Introduction:
This report describes a small-scale car rental business and the application developed to manage its operations efficiently. The Car Rental Service Application is a streamlined tool designed to manage the booking and rental of vehicles for a small business. The application integrates three core technologies: Microsoft Access, Excel, and VBA. By combining these tools, the application simplifies business operations by automating processes such as updating bookings, tracking car availability, and calculating revenue. It offers an organized and efficient way to handle all aspects of the rental process, saving time and minimizing errors.

The Database forms the backbone of the application, housing all critical business data in three interrelated tables:

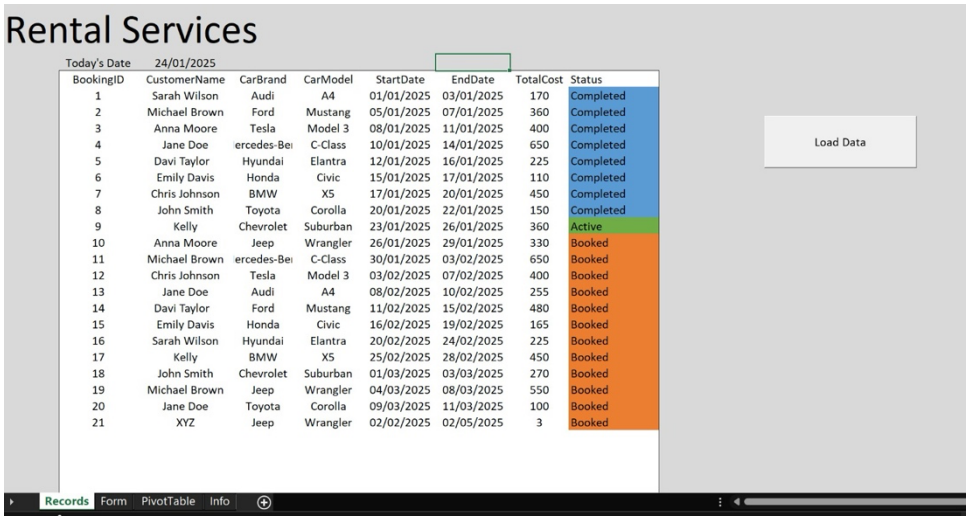
1. Customers Table:
    - Stores customer information such as CustomerID (Primary Key), CustomerName, Phone, Email, LicenseNumber, and Address.
2. Cars Table:
    - Contains details about available cars, including CarID (Primary Key), CarBrand, CarModel, CarType, Year, DailyRate, and NumberPlate.
3. Bookings Table:
    - Tracks rental information with fields such as BookingID (Primary Key), CustomerID, CarID, StartDate, EndDate, and TotalCost.
    - Links customers and cars using foreign keys (CustomerID and CarID).

Queries:

1. Average Rental Days Per Car:
    - Calculates the average rental duration for each car model.
2. Booking Count Per Car:
    - Counts the total number of bookings for each car model.
3. Detailed Booking Information:
    - Retrieves comprehensive booking data, linking customers, cars, and bookings.
4. Total Revenue by Car:
    - Calculates the Total Revenue by each car model

The Excel front-end provides an easy-to-use interface for users to interact with the application:

1. Records Sheet:



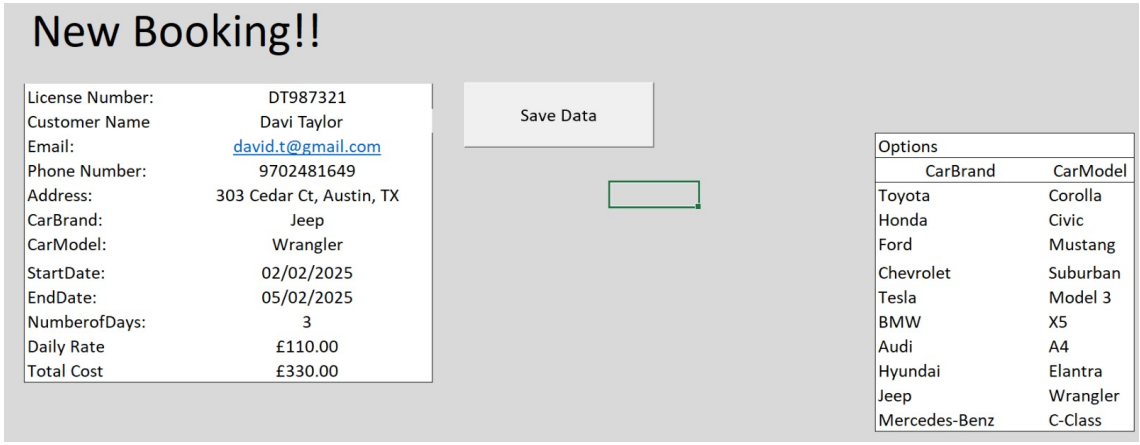
BookingID	CustomerName	CarBrand	CarModel	StartDate	EndDate	TotalCost	Status
1	Sarah Wilson	Audi	A4	01/01/2025	03/01/2025	170	Completed
2	Michael Brown	Ford	Mustang	05/01/2025	07/01/2025	360	Completed
3	Anna Moore	Tesla	Model 3	08/01/2025	11/01/2025	400	Completed
4	Jane Doe	ercedes-Be	C-Class	10/01/2025	14/01/2025	650	Completed
5	Davi Taylor	Hyundai	Elantra	12/01/2025	16/01/2025	225	Completed
6	Emily Davis	Honda	Civic	15/01/2025	17/01/2025	110	Completed
7	Chris Johnson	BMW	X5	17/01/2025	20/01/2025	450	Completed
8	John Smith	Toyota	Corolla	20/01/2025	22/01/2025	150	Completed
9	Kelly	Chevrolet	Suburban	23/01/2025	26/01/2025	360	Active
10	Anna Moore	Jeep	Wrangler	26/01/2025	29/01/2025	330	Booked
11	Michael Brown	ercedes-Be	C-Class	30/01/2025	03/02/2025	650	Booked
12	Chris Johnson	Tesla	Model 3	03/02/2025	07/02/2025	400	Booked
13	Jane Doe	Audi	A4	08/02/2025	10/02/2025	255	Booked
14	Davi Taylor	Ford	Mustang	11/02/2025	15/02/2025	480	Booked
15	Emily Davis	Honda	Civic	16/02/2025	19/02/2025	165	Booked
16	Sarah Wilson	Hyundai	Elantra	20/02/2025	24/02/2025	225	Booked
17	Kelly	BMW	X5	25/02/2025	28/02/2025	450	Booked
18	John Smith	Chevrolet	Suburban	01/03/2025	03/03/2025	270	Booked
19	Michael Brown	Jeep	Wrangler	04/03/2025	08/03/2025	550	Booked
20	Jane Doe	Toyota	Corolla	09/03/2025	11/03/2025	100	Booked
21	XYZ	Jeep	Wrangler	02/02/2025	02/05/2025	3	Booked

    - Displays all booking records, including BookingID, CustomerName, CarBrand, CarModel, StartDate, EndDate, TotalCost, and Status.
    - Conditional formatting is applied using new rules:

Red: For Booked.
Green: For Active.
Blue: For Completed.

- Includes a "Load Data" button that refreshes the sheet with data from the database.

2. Form Sheet:



- Serves as the main interface for adding new bookings.
- Includes fields for customer details (e.g., CustomerName, LicenseNumber) and booking details (e.g., CarBrand, CarModel, StartDate, EndDate, TotalCost).
- "Save Data" button: Submits form data to the database, creating or updating records in the Customers and Bookings tables.
- Features:
  Customer Name, Email,Phone Number, Address: Dynamically fetched using VLOOKUP
  CarBrand: Uses data validation to display a dropdown option.
  CarModel: Dynamically fetched using VLOOKUP
  Number of Days: Automatically calculated using IF statement
  Daily Rates: Retrieved from the Info worksheet using VLOOKUP

3. Pivot Table Sheet:

Sum of TotalCost	Column Labels			
Row Labels	Jan	Feb	Mar	Grand Total
Audi	**£170.00**	**£255.00**		**£425.00**
A4	£170.00	£255.00		£425.00
BMW	**£450.00**	**£450.00**		**£900.00**
X5	£450.00	£450.00		£900.00
Chevrolet	**£360.00**		**£270.00**	**£630.00**
Suburban	£360.00		£270.00	£630.00
Ford	**£360.00**	**£480.00**		**£840.00**
Mustang	£360.00	£480.00		£840.00
Honda	**£110.00**	**£165.00**		**£275.00**
Civic	£110.00	£165.00		£275.00
Hyundai	**£225.00**	**£225.00**		**£450.00**
Elantra	£225.00	£225.00		£450.00
Jeep	**£330.00**		**£550.00**	**£880.00**
Wrangler	£330.00		£550.00	£880.00
Mercedes-Benz	**£1,300.00**			**£1,300.00**
C-Class	£1,300.00			£1,300.00
Tesla	**£400.00**	**£400.00**		**£800.00**
Model 3	£400.00	£400.00		£800.00
Toyota	**£150.00**		**£100.00**	**£250.00**
Corolla	£150.00		£100.00	£250.00
Grand Total	**£3,855.00**	**£1,975.00**	**£920.00**	**£6,750.00**

- Summarizes total revenue generated by each car, grouped by month and year.

- Displays key metrics like TotalCost for each car brand and model, enabling better business decision-making.

VBA Middleware

The VBA middleware connects the Excel front-end with the Access database, automating data transfer and enabling advanced functionality. Key subroutines include:

LoadBookingData:

- Fetches booking data from the database and populates the Records Sheet.
- Dynamically applies booking statuses (Booked, Active, or Completed) based on the date in cell C3 as SpecifiedDate
- Uses a Do While loop to iterate through rows and update statuses.

```
' Add Status using Do While and If-Else
Dim Count As Long
Dim SpecifiedDate As Date
Dim StartDate As Date
Dim EndDate As Date

SpecifiedDate = Sheets("Records").Range("C3").Value

Count = 5

Do While Sheets("Records").Range("B" & Count).Value <> ""
    ' Read StartDate and EndDate from the sheet
    StartDate = Sheets("Records").Cells(Count, "F").Value
    EndDate = Sheets("Records").Cells(Count, "G").Value

    If SpecifiedDate < StartDate Then
        Sheets("Records").Cells(Count, "I").Value = "Booked"
    ElseIf SpecifiedDate >= StartDate And SpecifiedDate <= EndDate Then
        Sheets("Records").Cells(Count, "I").Value = "Active"
    Else
        Sheets("Records").Cells(Count, "I").Value = "Completed"
    End If

    Count = Count + 1
Loop
```

- Initialize the Counter as Count = 5 starts the iteration from row 5 (as the first row of data).
- Do While Sheets("Records").Range("B" & Count).Value <> "": Continues looping until an empty cell is found in column B (indicating the end of data).
- StartDate and EndDate are fetched from columns F and G, respectively, for the current row.
- Booked: If the SpecifiedDate (e.g., today's date) is earlier than the StartDate.
- Active: If the SpecifiedDate falls between the StartDate and EndDate
- Completed: If the SpecifiedDate is after the EndDate.
- Updates the "Status" column (I) with the calculated value for the current row.
- Count = Count + 1 increments the row counter to process the next record.

SaveNewBooking:

- Reads form data from the Form Sheet and saves customer and booking details to the database.
- Inserts a new record into the Customers table and links the booking with the correct CustomerID and CarID.

DisplayCarInfo:

- Retrieves car-related details (CarBrand, CarModel, DailyRate, TotalCost, StartDate) and populates the Info Sheet.
- Serves as the data source for the pivot table on the Pivot Table Sheet.

Conclusion

The Car Rental Service Application efficiently manages rental operations by integrating Microsoft Access, Excel, and VBA to handle customer, car, and booking data. It provides an intuitive interface for adding and tracking bookings, real-time status updates, and insightful analytics through pivot tables. This application can be further scaled to address more complex business requirements and enhance operational efficiency.

In the future, the system can incorporate additional features, such as a Maintenance Table to track car maintenance records. This table could include fields like MaintenanceID, CarID, MaintenanceDate, Description, and Cost. If a car requires maintenance, its associated costs could be automatically factored into the TotalCost for bookings, ensuring accurate cost calculations and improving fleet management.

Additionally, an Extra Cost Mechanism could be introduced by adding a new variable, PricePerHour. If a customer fails to return the car by the specified EndDate, they could be charged an hourly penalty based on this rate. This feature would not only incentivize timely returns but also ensure that the business covers any revenue losses caused by delayed returns.

These enhancements, combined with cloud-based deployment and advanced analytics, would make the application robust enough to support larger businesses. By continuously adapting to new requirements, the Car Rental Service Application can evolve into a comprehensive solution for rental service management

GitHub Link: https://github.com/KanishkaBisen/car-rental.git

Reference:
SQL: Completed these course in DataCamp
https://app.datacamp.com/learn/courses/intermediate-sql

https://app.datacamp.com/learn/courses/joining-data-in-sql

https://app.datacamp.com/learn/courses/data-manipulation-in-sql

Excel:
https://www.youtube.com/@KenjiExplains

VBA: Lecture/Tutorial Notes