

# Python Mini Project of Car Booking System



**Submitted to:**

Cherry Khosla 13436

**Submitted by:**

Name	Registration number
Sangapalli Jaswanth	12105557
Harsh Vedit	12104689
Radhika Goyal	12107577

**Section: K21GP**

**Course Code: INT 213**

# INDEX

S.No.	Title	Page
1	Introduction i. Description ii. Scope	3-3 3 3
2	Design i. User Interface ii. Database iii. Algorithms	3-4 3 3 4
3	Source Code	4-4
4	Results and Screenshots	5-8
5	Contributions	9-9
6	References	9-9

## 1. Introduction:

### i. Description:

This app automates the process of car rental system, by taking details of customer and his selection of cars and giving the bill after the car is relieved.

### ii. Scope:

This application handles user interfaces and databases and is limited to insert into, create and update the database records when a car is being rented or relieved and calculate the bill according to number of minutes it is used.

## 2. Design:

### i. User Interface:

User interface consists of two forms in separate frames, one takes inputs to book a car and another to relieve a car.

The form which allows the user to book consists of

- Text Entry elements for Name, Driving License Number, Mobile Number and Emergency Mobile number.
- Radio Buttons to get Gender
- Spin Box widget, which allows values from 18 to 100 to get Age
- Combo Box widget to select from available car
- Submit button which runs the command

The form which allows the user to relieve car and fetch the bill consists of two combo boxes which have the values referring to the registration number of booked cars and licenced of user who booked them respectively.

### ii. Database:

A table 'car' to store the details of cars, which has attributes:

- 'regNo,' a string variable to store registration number of car which serves as a primary key
- 'chasisNo,' a string variable to store chassis number of car
- 'manufacturer,' a string variable to store manufacturer company
- 'model,' a string variable to store the model of car
- 'no\_of\_seats,' and integer variable to store number of seats in car

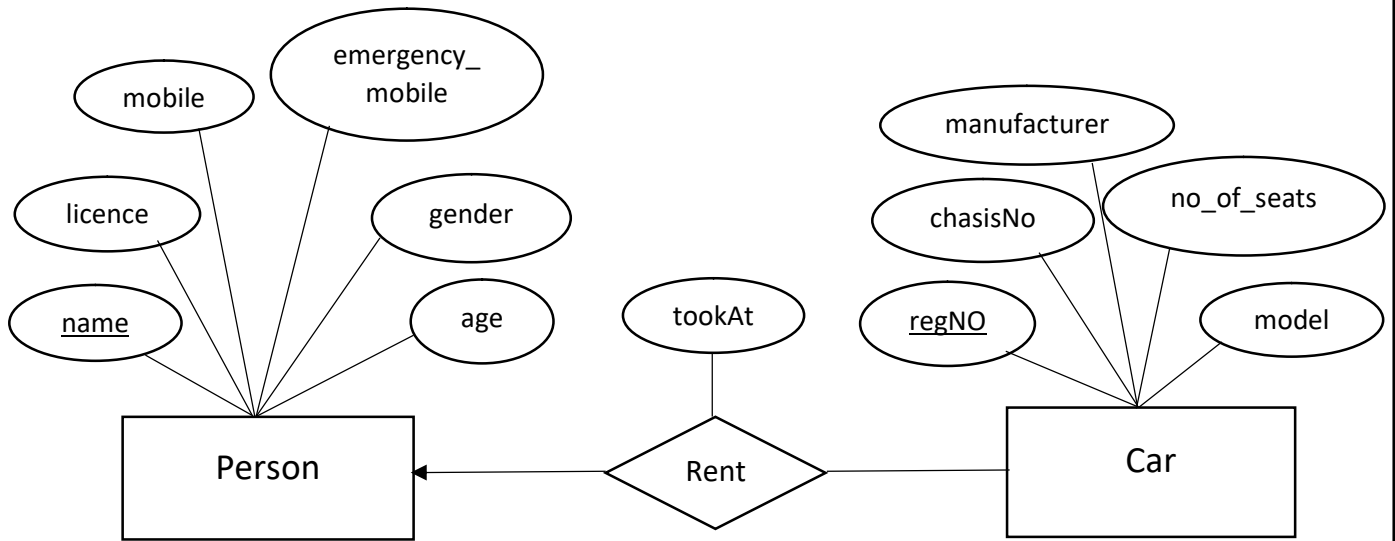
A table 'person' to store the details of the customer, which has attributes:

- 'name,' a string variable to store Name of the customer
- 'age,' an integer variable to store the Age of the customer
- 'licence,' a string variable to store the Driving Licence Number of the customer and serves as a primary key
- 'mobile,' an integer variable to store the Mobile Number of the customer
- 'emergency\_mobile', an integer variable to store the Emergency contact's Mobile number of the customer.

A table 'rent' which relates 'car' and 'person' tables to store rental details which has attributes:

- 'carNO,' a string variable to store the Registration Number of car
- 'plicence,' a string variable to store the Driving Licence of the customer

- ‘tookAt,’ an integer variable to store the time at which the car is taken for rent.



An Entity-Relationship Diagram denoting the attributes and relation between tables

### iii. Algorithms:

#### Algorithm for Booking a Car:

- User enters inputs and submits the form
- Application checks if the car is available and prompts user to select another car
- If the car is available, it searches for the user's licence in 'person' table.
- If found, it asks user if an update is needed in his information, updates if he selects yes and skips if no.
- If not found, it inserts the user's data into 'person' table.
- Finally, it inserts the car's registration number, user's driving licence number and current time into the 'rent' table.

#### Algorithm to relieve car:

- User enters the inputs and submits the form
- Application checks if the driving licence and car's registration number entered by the user matches according to the 'rent' table
- If it does not match, the application prompts user to select a valid combination
- If it matches, the application reads the corresponding time taken and subtracts it from current time
- The result undergoes floor division by 60 to calculate number of minutes followed by multiplication with rent per minute to calculate bill
- The bill calculated is informed to the user.

### 3. Project Source Code:

The source code to the project is on the is on GitHub on the URL

[https://github.com/jaswanthsnpg/Car\\_Booking\\_System\\_in\\_Python](https://github.com/jaswanthsnpg/Car_Booking_System_in_Python) .

## 4. Results and Screenshots:

**Car Booking System**

**Book Car**

Name:

Age:

Gender: ☒ Male ☐ Female ☐ Other

Driving Licence Number:

Mobile Number:

Emergency Mobile:

Select the car:

**Book**

**Relieve Car**

Select the Car:

Select the Licence:

**Relieve**

Figure 1: Initial Form

MySQL 8.0 Command Line Client

```
mysql> select * from person;
```

name	age	gender	licence	mobile	emergency_mobile
Deepak	20	Male	IND27030NE	1234567890	9012345678
Kunal	25	Male	IND89020LIK	2345678901	8901234567
Harsh	18	Male	P18A	7890123456	7765432109

3 rows in set (0.00 sec)

```
mysql> select * from car;
```

regNo	chassisNo	manufacturer	model	no_of_seats
KA19PB487	3H4KA1150RT000268	Renault	Kwid	5
KA19PB488	1G8G2G1270A2157299	Maruti	Alto k10	5
KA19PB489	5N37A0ME6AN006847	Hyundai	Santro	5
KA19PB490	3H4DA185035005062	Bajaj	Qute	4
KA19PB491	SCA15684XIX07444	Datsun	Redi- GO	5
KA19PB492	3H4DB1570P5000058	Maruti	Alto-800	5
KA19PB493	ZCF357A5BM1953433	Tata	Tiago	5
KA19PB494	13AFA29P4VP728937	Maruti	S-presso	5
KA19PB495	1FVACYDT19HA32694	Maruti	Suzuki-eco	7
KA19PB496	1G8MF35X68Y131819	Maruti	Suzuki-WagonR	5
KA19PB497	3F2SHAC20G417189	Mahindra	XUV300	5
KA19PB498	3T3H18516T0133040	Maruti	Suzuki Grand Vitara	5
KA19PB499	3W4S881H0M210368	Mahindra	Bolero Neo	7
KA19PB500	3H4KA162K018197	Toyota	Urban Cruiser	5
KA19PB501	3H4C2560H000414	Mahindra	Scorpio Classic	7
KA19PB502	3G1MR215XK752025	Hyundai	i20	5
KA19PB503	2CNB113C3Y6924710	Tata	Nexon	5
KA19PB504	3H4KA1151K019450	Hyundai	Verna	5
KA19PB505	3H4BA22NE5C0016953	Skoda	Kushaq	5
KA19PB506	3H4KA1170K0000665	Toyota	Innova-Crysta	7
KA19PB507	W4UDFAFI6D0014563	Lamborghini	Aventador	2
KA19PB508	3H4DA9460H5032070	Rolls-Royce	Phantom	5
KA19PB509	1MEBP67D58F617327	Rolls-Royce	Droptop-Coupe	5
KA19PB510	3H4BA1150RT000683	Bugatti	Veyron Grand sports	2
KA19PB511	2DM1U859A660015210	BMW	X7	7
KA19PB512	3W4P671K97M122542	Mercedes	Maybach-S	5
KA19PB513	5FMR118613B0046732	Toyota	Landcruiser	8
KA19PB514	3W4DX7A15B0006256	Mercedes	G-Wagon	5

28 rows in set (0.00 sec)

```
mysql> select * from rent;
```

carNo	pllicence	tookAt
KA19PB489	P18A	1667840559

Figure 2: Databases used along with values

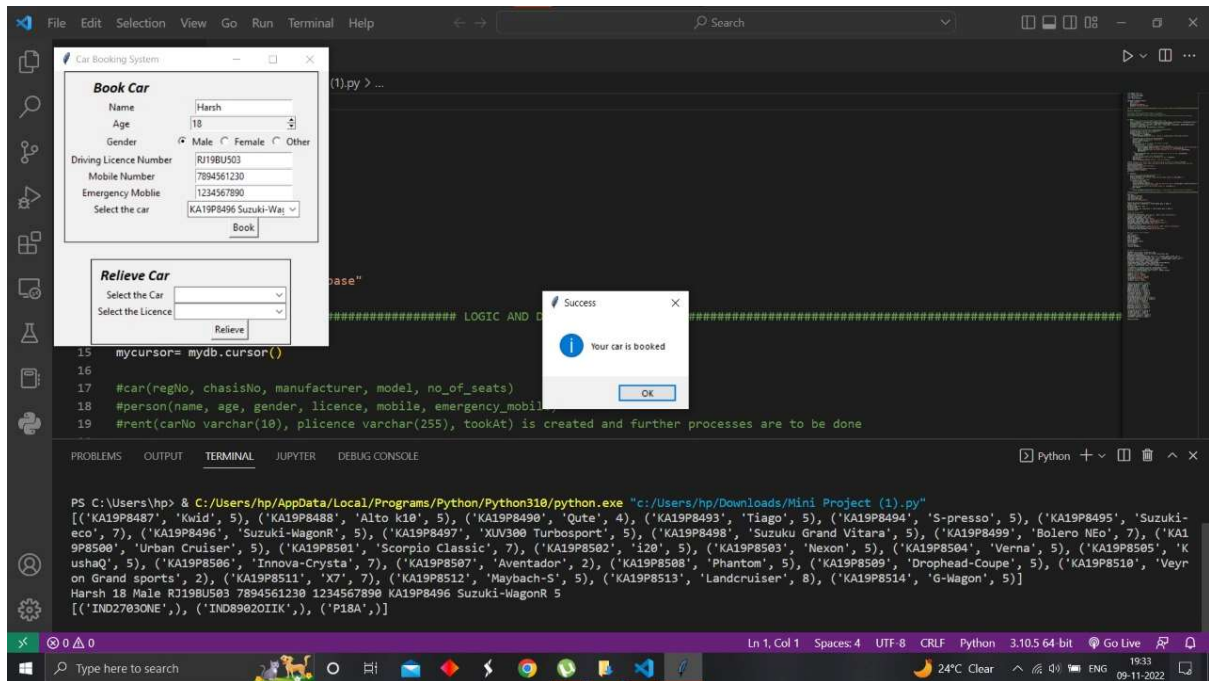


Figure 3: Message being displayed to the user that the car is booked

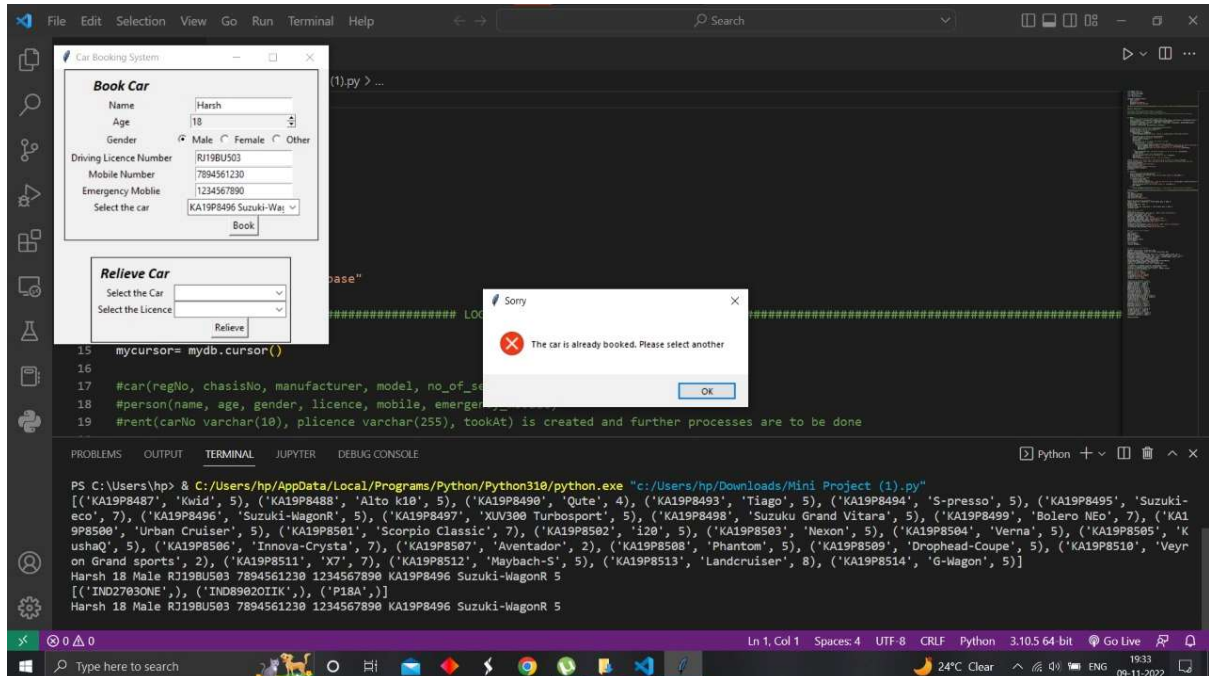


Figure 4: A message showing error message to the user that the car is already booked on repeated booking



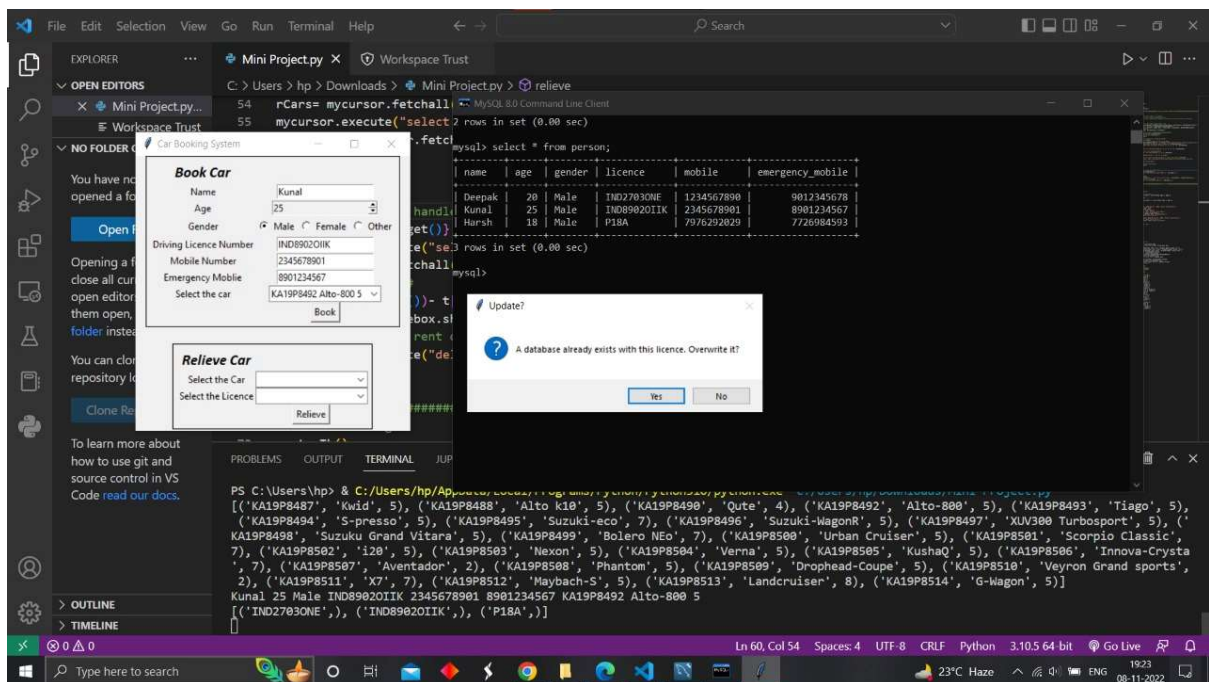


Figure 5: The Application asking user if it must update details after the form is filled again for same primary key

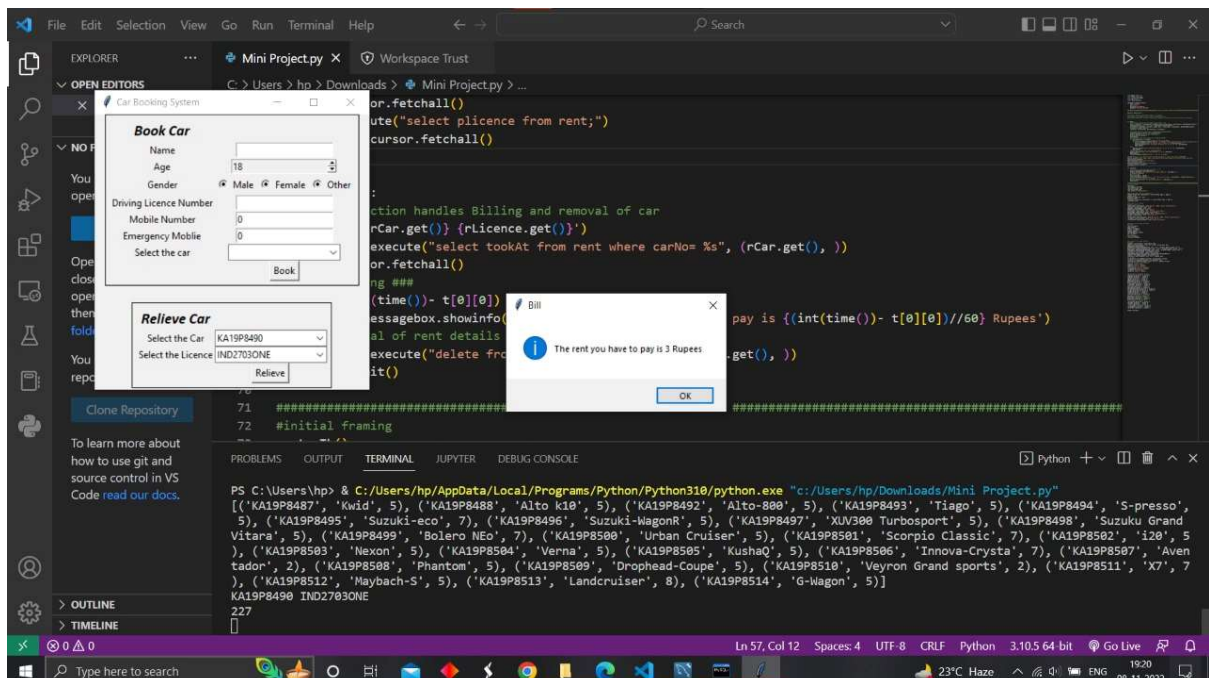


Figure 6: Informing user the calculated bill after relieving car

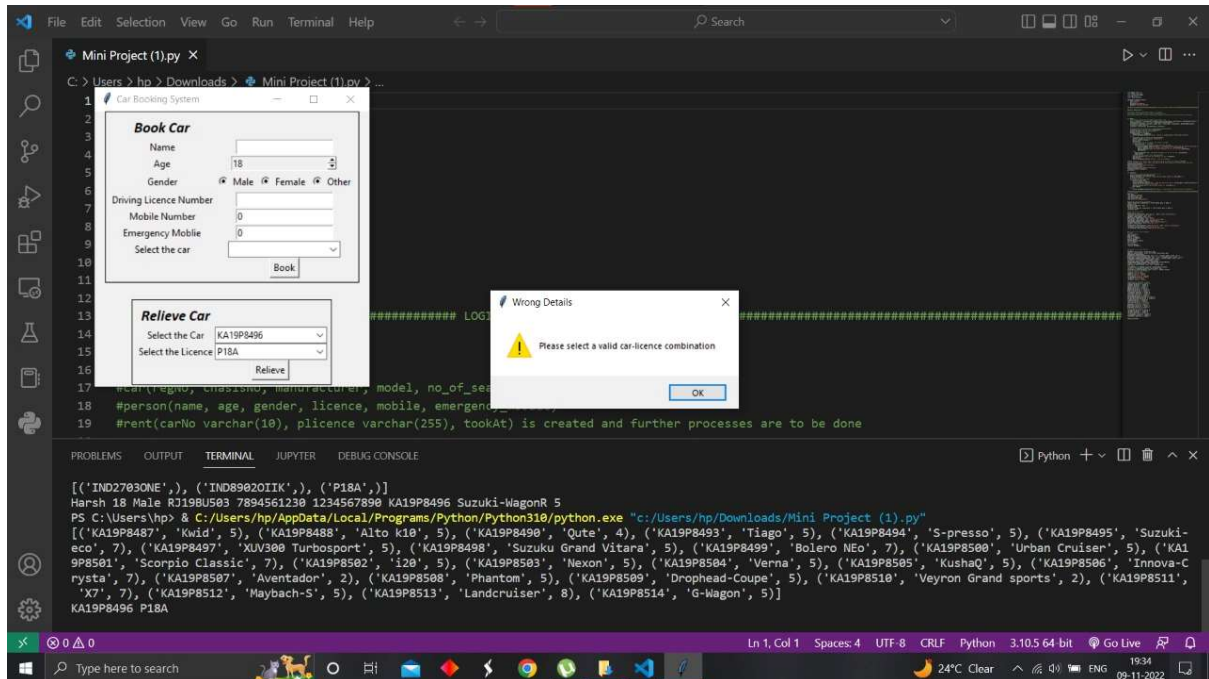


Figure 7: Prompt to user about wrong combination of vehicle and licence attributes

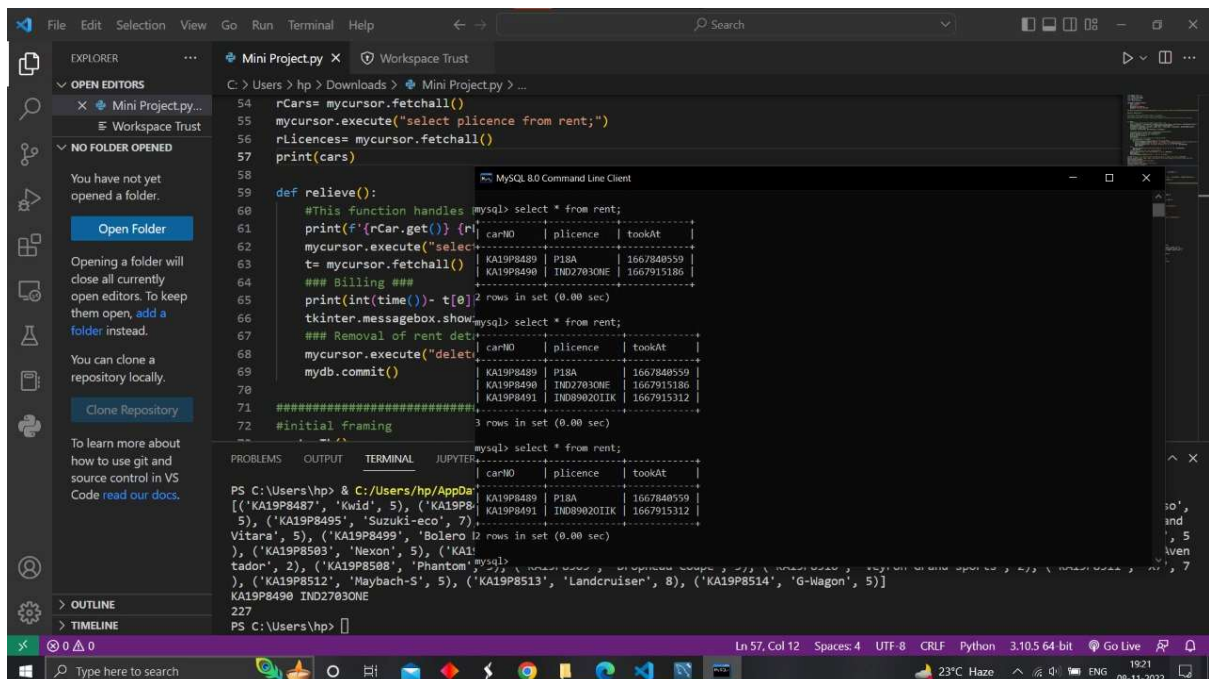


Figure 8: rent database before and after booking and relieving the car



## 5. Contribution:

### **Radhika Goyal:**

Report Writing and Reviewing the project and sample values for databases.

### **Jaswanth Sanagapalli:**

User Interface Design and Construction and Integrating it with the Databases.

### **Harsh Vedit:**

Design and Construction of Databases.

## 6. References:

- 'Python Time Module Reference' from GeeksForGeeks on URL <https://www.geeksforgeeks.org/python-time-module/> which is used to find number of seconds the car is on rent to calculate bill.
- 'Tkinter MessageBox Reference' from Python Documentation on URL <https://docs.python.org/3/library/tkinter.messagebox.html#module-tkinter.messagebox> which is used to display information, warning and error messages on the interface.
- 'Python Tkinter ComboBox Reference' from GeeksForGeeks on URL <https://www.geeksforgeeks.org/combobox-widget-in-tkinter-python/> which is used to make user select the car when booking.
- 'Python Tkinter ComboBox Tutorial' from Python Tutorial on URL <https://www.pythontutorial.net/tkinter/tkinter-combobox/> which is used to limit the user to select only those car available (by changing state of widget to 'readonly').
- 'Python Tkinter SpinBox Reference' from GeeksForGeeks on URL <https://www.geeksforgeeks.org/python-tkinter-spinbox/> which is used to take the input for age to book car.
- 'Python Tkinter SpinBox Reference' from Python Documentation on URL <https://docs.python.org/3/library/tkinter.ttk.html#spinbox> which is used to restrict user from entering out of bound values.
- 'Python MySQL Tutorial' from Tech With Tim on YouTube [https://www.youtube.com/watch?v=3vsC05rxZ8c&ab\\_channel=TechWithTim](https://www.youtube.com/watch?v=3vsC05rxZ8c&ab_channel=TechWithTim)