-
07-17 23:45:00', NULL, 'active', 12400, NULL, 'cash', 'CMBJ1234567890124',
'YUNN', 'YUNN', 'Y25X002',NULL,NULL),
('2025-06-14 18:59:20', '192703829478239801723', '2025-06-18 13:10:00', '2025-
06-20 00:40:00', '2025-06-20 00:30:00', 'completed', 5200, '2025-06-15 19:25:00',
'linepay', '5YJ3E1EA7KF123456', 'KHLG', 'KHLG', 'Y25E001','2025-06-18
13:05:20','KHLG'),
('2025-07-08 19:22:26', '111996782304424104647', '2025-07-11 21:30:00', '2025-
07-13 21:00:00', NULL, 'active', 4400, '2025-07-08 21:52:34', 'credit',
'1N4A11APXJC123899', 'TCNT', 'TCNT', 'Y25S003',NULL,NULL),
('2025-06-15 11:44:58', '102345678901234567890', '2025-06-23 09:30:00', '2025-
06-27 18:00:00', '2025-06-27 17:55:00', 'completed', 10000, '2025-06-23 09:20:00',
'cash', 'KM8K6CA43LU123456', 'TCWF', 'CHIC', 'Y25S001','2025-06-23
09:32:00','CHIC'),
('2025-06-01 19:45:39', '213456789012345678901', '2025-07-12 07:30:00', '2025-
07-13 20:00:00', NULL, 'cancelled', 5200, NULL, 'cash', 'KM8K6CA43LU123456',
'TCWF', 'TCWF', 'Y25S004',NULL,NULL),
('2025-06-01 12:22:28', '324567890123456789012', '2025-06-02 06:03:00', '2025-
06-04 19:05:00', NULL, 'cancelled', 7800, NULL, 'cash', '4T3HV36KXPX456781',
'KHLG', 'YUNH', 'Y25S003',NULL,NULL),
('2025-06-03 09:25:35', '435678901234567890123', '2025-06-07 20:00:00', '2025-
06-10 19:05:00', '2025-06-10 19:08:00', 'completed', 10800, '2025-06-07 19:45:30',
'cash', '4T1BE46KX7U123456', 'YUNH', 'YUNL', 'Y25Z001','2025-06-07
20:00:00','YUNL'),

```

```
('2025-06-12 08:11:43', '213456789012345678901', '2025-06-19 05:00:00', '2025-06-19 23:00:00', NULL, 'active', 3800, NULL, 'cash', '5FNRL38647B654321', 'YUNH', 'YUNH', 'Y25X003',NULL,NULL),
('2025-06-02 13:13:33', '213456789012345678901', '2025-07-15 07:03:00', '2025-07-18 16:06:00', NULL, 'active', 18800, NULL, 'cash', '5FNRL38647B654321', 'YUNH', 'TCNT', 'Y25Z002',NULL,NULL);
```

DCL / VIEW (External Level)

```
USE car_rental_db;

CREATE VIEW GU_member_google_view AS
SELECT google_id FROM member;
GRANT SELECT ON car_rental_db.GU_member_google_view TO
'guest_user'@'localhost';
CREATE VIEW GU_cars_view AS
SELECT
    c.vin,
    m.brand,
    m.model_name,
    m.car_type,
    m.fuel_type,
    m.engine_cc,
    m.transmission,
    m.image_url,
    c.daily_fee,
    c.late_fee,
    c.seat_num,
    c.notes,
    l.loc_name
FROM car c
JOIN model m ON c.model_id = m.model_id
JOIN location l ON c.loc_id = l.loc_id
WHERE c.car_status = 'available';
GRANT SELECT ON car_rental_db.GU_cars_view TO 'guest_user'@'localhost';
CREATE VIEW GU_rental_schedule_view AS
SELECT
    r.car_vin as vin,
    r.start_date,
```

```

r.end_date
FROM rental r
JOIN car c ON r.car_vin = c.vin
WHERE r.rental_status IN ('pending', 'active')
    AND r.end_date > NOW()
ORDER BY r.car_vin, r.start_date;
GRANT SELECT ON car_rental_db.GU_rental_schedule_view TO
'guest_user'@'localhost';
CREATE VIEW GU_car_with_rented_time_view AS
SELECT
    c.vin,
    c.brand,
    c.seat_num,
    c.model_name,
    c.daily_fee,
    c.car_type,
    c.fuel_type,
    c.loc_name,
    engine_cc,
    transmission,
    image_url,
    t.start_date,
    t.end_date
FROM GU_cars_view as c
LEFT JOIN GU_rental_schedule_view t ON c.vin = t.vin;
GRANT SELECT ON car_rental_db.GU_car_with_rented_time_view TO
'guest_user'@'localhost';
CREATE VIEW GU_top5_models_view AS
SELECT
    m.brand,
    m.model_name,
    m.fuel_type,
    COUNT(*) AS rental_count
FROM rental r
JOIN car c ON r.car_vin = c.vin
JOIN model m ON c.model_id = m.model_id
GROUP BY m.model_id
ORDER BY rental_count DESC

```

```

LIMIT 5;

GRANT SELECT ON car_rental_db.GU_top5_models_view TO
'guest_user'@'localhost';

CREATE VIEW view_brand_car_count AS
SELECT m.brand, COUNT(*) AS car_count
FROM car c
JOIN model m ON c.model_id = m.model_id
GROUP BY m.brand;

GRANT SELECT ON car_rental_db.view_brand_car_count TO
'guest_user'@'localhost';

GRANT INSERT ON car_rental_db.member TO 'guest_user'@'localhost';
GRANT SELECT ON car_rental_db.location TO 'guest_user'@'localhost';
GRANT SELECT ON car_rental_db.insurance TO 'guest_user'@'localhost';

CREATE VIEW CM_member_phone_VIEW AS
SELECT google_id, phone FROM member;
GRANT SELECT,UPDATE ON car_rental_db.CM_member_phone_VIEW TO
'customer_user'@'localhost';

CREATE VIEW CM_rental_basic_view AS
SELECT
    r.google_id,
    r.start_date,
    r.end_date,
    r.rental_status,
    r.amount,
    c.vin
FROM rental r
JOIN car c ON r.car_vin = c.vin;
GRANT SELECT ON car_rental_db.CM_rental_full_view TO
'customer_user'@'localhost';

CREATE VIEW CM_rental_full_view AS
SELECT
    r.google_id,
    r.created_at,
    r.start_date,
    r.end_date,
    r.rental_status,
    r.amount,
    r.method,
    r.duration
FROM rental r
JOIN car c ON r.car_vin = c.vin;

```

```

r.actual_drop_loc,
r.actual_pickup,
m.brand,
m.model_name,
m.car_type,
m.fuel_type,
m.engine_cc,
m.transmission,
m.image_url,
c.daily_fee,
c.seat_num

FROM rental r
JOIN car c ON r.car_vin = c.vin
JOIN model m ON c.model_id = m.model_id;
GRANT SELECT ON car_rental_db.CM_rental_full_view TO
'customer_user'@'localhost';
CREATE VIEW CM_rental_updatable_view AS
SELECT
    google_id,
    created_at,
    rental_status,
    insurance_id
FROM rental;
GRANT UPDATE ON car_rental_db.CM_rental_updatable_view TO
'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.location TO 'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.insurance TO 'customer_user'@'localhost';
GRANT INSERT ON car_rental_db.rental TO 'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.GU_cars_view TO 'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.GU_rental_schedule_view TO
'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.GU_car_with_rented_time_view TO
'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.view_brand_car_count TO
'customer_user'@'localhost';
GRANT SELECT ON car_rental_db.GU_top5_models_view TO
'customer_user'@'localhost';
CREATE VIEW AD_rental_summary_view AS

```

```

SELECT
    r.google_id,
    r.created_at,
    r.start_date,
    r.end_date,
    r.rental_status,
    r.amount,
    r.car_vin
FROM rental r;
GRANT SELECT ON car_rental_db.AD_rental_summary_view TO
'staff_user'@'localhost';
CREATE VIEW AD_rental_status_count_view AS
SELECT
    CASE
        WHEN r.rental_status IN ('canceled', 'rejected') THEN 'canceled'
        WHEN r.rental_status = 'active' AND NOW() BETWEEN r.start_date AND
r.end_date THEN 'active_onging'
        WHEN r.rental_status = 'active' AND NOW() < r.start_date THEN
'active_waiting'
        WHEN r.rental_status = 'completed' THEN 'completed'
        ELSE 'other'
    END AS rental_status_group,
    COUNT(*) AS total_orders
FROM rental r
GROUP BY rental_status_group

UNION ALL

SELECT
    'TOTAL' AS rental_status_group,
    COUNT(*) AS total_orders
FROM rental;
GRANT SELECT ON car_rental_db.AD_rental_summary_view TO
'staff_user'@'localhost';
CREATE VIEW AD_rental_detail_view AS
SELECT
    r.google_id,
    r.created_at,
    r.start_date,

```

```

r.end_date,
r.actual_return,
r.rental_status,
r.amount,
r.method,
r.pickup_loc,
r.drop_loc,
r.insurance_id,
    c.vin,
    m.model_name,
    m.brand,
    c.daily_fee,
    c.late_fee
FROM rental r
JOIN car c ON r.car_vin = c.vin
JOIN model AS m ON c.model_id = m.model_id;
GRANT SELECT ON car_rental_db.AD_rental_detail_view TO
'staff_admin'@'localhost';
GRANT SELECT ON car_rental_db.member TO 'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.location TO
'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.insurance TO
'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.model TO
'staff_admin'@'localhost';
GRANT SELECT, INSERT, UPDATE, DELETE ON car_rental_db.car TO
'staff_admin'@'localhost';
GRANT SELECT, UPDATE ON car_rental_db.rental TO 'staff_admin'@'localhost';
-- -----
FLUSH PRIVILEGES;

```

Transaction 搭配 php 程式避免重複租車寫入

為了防止同一時間多位會員操作導致同一輛車時段被重複租借，可使用 SQL Transaction 搭配 PHP 程式邏輯來進行「查詢+鎖定+判斷+寫入」來處理。

1. 鎖定該車輛資料：使用 SELECT ... FOR UPDATE，避免其他交易同時讀寫。
2. 由 PHP 判斷車況：如果狀態不是'available'，則 Rollback。
3. 查詢是否有租借重疊時段，並鎖定符合條件的 record。
4. 若有重疊則由 PHP 判斷並 ROLLBACK，否則進行資料寫入。
5. 成功後 COMMIT：資料正式寫入，並解鎖。

假設 vin 是'4T1BE46KX7U123456' 租用時間為 '2025-06-01' 至 '2025-06-05'

```
START TRANSACTION;

-- 鎖住指定車輛
SELECT car_status FROM car
WHERE vin = '4T1BE46KX7U123456'
FOR UPDATE;

-- PHP: 程式取出結果並判斷狀態是否為 'available'，否則 rollback

-- 檢查重疊時段
SELECT 1 FROM rental
WHERE car_vin = '4T1BE46KX7U123456'
    AND rental_status IN ('pending', 'active')
    AND (
        '2025-06-01' <= DATE_ADD(end_date, INTERVAL 1 DAY)
        AND '2025-06-05' >= DATE_SUB(start_date, INTERVAL 1 DAY)
    )
FOR UPDATE;

-- PHP: 檢查上一步是否有資料，若有資料則表示有重疊時段，則 Rollback。

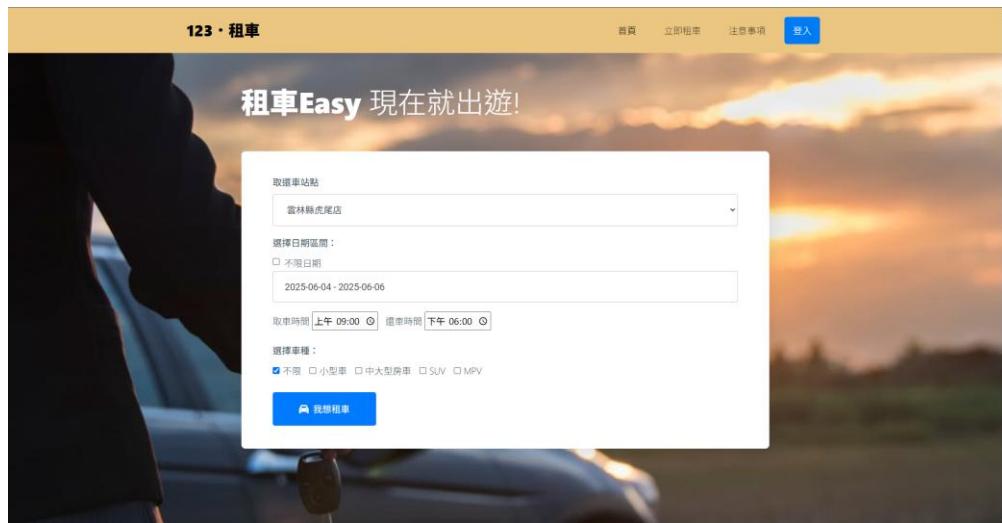
-- 寫入 rental 資料
INSERT INTO rental (created_at, google_id, start_date, end_date, actual_return,
rental_status, amount, payment_date, method, car_vin,
pickup_loc, drop_loc, insurance_id, actual_pickup, actual_drop_loc) VALUES
-- (' ', ' ', ...)

COMMIT;
```

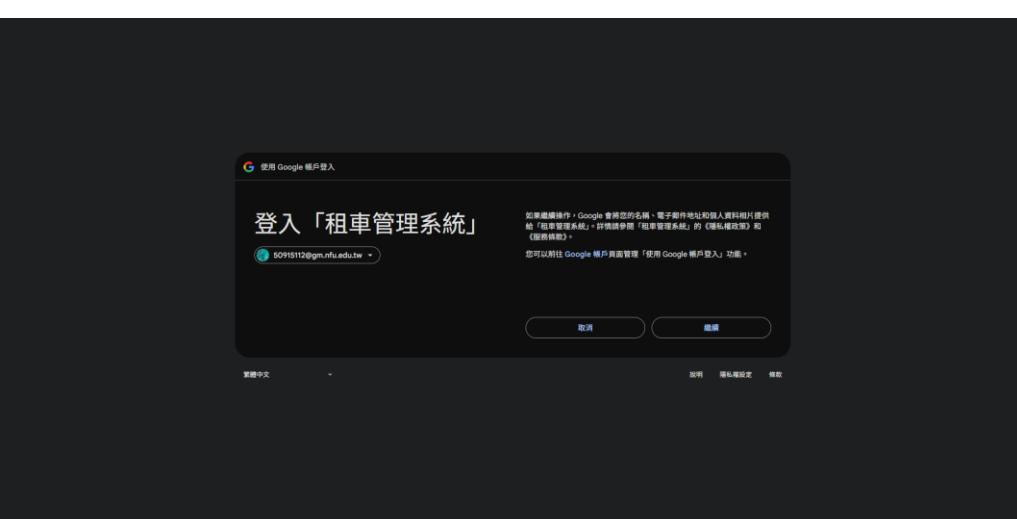
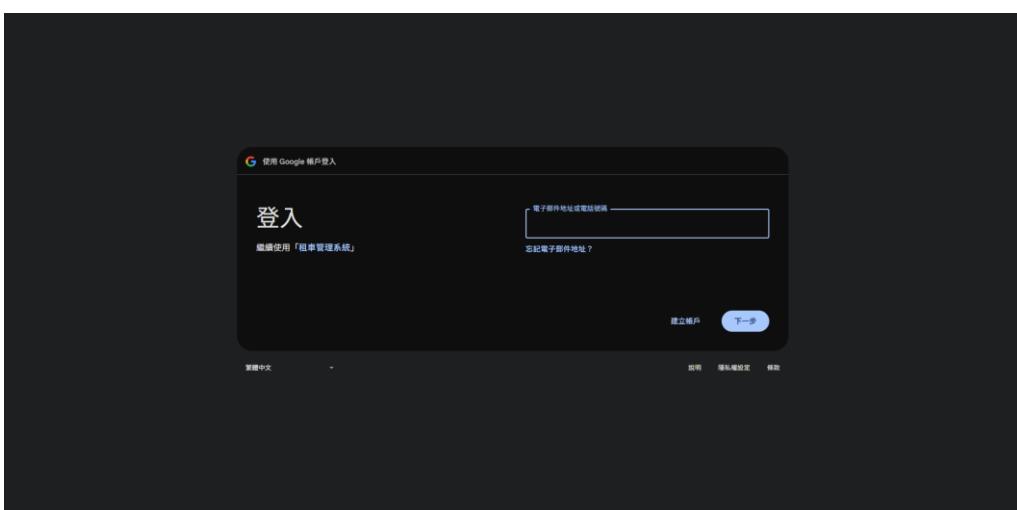
捌、系統結果

1. 會員 Google 驗證功能

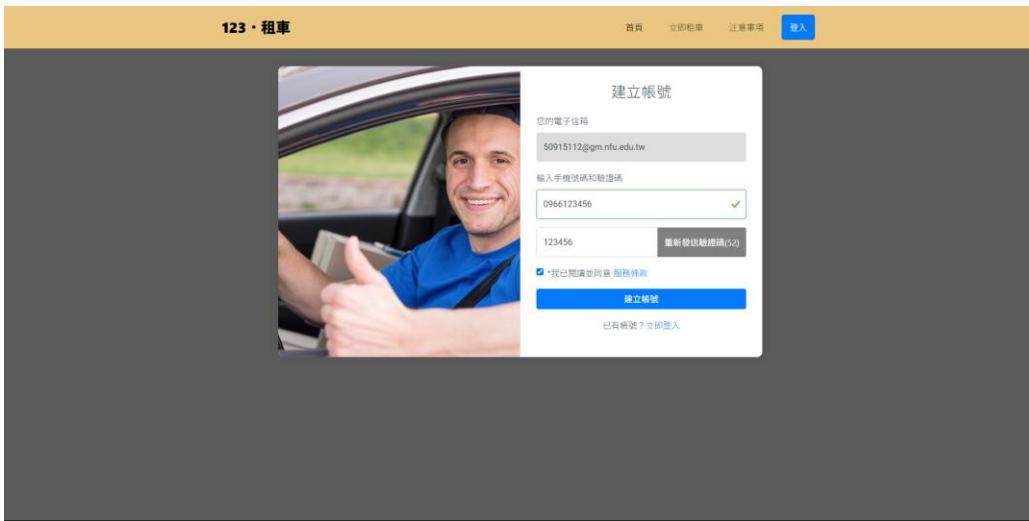
首頁介面



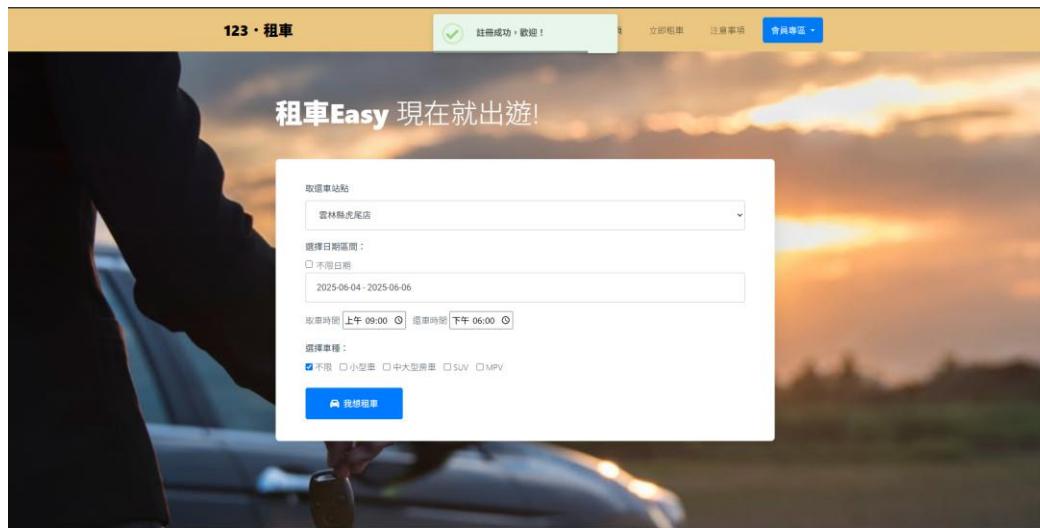
點擊登入後會導到 Google 登入介面



若初次登入，會導入到註冊介面，屆時需要輸入手機號碼、驗證碼。

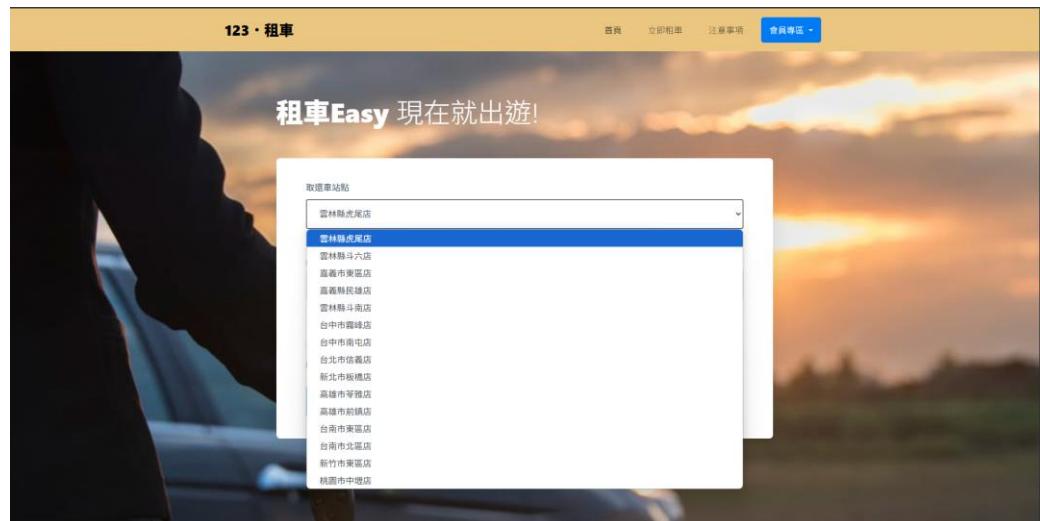


屆時建立註冊後，將會在資料庫新增一筆會員資料。

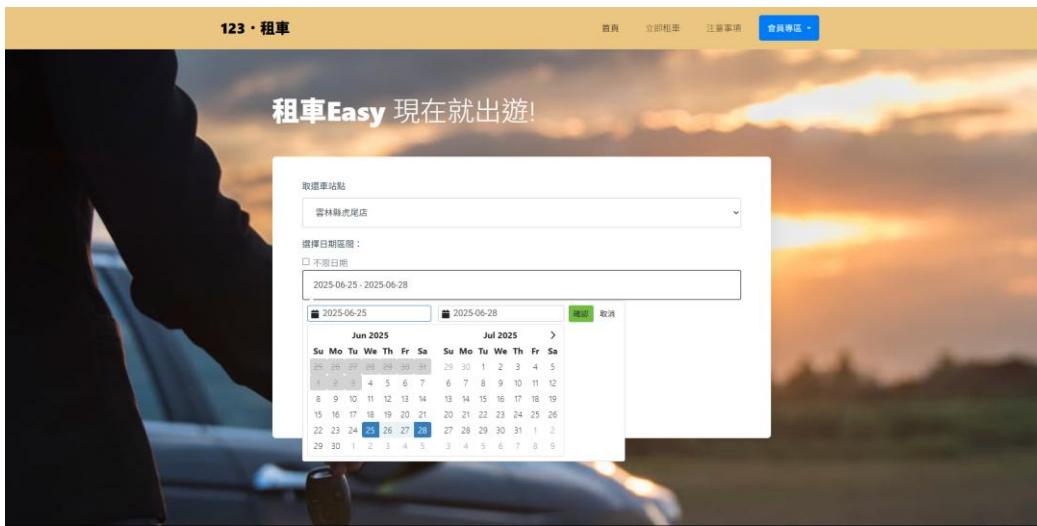


2. 會員租車功能

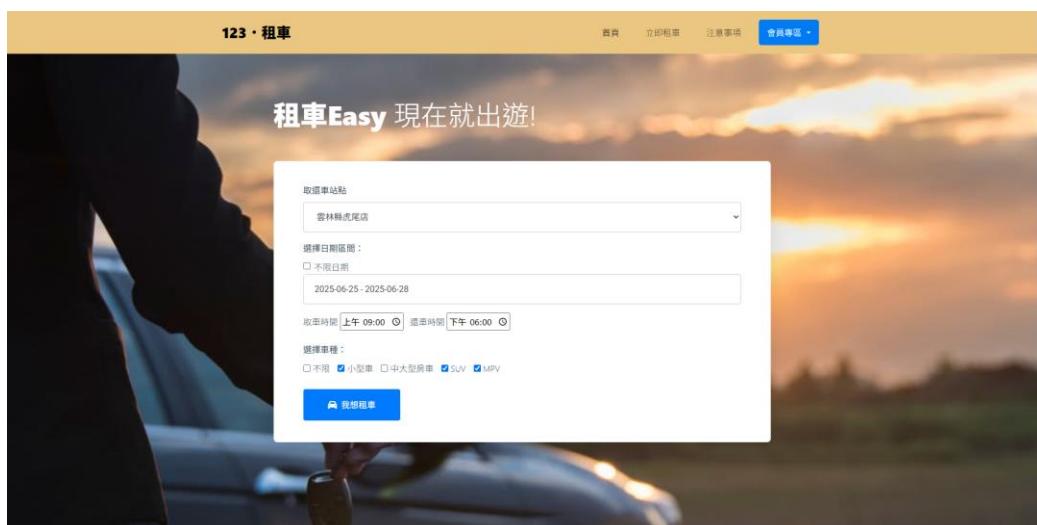
於首頁可以初步篩選租用條件
(地點)



(租用日期、時間區間) 本次選擇 2025/06/25 09:00 ~ 2025/06/28 18:00



(車種) 本次選擇小型車、SUV、MPV



選擇"我想租車"後，會進入進階查詢/結果介面

The screenshot shows the '筛选条件' (Filter Conditions) section of the website. It includes fields for pick-up location (Yunlin County Houtui Store), rental date range (from June 25, 2025, to June 28, 2025), and time (from 09:00 to 16:00). Filters for vehicle type (Small Car, SUV, MPV), fuel type (Unleaded, Gasoline, Electric, Hybrid), and passenger capacity (Unlimited) are also present. Below this, a daily rental price range slider is set from 0 to 6000. A blue button labeled '查詢可租車輛' (Search Rentable Vehicles) is at the bottom.

Below the filter section, there are three vehicle categories:

- TOYOTA Altis 12 / 1.8L**: White, NTD 1800 / day, 4 seats, Gasoline. Buttons: 租用 / 查詢可租時間段 (Rent / Query Rentable Time Period).
- MITSUBISHI Grand Lancer / 1.5L**: White, NTD 1700 / day, 4 seats, Gasoline. Buttons: 租用 / 查詢可租時間段 (Rent / Query Rentable Time Period).
- SUV**: A white Toyota RAV4 2022 / 2.5L. Buttons: 租用 / 查詢可租時間段 (Rent / Query Rentable Time Period).

At the bottom, there is another category labeled **MPV**.

接著我們可以選擇這些時段可租用的車輛。本次選擇 Grand Lancer。
(再次確認/選擇租用日期)

A modal dialog box titled '預約 - MITSUBISHI Grand Lancer' is displayed. It shows a date picker for '租車日期' (Rental Date) set from June 25, 2025, to June 28, 2025. The calendar grid highlights these dates. A green '下一步' (Next Step) button is visible at the bottom right of the dialog.

Below the dialog, the main page shows the '可租用車輛' (Rentable Vehicles) section for '小型車' (Small Car). It displays images of the TOYOTA Altis 12 / 1.8L and the MITSUBISHI Grand Lancer.

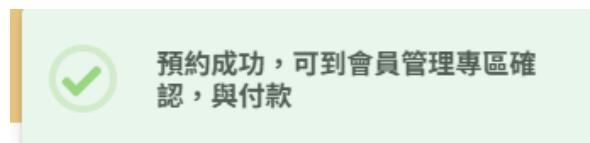
選擇保險方案



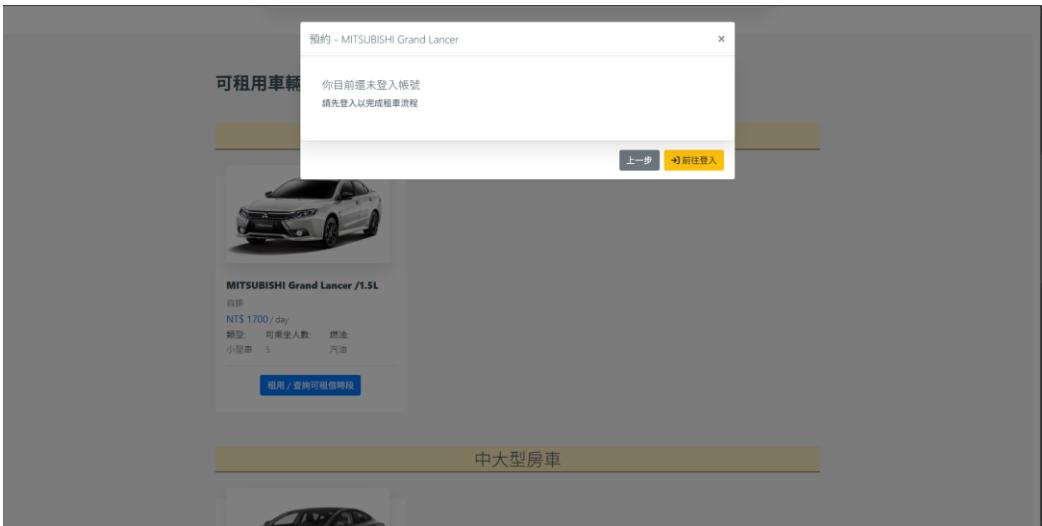
最後確認預約資訊，4 天×每日租金(車輛費 1700 + 保險費 400) = NT\$ 8400 元
共 8400 元，送出預約。



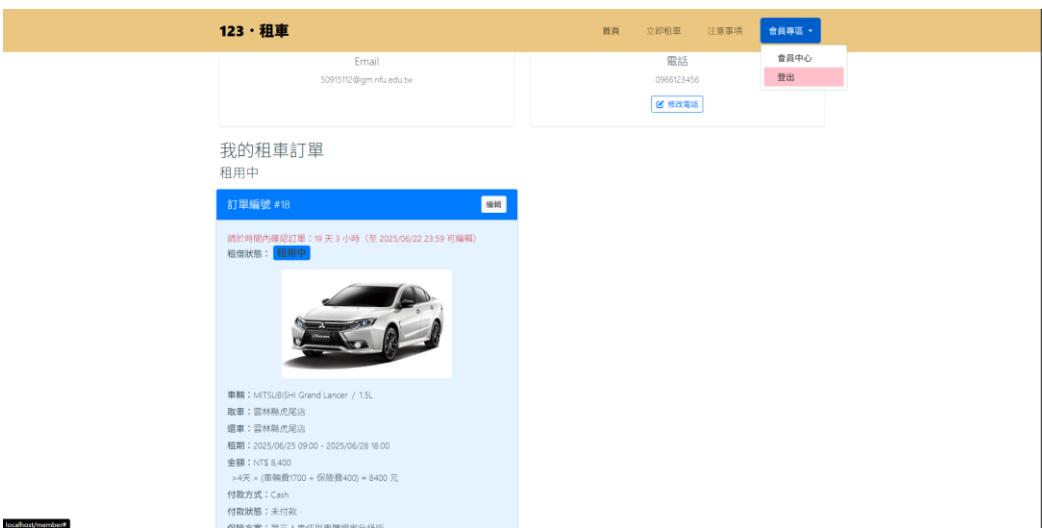
顯示預約成功



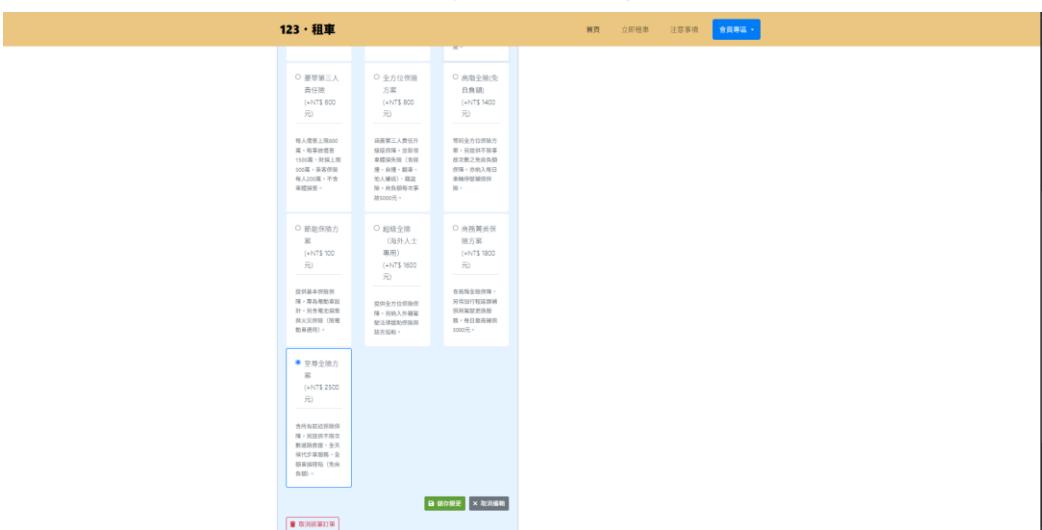
在此補充，若是未登入帳戶，在預約時，會顯示需前往登入。



到會員中心，查看歷史租車紀錄(租用中、已完成、已取消、被拒絕)
每個訂單於期限內，可以修改訂單內容(保險方案)或是取消預約。



於期限內更新保險方案 本次選擇修改為"至尊全險方案"



保險方案更新成功

我的租車訂單
租用中

訂單編號 #18
請於時間內確認訂單：1天 2 小時（至 2025/06/22 23:59 可編輯）
租借狀態：已修改

車輛：MITSUBISHI Grand Lancer / 1.5L
取車：雲林縣虎尾店
還車：雲林縣虎尾店
租期：2025/06/25 09:00 - 2025/06/28 18:00
金額：NT\$ 16,800
$$4 \times (\text{車輛費} 1700 + \text{保險費} 2500) = 16800 \text{ 元}$$

付款方式：Cash
付款狀態：未付款
保險方案：至尊全險方案
查看保險內容
含所有前述保險保障，另提供不同次數救援救援、全天候代步車服務、全額車輛保固（免自負額）。

等待審核

於期限內可以取消訂單

123 · 租車

取消訂單 (編號#18)

◎ 基本保險方案 (+NT\$ 100 元)
◎ 超級全險方案 (海外人壽專用) (+NT\$ 180 元)

提供全方位保險保障，另納入外籍車輛法律援助與賠償責任擔保。

● 至尊全險方案 (+NT\$ 2500 元)
含所有前述保險保障，另提供不限次救援救援、全天候代步車服務、全額車輛保固（免自負額）。

■ 取消此訂單 儲存編號 X 取消編號

確定要取消此訂單嗎？

取消後的行程將無法恢復，每日最高補償金額 3000 元。

關閉 確定取消訂單

等待審核

目前沒有等待中的訂單。

已完成

取消租車預約 成功，資料庫會將該筆訂單狀態更改為已取消。

訂付查詢

訂單取消成功

已完成

目前沒有已完成的訂單。

已取消

訂單編號 #18
租借狀態：已取消

車輛：MITSUBISHI Grand Lancer / 1.5L
取車：雲林縣虎尾店
還車：雲林縣虎尾店
租期：2025/06/25 09:00 - 2025/06/28 18:00
金額：NT\$ 16,800
$$4 \times (\text{車輛費} 1700 + \text{保險費} 2500) = 16800 \text{ 元}$$

付款方式：Cash
付款狀態：未付款
保險方案：至尊全險方案
查看保險內容

接著我們重新預約一筆租車 TOYOTA Altis 12 日期為 2025/06/19 ~ 2025/06/21

123 · 租車

預約 - TOYOTA Altis 12

TOYOTA Altis 12

自取
NT\$ 1800 / day

類型：可乘坐人
小型車 5

租用 / 查詢可租借時段

預約總覽確認

車型：TOYOTA Altis 12 (小型車 - 5人座) 燃油方式:油電混合

會員Email : 5095102@gm.nfue.edu.tw

會員電話 : 0966123456

取車日期時間 : 2025-06-19 08:00

還車日期時間 : 2025-06-21 17:00

保險方案：全方位保險方案 (+NT\$ 800 元/每日)

送客第三責任升級保全，並新增車體損失險（含碰撞、自撞、翻車、他人肇造）、商賈險。自負額每次事故3000元。

總租金：3天 × 每日租金(車輛費 1800 + 保險費 800) = NT\$ 7800元

若逾時未歸還車輛：每小時收取NT\$ 200元

上一步 送出預約

123 · 租車

我的租車訂單

租用中

訂單編號 #99

請於時間內確認訂單：3 天 2 小時（至 2025/06/16 23:59 可編輯）

租借狀態：預約中

車輛：TOYOTA Altis 12 / 1.8L

取車：雲林縣虎尾店

還車：雲林縣虎尾店

租期：2025/06/19 08:00 - 2025/06/21 17:00

金額：NT\$ 7,800

>3天 × (車輛費1800 + 保險費800) = 7800 元

付款方式：Cash

付款狀態：未付款

保險方案：全方位保險方案

查看保險內容

等待審核

若其他人剛好也想預約同台車但有重疊時段，會顯示該日期不可預約。

可租用車輛

預約 - TOYOTA Altis 12

日期包含不可預約的日期！

租車日期：

2025-06-17 - 2025-06-23

2025-06-17

2025-06-18

2025-06-19

2025-06-20

2025-06-21

2025-06-22

2025-06-23

2025-06-24

2025-06-25

2025-06-26

2025-06-27

2025-06-28

2025-06-29

2025-06-30

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

2025-07-03

2025-07-04

2025-07-05

2025-07-06

2025-07-07

2025-07-08

2025-07-09

2025-07-10

2025-07-11

2025-07-12

2025-07-13

2025-07-14

2025-07-15

2025-07-16

2025-07-17

2025-07-18

2025-07-19

2025-07-20

2025-07-21

2025-07-22

2025-07-23

2025-07-24

2025-07-25

2025-07-26

2025-07-27

2025-07-28

2025-07-29

2025-07-30

2025-07-31

2025-07-01

2025-07-02

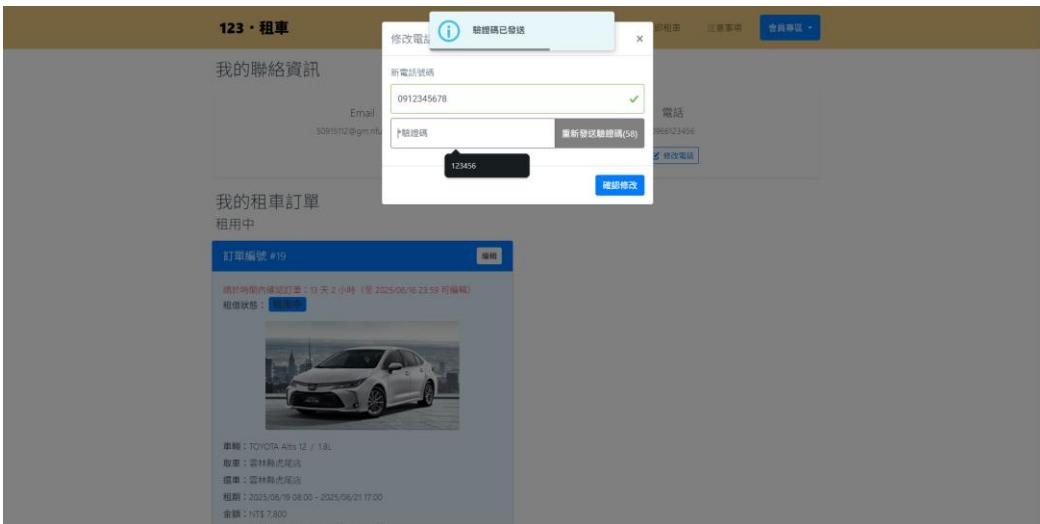
2025-07-03

2025-07-04

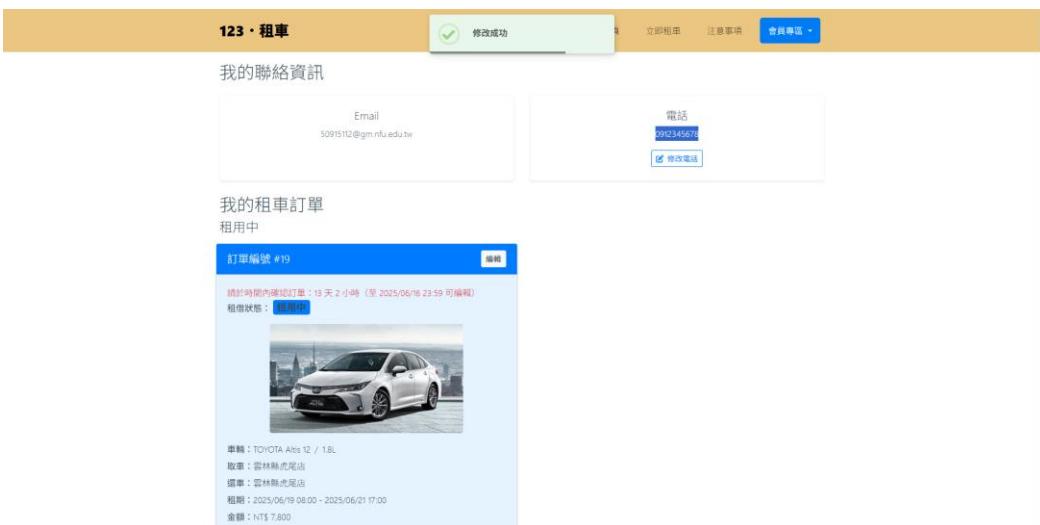
2025-07-05

2025

而外補充，可以在會員中心進行電話號碼的更改。



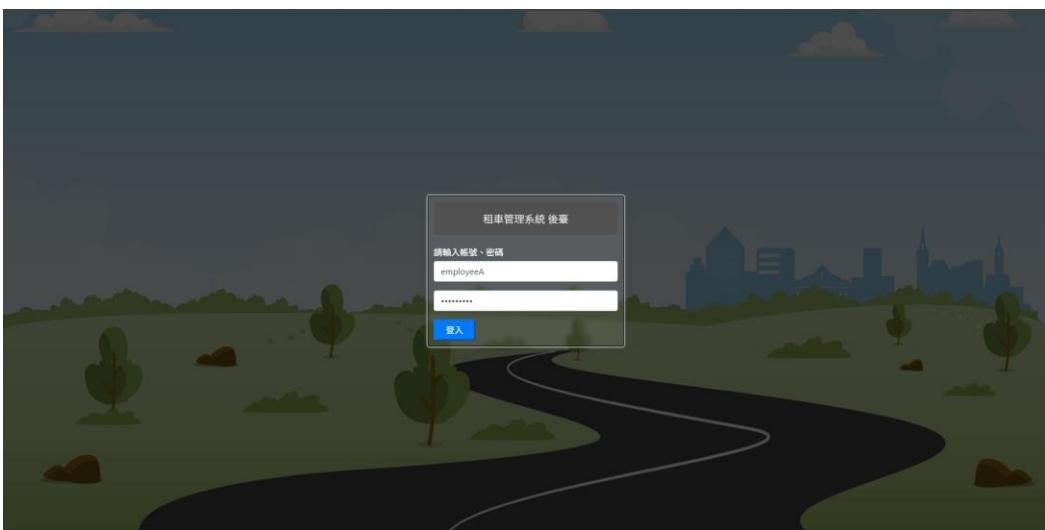
電話號碼更新成功 (更新為 0912345678)



3. 管理員功能

驗證系統使用 Python JWT 做模擬

管理員登入介面



首頁畫面

針對車型圖片做管理(新增、刪除圖片)

可針對車輛、車型做管理(新增、刪除、修改)

車型編號	品牌	車型名稱	車型類別	排氣量	引擎排氣量 (CC)	驅動方式	圖片	距離	狀態	操作
1	TOYOTA	Altis 12	Compact	Hybrid	1800 cc	自排		15000 公里	available	<button>編輯車型</button> <button>刪除車型</button> <button>新增汽車</button>
2	RAA-7777	1800	日規車	200 元	2020	5	白色	16500 公里	available	<button>編輯</button> <button>刪除</button>
2	HONDA	Odyssey 2019	MPV	Gasoline	2500 cc	自排		12000 公里	available	<button>編輯車型</button> <button>刪除車型</button> <button>新增汽車</button>
3	RAR-2385	2200	日規車	200 元	2020	7	黑色	12000 公里	available	<button>編輯</button> <button>刪除</button>
3	TESLA	Model 3	Sedan	Electric	0 cc	自排		12000 公里	available	<button>編輯車型</button> <button>刪除車型</button> <button>新增汽車</button>

車型新增介面

新增車型

品牌: TOYOTA

車型名稱: Altis 12

車型類別: Compact

排氣量: Gasoline

引擎排氣量 (CC): 1800

駕駛座: 自排

圖片 URL: img/Altis12.jpg

新增 取消

車型編輯介面

編輯車型

品牌: TOYOTA

車型名稱: Altis 12

車型類別: Compact

排氣量: Hybrid

引擎排氣量 (CC): 1800

駕駛座: 自排

圖片 URL: img/Altis12.jpg

儲存變更 取消

車輛新增介面

The screenshot shows the '新增汽車' (Add Car) interface. It includes fields for basic car information (brand, model, year, license plate), usage fees (daily rate, monthly rate, seat capacity, color), and maintenance costs (daily rate, monthly rate). On the right, there are three preview cards showing different car models (Toyota Altis, Honda Odyssey, Tesla Model 3) with their respective details and operation buttons.

車輛編輯介面

The screenshot shows the '編輯汽車' (Edit Car) interface. It displays the current car details (RAA-7777, Toyota Altis, 2020, etc.) and allows editing of these fields. The right side shows preview cards for the same three car models as the addition interface.

車輛刪除功能(刪除 Car ID:2)

The screenshot shows the '車型與車輛管理' (Car Type and Vehicle Management) interface. A green success message '汽車刪除成功' (Car deleted successfully) is displayed. The main table lists three cars, with the second row (HONDA Odyssey 2019) being the one that was deleted. The table includes columns for car details, usage fees, and operation buttons.

若車輛 PK 有被訂單關聯，則無法被刪除(DELETE Car 1: NO ACTION)。

車輛編號	車牌號碼	日租金	近期費用	製造年份	座位數	顏色	里程數	狀態	操作
1	RAA-7777	1800 元	200 元	2020	5	白色	15000 公里	available	<button>編輯</button> <button>刪除</button>
3	REK-3556	2500 元	300 元	2021	5	銀色	8000 公里	available	<button>編輯</button> <button>刪除</button>

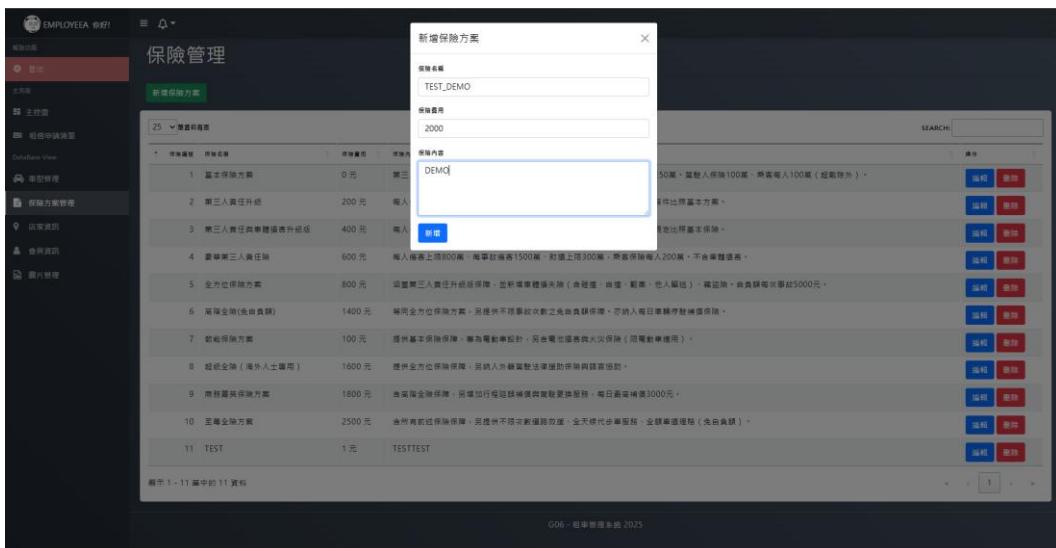
若車型 PK 有被車輛關聯，也無法被刪除(DELETE Model 1: NO ACTION)。

車型編號	車牌號碼	日租金	近期費用	製造年份	座位數	顏色	里程數	狀態	操作
1	RAA-7777	1800 元	200 元	2020	5	白色	15000 公里	available	<button>編輯</button> <button>刪除</button>
2	HONDA	Odyssey 2019	MPV	Gasoline	2500 cc	自排			<button>編輯車型</button> <button>刪除車型</button> <button>新增汽車</button>
3	RAR-2385	2200 元	200 元	2020	7	黑色	12000 公里	available	<button>編輯</button> <button>刪除</button>
3	REK-3556	2500 元	300 元	2021	5	銀色	8000 公里	available	<button>編輯</button> <button>刪除</button>

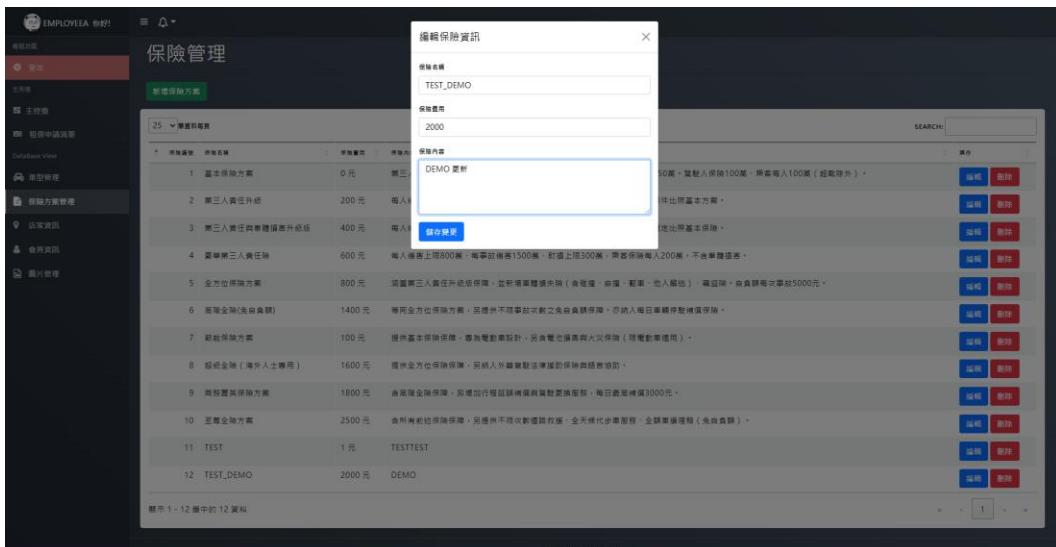
針對保險方案做管理(新增、修改、刪除)

序號	保費類別	保險費用	保險內容	操作
1	基本保費方案	0 元	第三人責任保險：每人傷害上限200萬、每車財產上限400萬、財損上限50萬。業主保險100萬、乘客每人100萬（超載除外）。	<button>編輯</button> <button>刪除</button>
2	第三人責任升級	200 元	每人傷害上限提升至500萬、每車財產1000萬、財損上限200萬。其餘條件比照基本方案。	<button>編輯</button> <button>刪除</button>
3	第三人責任升級搭配升級	400 元	每人傷害上限提升至500萬、每車財產1000萬、財損上限200萬。其餘條件比照基本方案。	<button>編輯</button> <button>刪除</button>
4	豪華第三人責任險	600 元	每人傷害上限800萬、每車財產最高1500萬、財損上限300萬。業主保險每人200萬。不含車體保養。	<button>編輯</button> <button>刪除</button>
5	全方位保費方案	800 元	豪華第三人責任升級保費 + 豪華車體保養保費 (含碰撞、自燃、刮擦、凹陷修復)、業主保險每車5000元。	<button>編輯</button> <button>刪除</button>
6	豪華全險(免白費)	1400 元	臺灣全方位保費方案，另包含不必要事故計劃之免白費保費，即燒、自燃、刮擦、凹陷修復保費。	<button>編輯</button> <button>刪除</button>
7	船底保費方案	100 元	提供全方位保費保底，適用於電動車底盤，遇水電池損害與火災保底（荷電動車底盤）。	<button>編輯</button> <button>刪除</button>
8	超級全險 (海外人士專用)	1600 元	提供全方位保費保底，另納入外籍駕駛法律援助保費保底。	<button>編輯</button> <button>刪除</button>
9	商務車險保費方案	1800 元	商業車全險保費，另進行程記錄保費與駕駛更換保費。每日商業保費3000元。	<button>編輯</button> <button>刪除</button>
10	至尊全險方案	2500 元	臺灣有前店保費保底，另提供不限次數道路救援、全天候代步車服務、全球救援服務（免白費保底）。	<button>編輯</button> <button>刪除</button>
11	TEST	1 元	TESTTEST	<button>編輯</button> <button>刪除</button>

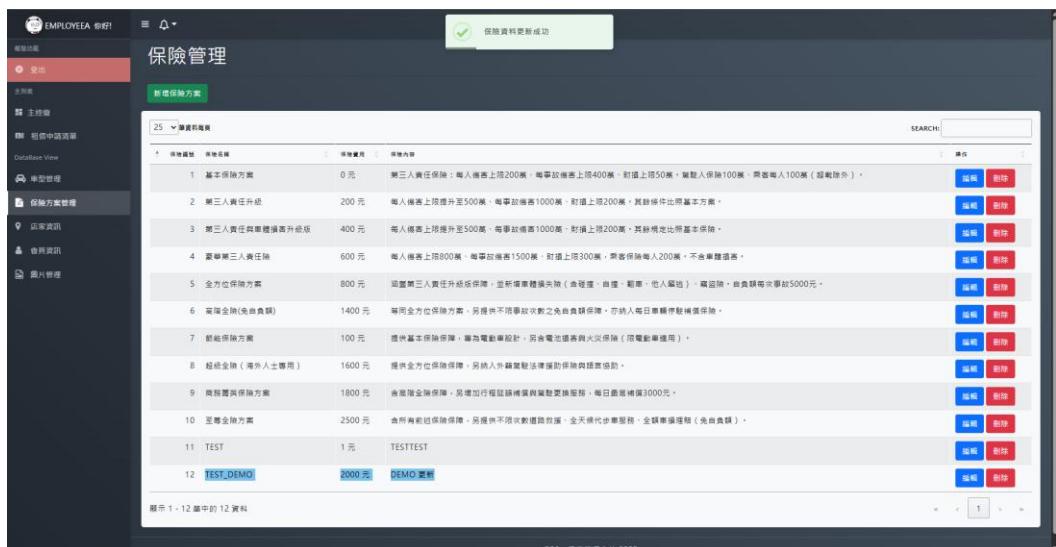
保險方案新增介面



保險方案編輯介面



保險方案編輯成功



保險方案刪除

The screenshot shows a list of 11 insurance policies. Each row contains policy details such as name, premium, and coverage. Buttons for '編輯' (Edit) and '刪除' (Delete) are on the right. A green success message at the top right indicates a successful deletion.

序號	保險名稱	保險費用	保險內容	操作
1	基本保險方案	0 元	第三人責任保險：每人傷害上限200萬、每單事故傷害400萬、財產上限50萬、駕駛人保險100萬、乘客每人100萬（超載除外）。	<button>編輯</button> <button>刪除</button>
2	第三人責任升級	200 元	每人傷害上限提升至500萬、每單事故傷害1000萬、財產上限200萬、其餘規定比照基本方案。	<button>編輯</button> <button>刪除</button>
3	第三人責任與車體保養升級	400 元	每人傷害上限提升至500萬、每單事故傷害1000萬、財產上限200萬、其餘規定比照基本方案。	<button>編輯</button> <button>刪除</button>
4	豪華第三人責任險	600 元	每人傷害上限800萬、每單事故傷害1500萬、財產上限300萬、兩者保險每人200萬、不含車體損害。	<button>編輯</button> <button>刪除</button>
5	全方位保險方案	800 元	頂級第三人責任升級保全，並新增車體損壞保險（含碰撞、盜竊、車輛、他人駕駛）、高級版，每單事故每次事故5000元。	<button>編輯</button> <button>刪除</button>
6	豪華全險(全險貴族)	1400 元	獨創全方位保險方案，另增加不禮讓行車數次數之多向負擔保障，亦納入每日最高總額財產保險。	<button>編輯</button> <button>刪除</button>
7	駕駛保險方案	100 元	提供基本駕駛保險，專為電動車駕駛設計，並含電池損害與人災保險（防電動車進用）。	<button>編輯</button> <button>刪除</button>
8	經紀全險 (海外人士適用)	1600 元	提供全方位保險保障，另納入外籍駕駛法律援助服務與緊急回國諮詢。	<button>編輯</button> <button>刪除</button>
9	商務賑災保險方案	1800 元	商業賑全險保障，另導入行程記錄保險與駕駛員換證服務，每日最高總額3000元。	<button>編輯</button> <button>刪除</button>
10	至尊全險方案	2500 元	含所有前述保險保障，另提供不禮讓行車路旁證、全天時代步車服務、全球車道理賠（免由貴族）。	<button>編輯</button> <button>刪除</button>
11	TEST	1 元	TESTTEST	<button>編輯</button> <button>刪除</button>

也能針對店家(Location)進行新增、修改、刪除

The screenshot shows a list of 10 locations. Each row contains location details such as name, city, district, and address. Buttons for '編輯' (Edit) and '刪除' (Delete) are on the right. A green success message at the top right indicates a successful addition.

序號	店家名稱	城市	區域	地址	操作
1	雲林縣虎尾店	雲林縣	虎尾鎮	文化路64號	<button>編輯</button> <button>刪除</button>
2	雲林縣斗六店	雲林縣	斗六市	大學路三段123號	<button>編輯</button> <button>刪除</button>
3	嘉義市東區店	嘉義市	東區	學府路300號	<button>編輯</button> <button>刪除</button>
4	嘉義新民店	嘉義縣	民雄鄉	神農路145號	<button>編輯</button> <button>刪除</button>
5	雲林縣斗南店	雲林縣	斗南鎮	大業路106號	<button>編輯</button> <button>刪除</button>
6	台中市豐原店	台中市	豐原區	吉峰路168號	<button>編輯</button> <button>刪除</button>
7	台中市南屯店	台中市	南屯區	大雅路200號	<button>編輯</button> <button>刪除</button>
8	台北市信義店	台北市	信義區	松山街100號	<button>編輯</button> <button>刪除</button>
9	新北市板橋店	新北市	板橋區	文化路二段50號	<button>編輯</button> <button>刪除</button>
10	高雄市苓雅店	高雄市	苓雅區	和平一路300號	<button>編輯</button> <button>刪除</button>

提供會員列表總覽

The screenshot shows a detailed list of 22 members. Each row contains member information such as ID, email, and phone number. Buttons for '編輯' (Edit) and '刪除' (Delete) are on the right. A red button at the top right indicates a new member can be added.

會員編號	電子郵件	電話	
1	879012345678901234567	quincy_ju@gmail.com	0978999000
2	657890123456789012345	kulpia@gmail.com	0978985123
3	111996782304424104647	gene_st@gmail.com	0914238500
4	19270782947823801723	Wen_Che@gmail.com	0932535148
5	215784230947120938765	asd_tw@gmail.com	0931164655
6	34205739025847386423	kevin_liu_jr@gmail.com	0923969027
7	102345678901234567890	jane_liu@gmail.com	0999000111
8	213456789012345678901	kevin_liu@yahoo.com	0912334444
9	324567890123456789012	llisa_wang@nfu.edu.tw	0923444555
10	435678901234567890123	mike_smith@wistron.com	0934556666
11	546789012345678901234	nancy_johnson@foxconn.com	0945666777
12	567890123456789012345	ethan_liu@yahoo.com	0944555666
13	678901234567890123456	fiona_lee@nfu.edu.tw	0955666777
14	789012345678901234567	george.lin@gmail.com	0966777888
15	123456789012345678901	alice_wu@gmail.com	0987123456
21	106455916943881880515	a0978435327@gmail.com	0978435326
22	115281260363145187252	50915112@gn.nfu.edu.tw	0912345678

預約訂單查詢，可透過上方列狀態查詢

序號	電子郵件	車款	車牌	預取時間	預交時間	狀態	付款方式	金額
20	a0978435327@gmail.com	HONDA CRV-6	RAB-2588	2025-06-04 09:00:00	2025-06-06 18:00:00	●已完成	已付數	NT\$ 7200
19	50915112@gm.nfu.edu.tw	TOYOTA Altis 12	RAA-7777	2025-06-19 08:00:00	2025-06-21 17:00:00	●審過-未開始	未付款	NT\$ 7800
18	50915112@gm.nfu.edu.tw	MITSUBISHI Grand Lance	RAF-5678	2025-06-25 09:00:00	2025-06-28 18:00:00	●已取消	未付款	NT\$ 16800
16	a0978435327@gmail.com	TOYOTA RAV4 2022	RBC-8811	2025-06-14 08:00:00	2025-06-15 19:00:00	●已取消	未付款	NT\$ 5200
11	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-07-15 07:00:00	2025-07-18 16:00:00	●審過-未開始	未付款	NT\$ 18000
10	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-06-19 05:00:00	2025-06-19 23:00:00	●審過-未開始	未付款	NT\$ 3800
9	mike_smith@wistron.com	TOYOTA Altis 12	RAA-7777	2025-06-07 20:00:00	2025-06-10 19:05:00	●已完成	已付數	NT\$ 10800
8	lisa_wang@nfu.edu.tw	TOYOTA RAV4 2022	RBC-8811	2025-06-02 06:00:00	2025-06-04 19:05:00	●已取消	未付款	NT\$ 7800
7	kevin_liu@yahoo.com	HYUNDAI Tucson 2020	RBD-4567	2025-07-12 07:30:00	2025-07-13 20:00:00	●已取消	未付款	NT\$ 5200
6	jane_liu@gmail.com	HYUNDAI Tucson 2020	RBD-4567	2025-06-23 09:30:00	2025-06-27 18:00:00	●已完成	已付數	NT\$ 10000
5	gene_st@gmail.com	NISSAN Leaf 2021	RBD-6789	2025-07-11 21:30:00	2025-07-13 21:00:00	●審過-未開始	已付數	NT\$ 4400
4	Wen_Che@gmail.com	TESLA Model 3	REK-3556	2025-06-18 13:00:00	2025-06-20 00:40:00	●已完成	已付數	NT\$ 5200
3	kulpia@gmail.com	CMC J-SPACE	RAS-5678	2025-07-14 11:05:00	2025-07-17 23:45:00	●審過-未開始	未付款	NT\$ 12400
2	quincy_liu@gmail.com	TOYOTA Altis 12	RAA-7777	2025-06-13 08:00:00	2025-06-14 17:00:00	●已取消	未付款	NT\$ 6400

序號	電子郵件	車款	車牌	預取時間	預交時間	狀態	付款方式	金額
19	50915112@gm.nfu.edu.tw	TOYOTA Altis 12	RAA-7777	2025-06-19 08:00:00	2025-06-21 17:00:00	●審過-未開始	未付款	NT\$ 7800
11	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-07-15 07:00:00	2025-07-18 16:00:00	●審過-未開始	未付款	NT\$ 18000
10	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-06-19 05:00:00	2025-06-19 23:00:00	●審過-未開始	未付款	NT\$ 3800
9	mike_smith@wistron.com	TOYOTA Altis 12	RAA-7777	2025-06-07 20:00:00	2025-06-10 19:05:00	●已完成	已付數	NT\$ 10800
6	jane_liu@gmail.com	HYUNDAI Tucson 2020	RBD-4567	2025-06-23 09:30:00	2025-06-27 18:00:00	●已完成	已付數	NT\$ 10000
5	gene_st@gmail.com	NISSAN Leaf 2021	RBD-6789	2025-07-11 21:30:00	2025-07-13 21:00:00	●審過-未開始	已付數	NT\$ 4400
4	Wen_Che@gmail.com	TESLA Model 3	REK-3556	2025-06-18 13:00:00	2025-06-20 00:40:00	●已完成	已付數	NT\$ 5200
3	kulpia@gmail.com	CMC J-SPACE	RAS-5678	2025-07-14 11:05:00	2025-07-17 23:45:00	●審過-未開始	未付款	NT\$ 12400

管理員對訂單進行編輯介面

序號	電子郵件	車款	車牌	預取時間	預交時間	狀態	付款方式	金額
19	50915112@gm.nfu.edu.tw	TOYOTA Altis 12	RAA-7777	2025-06-19 08:00:00	2025-06-21 17:00:00	●審過-未開始	未付款	NT\$ 7800
11	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-07-15 07:00:00	2025-07-18 16:00:00	●審過-未開始	未付款	NT\$ 18000
10	kevin_liu@yahoo.com	HONDA Odyssey 2019	RAR-2385	2025-06-19 05:00:00	2025-06-19 23:00:00	●審過-未開始	未付款	NT\$ 3800
9	mike_smith@wistron.com	TOYOTA Altis 12	RAA-7777	2025-06-07 20:00:00	2025-06-10 19:05:00	●已完成	已付數	NT\$ 10800
6	jane_liu@gmail.com	HYUNDAI Tucson 2020	RBD-4567	2025-06-23 09:30:00	2025-06-27 18:00:00	●已完成	已付數	NT\$ 10000
5	gene_st@gmail.com	NISSAN Leaf 2021	RBD-6789	2025-07-11 21:30:00	2025-07-13 21:00:00	●審過-未開始	已付數	NT\$ 4400
4	Wen_Che@gmail.com	TESLA Model 3	REK-3556	2025-06-18 13:00:00	2025-06-20 00:40:00	●已完成	已付數	NT\$ 5200
3	kulpia@gmail.com	CMC J-SPACE	RAS-5678	2025-07-14 11:05:00	2025-07-17 23:45:00	●審過-未開始	未付款	NT\$ 12400

假設進行編輯(狀態:已完成、保險方案:至尊全險、付款狀態:已付款)

localhost 展示
確定要變更的以下資訊？

車款：TOYOTA Altis 1.2
車頭型：小頭車
頭燈方式：泡頭燈
CC數：1800
傳動力：自然進氣
可乘坐人數：5
車牌：AAA-7777
每小時費用(NT\$/每小時)：200 元

訂單資訊

取還車地點：新竹縣竹北市
租用結束時間：2025-06-19 08:00:00
租用結束時間：2025-06-21 17:00:00
狀態：
已完成
保險方案：
至尊全險方案 (費用 = NT\$2500)
保險額度：
保險項目：
保險說明：
保險說明：

付款資訊
總費用(未更動前)：3天 × (車輛費1800 + 保險費400) = 7800 元
總費用(更動後)：3天 × (車輛費1800 + 保險費2500) = 12900 元
付費方式：
現金
付費狀態：
已付款

顯示 | 資料檢視

日期	狀態	付費	費用	操作
2025-06-19 08:00:00	● 已完成	未付費	NT\$ 7800	查看/編輯
2025-06-19 23:00:00	● 已完成	未付費	NT\$ 3800	查看/編輯
2025-06-10 19:05:00	● 已完成	已付費	NT\$ 10800	查看/編輯
2025-06-27 18:00:00	● 已完成	已付費	NT\$ 10800	查看/編輯
2025-07-13 21:00:00	● 已完成	已付費	NT\$ 4400	查看/編輯
2025-06-28 00:40:00	● 已完成	已付費	NT\$ 5200	查看/編輯
2025-07-17 23:45:00	● 已完成	未付費	NT\$ 12400	查看/編輯

編輯成功

localhost 展示
修改成功

車款：TOYOTA Altis 1.2
車頭型：小頭車
頭燈方式：泡頭燈
CC數：1800
傳動力：自然進氣
可乘坐人數：5
車牌：AAA-7777
每小時費用(NT\$/每小時)：200 元

訂單資訊

取還車地點：新竹縣竹北市
租用結束時間：2025-06-19 08:00:00
租用結束時間：2025-06-21 17:00:00
狀態：
已完成
保險方案：
至尊全險方案 (費用 = NT\$2500)
保險額度：
保險項目：
保險說明：
保險說明：

付款資訊
總費用(未更動前)：3天 × (車輛費1800 + 保險費400) = 7800 元
總費用(更動後)：3天 × (車輛費1800 + 保險費2500) = 12900 元
付費方式：
現金
付費狀態：
已付款

顯示 | 資料檢視

日期	狀態	付費	費用	操作
2025-06-19 08:00:00	● 已完成	未付費	NT\$ 12900	查看/編輯
2025-07-18 16:06:00	● 已完成	未付費	NT\$ 18800	查看/編輯
2025-06-27 23:00:00	● 已完成	未付費	NT\$ 3800	查看/編輯
2025-06-10 19:05:00	● 已完成	已付費	NT\$ 10800	查看/編輯
2025-06-27 18:00:00	● 已完成	已付費	NT\$ 10800	查看/編輯
2025-07-13 21:00:00	● 已完成	已付費	NT\$ 4400	查看/編輯
2025-06-28 00:40:00	● 已完成	已付費	NT\$ 5200	查看/編輯
2025-07-17 23:45:00	● 已完成	未付費	NT\$ 12400	查看/編輯

玖、組員分工與心得

- 50915112 廖子科
 - 期末報告文件撰寫
 - 完整性限制
 - VIEW 規劃
 - 系統實作

心得：

在之前在學期間曾修過 2 次的資料庫相關課程，本次是第 3 次修資料庫，第一次是在專三必修課，當時必修課剛好有 php 課程，因此在兩門課程規劃上偏向以基本規劃資料庫架構並"實作出系統結果"為主，資料庫課程是使用 SQL Server、MySQL，大部分使用 GUI 方式建立系統資料庫。第二次則是業界課程，都是教資料庫的 CRUD。

但在本次大學部課程中，發現到與先前課程有所差異，很注重資料庫設計過程，像是規劃概念層到外部層循跡漸進，最有感還是完整性限制，之前認知都是單純以前+端程式去避免不正確格式的資料，但忽略到"資料庫"儲存的資料也要符合限制，可以使用 CHECK 這類，甚至搭配 Regex 正規表達式，是一個很好的實作限制的功能。同時在期末報告中，我也思考到一個問題，在於同一時間，有兩個人預約同一輛車且租用時間重疊，這時後端開發處理不好做，但可以使用資料庫的 Transaction(lock)解決該問題。還有一點是 VIEW 的實作，在過往也是完全使用原始 Table 實作出系統，上完本堂課，瞭解到 VIEW 是一個很好用的物件，一部分是從既有 Table 可以延伸出符合不同使用者族群的 Table，另一部分是可以減少後端程式開發的複雜性，以車輛系統為例，開一個 VIEW 單純 JOIN 汽車和車型資料表，後端程式就能少寫很多 SQL 語法，直接 SELECT 那個 VIEW 就好。透過 VIEW 也可以搭配 GRANT 做使用者權限控制，達到讓每個使用者群最小資料存取權限。

- 51015107 廖玟嫻
 - 期末報告文件撰寫
 - 規劃 ERD 圖

心得：

- 51015130 林敬雅
 - 期末報告文件撰寫
 - 資料庫 schema 設計
 - 規劃 ERD 圖

心得：

雖然我之前就修過資料庫相關的課程，但這學期重新接觸這門課之後，才發現其實自己以前對資料庫設計的很多觀念都還不夠扎實。

上課聽完老師講解都覺得好像都懂了，可是實際在做作業的時候，才會意識到原來有這麼多細節要顧，也讓我重新反思資料庫設計這件事，其實比我想像中更複雜更重要。我們這組做的是租車管理系統的資料庫，剛開始我真的不太知道從哪裡下手，後來除了看講義，也去查了不少別人的專題、GitHub 上類似的專案，觀察他們怎麼處理租賃資料、會員權限、罰金邏輯等等。從別人的設計中學到了不少細節，慢慢建立自己的判斷基準，也更懂得怎麼思考資庫的結構。

如果有學弟妹問我這門課到底值不值得修，我會很願意推薦，就算以前學過資料庫，再修一次這門課真的會有不同的收穫，很多以前我以為會的，其實只是「看得懂」，但不一定「做得出來」，尤其是要實作之後才會知道哪些觀念自己其實還不熟。

壹拾、 參考文獻

- [1]. Chen, K.-J. (2023, November 22). PHP 接 Google 第三方登入 API 最完整與最簡易範例. Medium.
- [2]. Lin, V. (2020, December 18). 透過 Google People API 取得更多使用者資料. Vocus.
- [3]. 常駐程式. (n.d.). 大家遺忘的 enum. iT 邦幫忙.'
- [4]. 公路總局. (2025). 《合法租賃車輛選擇與消費者保障指南》. 高雄市區監理所秘書室公告。來源：交通部公路總局，<https://www.thb.gov.tw>
- [5]. 內政部 (2025)。地址資料標準。政府資料標準平台。取自 <https://schema.gov.tw/lists/161>