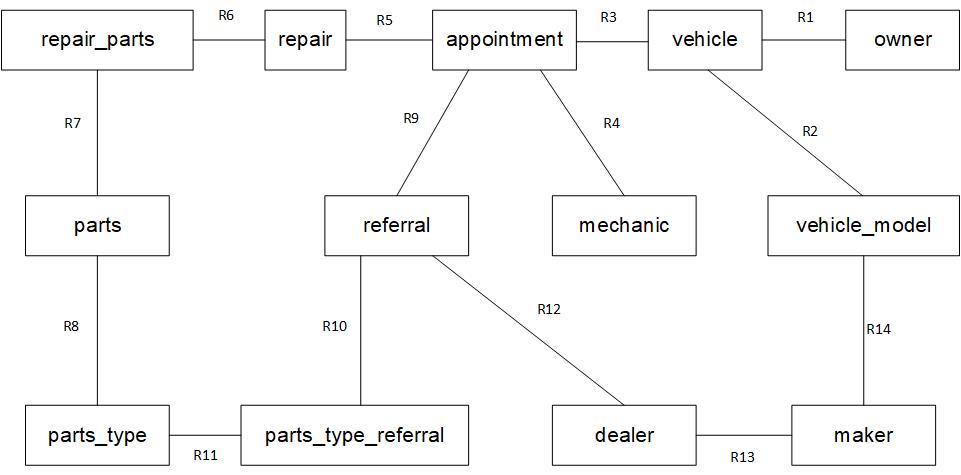
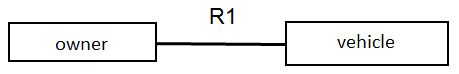
Introduction: Wynd’s Motoring Repairs is a car repair business based in London, UK. For their business improve purpose they wants a design centralized database system. So, database system how to design and developing in a structured, organized and standardized way. It’s discussed on task based.

Task-1

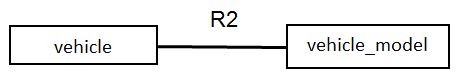
1. Wynd’s Motoring Repairs Entity Relationship (ER) model.





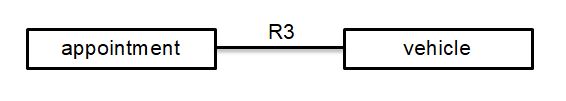
**Relationship type:** One to many

**Relationship description:** One vehicle must have one owner and one owner can have one or more vehicle.



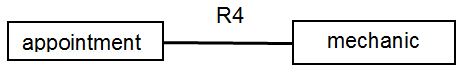
**Relationship type:** One to many

**Relationship description:** One or more vehicle model must have one vehicle and one vehicle have one or more vehicle model.



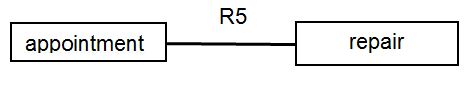
**Relationship type:** One to many

**Relationship description:** One or more appointment must have one vehicle and one vehicle have one or more appointment.



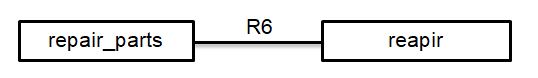
**Relationship type:** One to many.

**Relationship description:** One or more appointment must have one mechanic and one mechanic have one or more appointment.



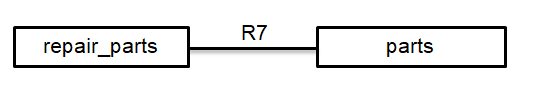
**Relationship type:** One to many.

**Relationship description:** One or more repair must have one appointment and one appointment can have one or more repair.



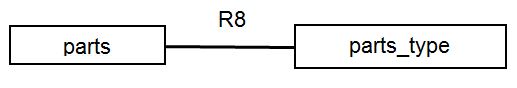
**Relationship type:** One to many.

**Relationship description:** One or more repair parts must have one repair and one repair can have one or more repair parts.



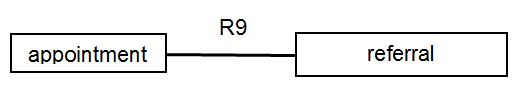
**Relationship type:** One to many.

**Relationship description:** One or more repair parts must have one parts and one parts can have one or more repair parts.



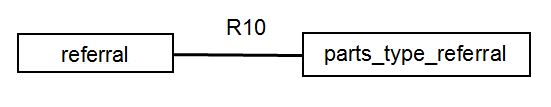
**Relationship type:** One to many.

**Relationship description:** One or more parts type must have one parts and one parts can have one or more parts type.



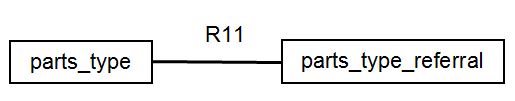
**Relationship type:** One to many.

**Relationship description:** One or more referral must have one appointment and one appointment can have one or more referral.



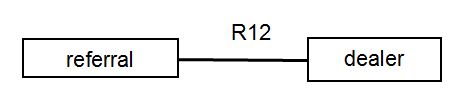
**Relationship type:** One to many.

**Relationship description:** One or more parts type referral must have one referral and one referral can have one or more parts type referral.



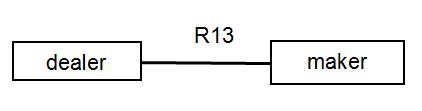
**Relationship type:** One to many.

**Relationship description:** One or more parts type must have one parts type referral and one parts type referral can have one or more parts type.



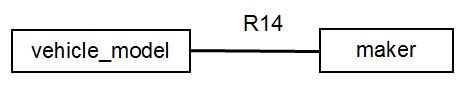
**Relationship type:** One to many.

**Relationship description**: One or more referral must have one dealer and one dealer can have one or more referral.



**Relationship type:** One to many.

**Relationship description:** One or more maker can have one dealer and one dealer can have one or more maker.



**Relationship type:** One to many.

**Relationship description:** One or more vehicle model must have one maker and one maker can have one or more vehicle model.

1. Relational schema ER diagram primary key and foreign key with data dictionary.

owner

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Owner\_Id | varchar | 10 | Yes |  |  |
| Owner\_Name | varchar | 25 |  |  |  |
| Owner\_Address | varchar | 150 |  |  |  |
| Owner\_Telephone | varchar | 15 |  |  |  |

mechanic

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Mechanic\_Id | int | 10 | Yes |  |  |
| Mechanic\_Name | varchar | 20 |  |  |  |

maker

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Maker\_Id | int | 10 | Yes |  |  |
| Maker\_Name | varchar | 25 |  |  |  |

dealer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Dealer\_Id | int | 10 | Yes |  |  |
| Dealer\_Name | varchar | 30 |  |  |  |
| Maker\_Id | int | 10 |  | Yes | maker |

vehicle\_model

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Vehicle\_Model\_Id | int | 20 | Yes |  |  |
| Vehicle\_Model\_Name | varchar | 25 |  |  |  |
| Maker\_Id | int | 10 |  | Yes | maker |

vehicle

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Vehicle\_Id | varchar | 10 | Yes |  |  |
| Vehicle\_Model\_Id | int | 10 |  | Yes | vehicle\_model |
| Owner\_Id | varchar | 10 |  | Yes | owner |

appointment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Appointment\_Id | int | 10 | Yes |  |  |
| Appointment\_Time | varchar | 7 |  |  |  |
| Appointment\_Date | date | 10 |  |  |  |
| Appointment\_Cost | float | 10 |  |  |  |
| Vehicle\_Id | varchar | 10 |  | Yes | vehicle |
| Mechanic\_Id | int | 10 |  | Yes | mechanic |

parts\_type

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Parts\_Type\_Id | int | 10 | Yes |  |  |
| Parts\_Type\_Name | varchar | 25 |  |  |  |

parts

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Parts\_Id | int | 10 | Yes |  |  |
| Parts\_Name | varchar | 25 |  |  |  |
| Parts\_Quantity | int | 3 |  |  |  |
| Parts\_Type\_Id | Int | 10 |  | Yes | parts\_type |

referral

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Referral\_Id | int | 10 | Yes |  |  |
| Referral\_Name | varchar | 20 |  |  |  |
| Referral\_Date | date | 10 |  |  |  |
| Referral\_Problem | varchar | 255 |  |  |  |
| Dealer\_ID | int | 10 |  | Yes | dealer |
| Appointment\_Id | int | 10 |  | Yes | appointment |

parts\_type\_referral

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Parts\_Referral\_Id | int | 10 | Yes |  |  |
| Referral\_Id | int | 10 |  | Yes | referral |
| Parts\_Type\_Id | int | 10 |  | Yes | parts\_type |

repair

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Repair\_Id | int | 10 | Yes |  |  |
| Repair\_Details | varchar | 255 |  |  |  |
| Appointment\_Id | int | 10 |  | Yes | appointment |

repair\_ parts

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Repair\_Parts\_Id | int | 10 | Yes |  |  |
| Parts\_Id | int | 10 |  | Yes | parts |
| Reapir\_Id | int | 10 |  | Yes | repair |

1. Database management system relational schema many to many relationships in database design don’t support. So, many to many relationship breakdowns to create a one to many relationships in particular table and linked with primary and foreign key in entity relationship model.

**Data integrity:** In the information security processes data integrity is a fundamental component. It’s design and authenticate for checking error and validation requirements.

**Example:** Maintained whole database system data integrity maintains database numeric columns which can only accept numeric value in database.

**Referential integrity:** In the Relational database concept reference data consist any foreign key attribute which must be agree with another table primary key reference table attribute.

**Example:** relational database system every table has one primary key which can be reference on another table primary key against another attribute.

**Functional dependence:** Function helps ensure the validity of data for uniquely determines relationship between one to another attribute.

**Example:** Relational database table has one unique Key. This table must have one or more attribute. So, this table another attributes are depend on unique key attribute.

So, the Wynd’s Motoring Repairs database system relational schema has data integrity, referential integrity and functional dependence in developing the database system.

Task-2

1. All owner, maker, vehicle and maker table data insert and show in figure 1 to 8.

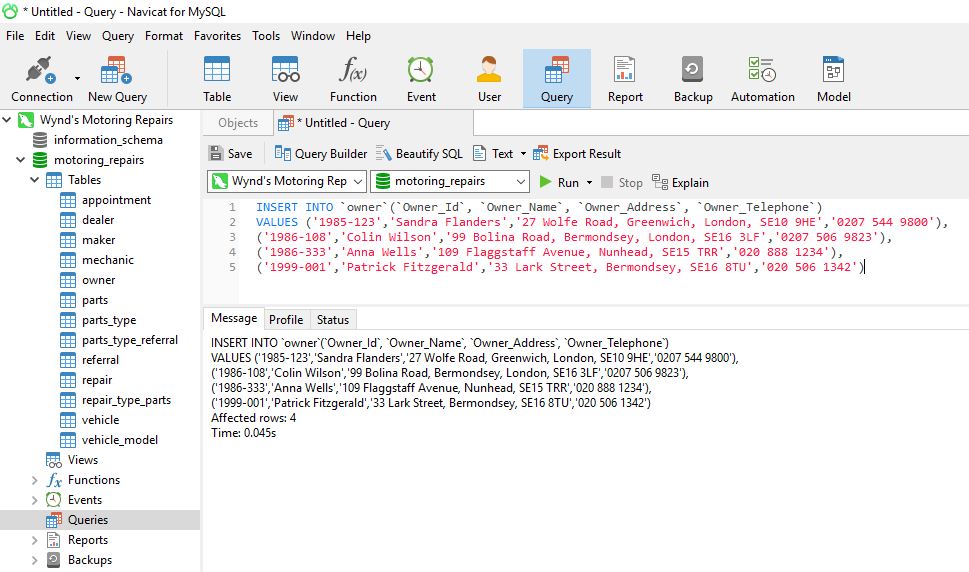


Figure-1

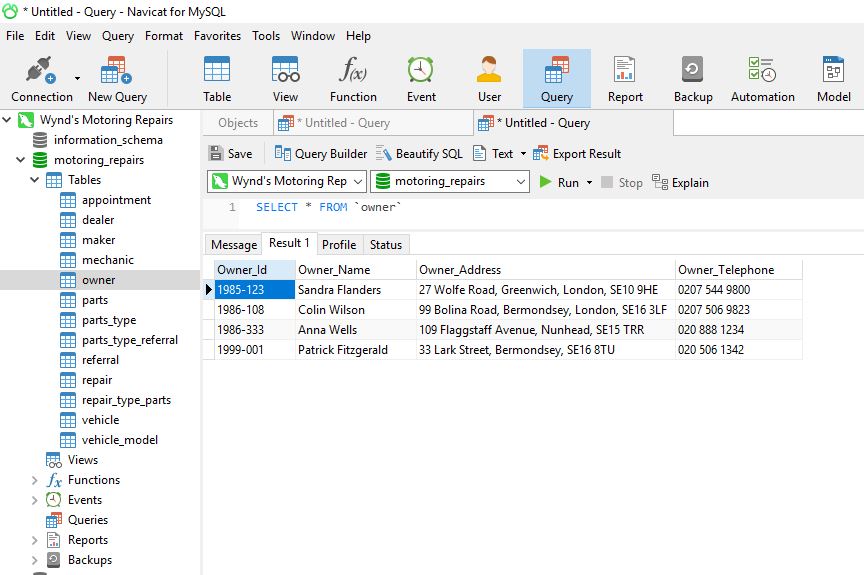


Figure-2

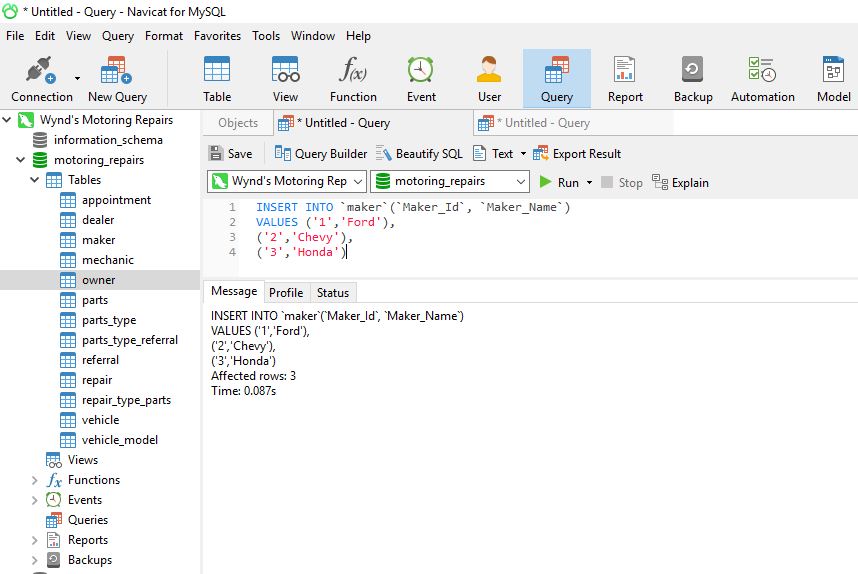
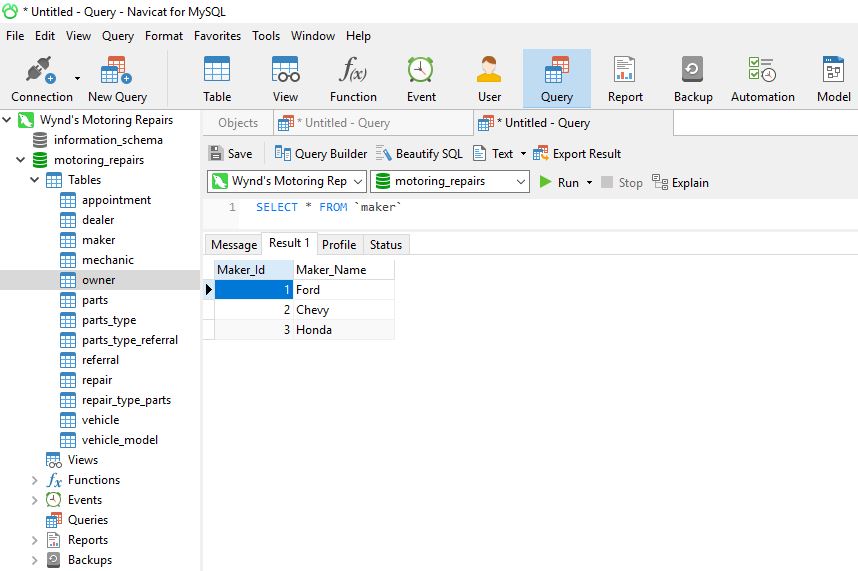
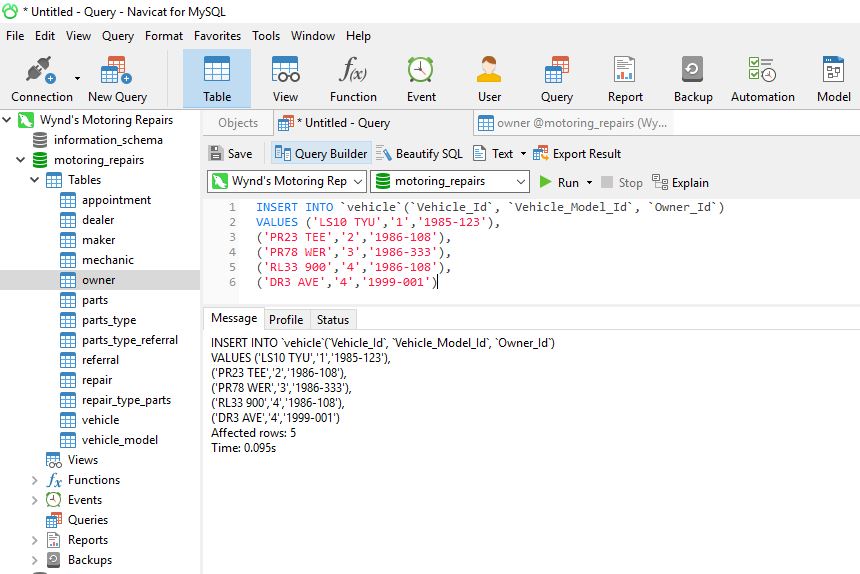
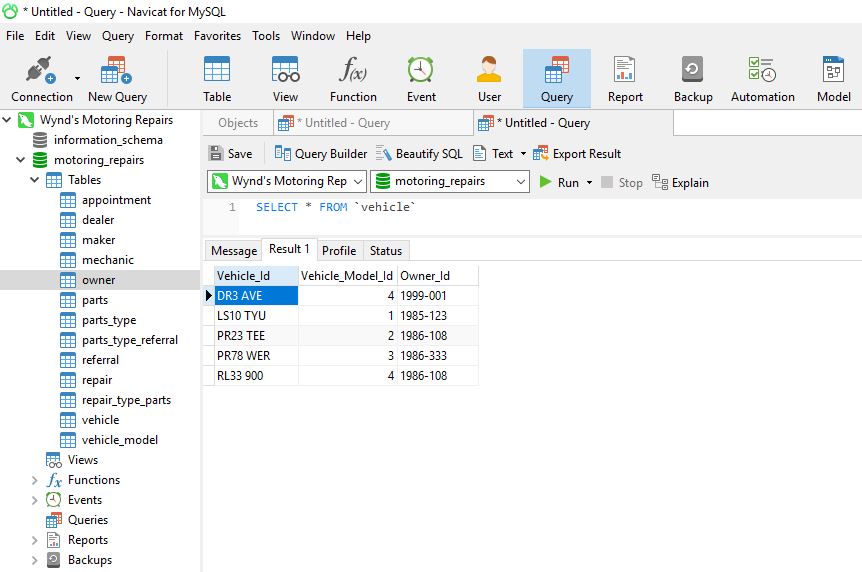
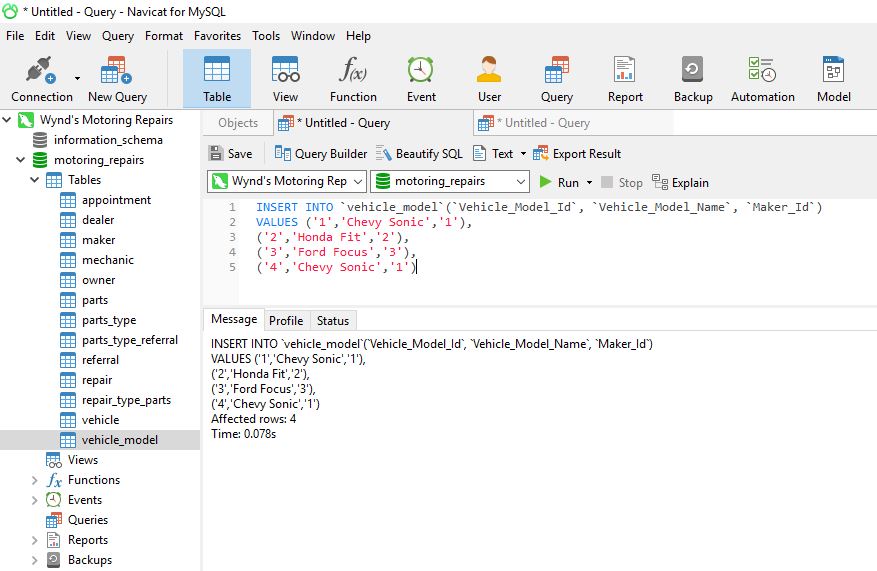


Figure-3

 Figure-4

Figure-5 Figure-6

Figure-7

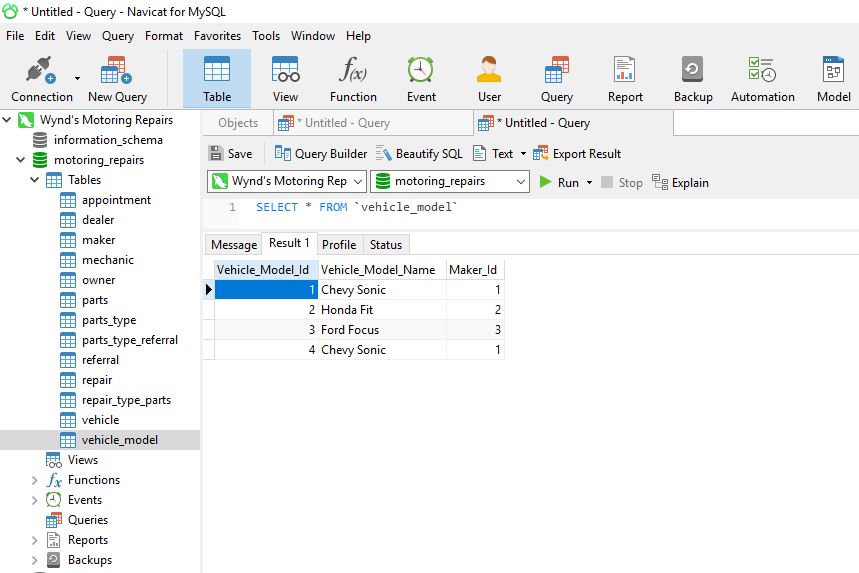


Figure-8

1. All Mechanics data insert and show in figure 9 to 10.

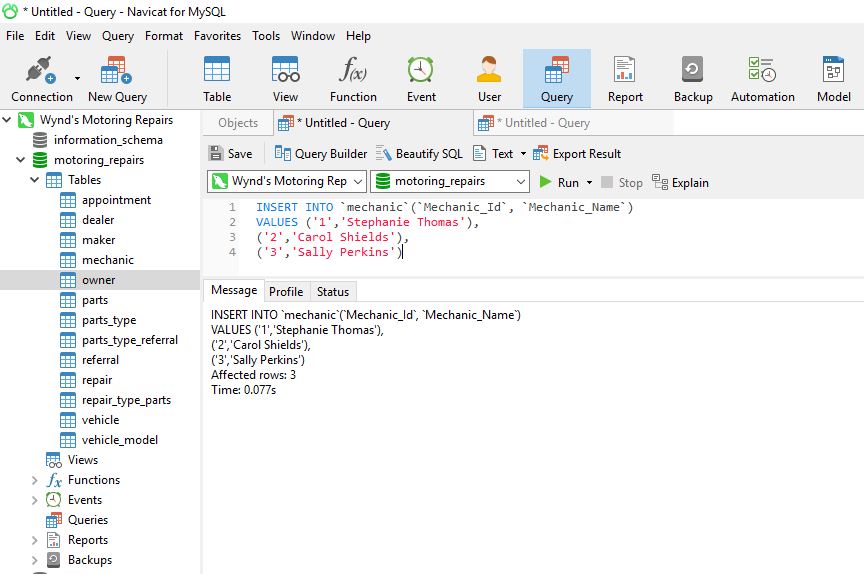
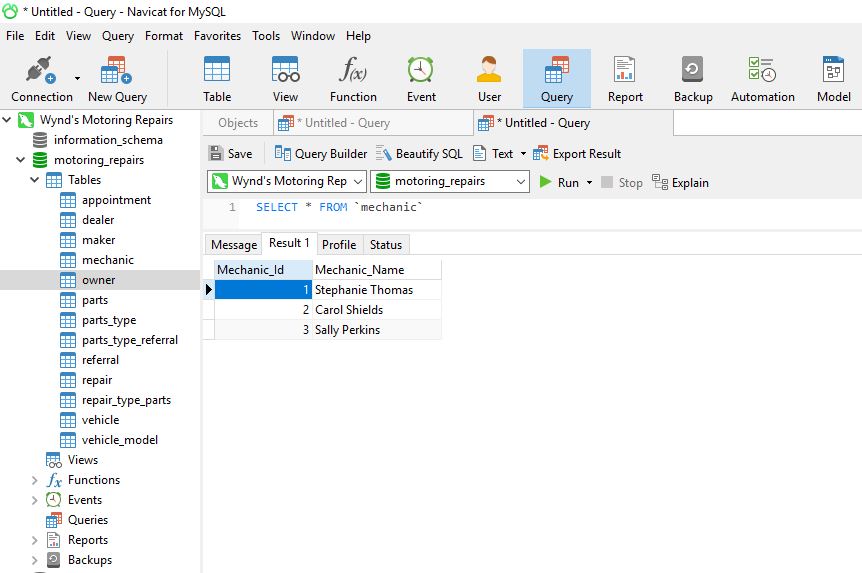
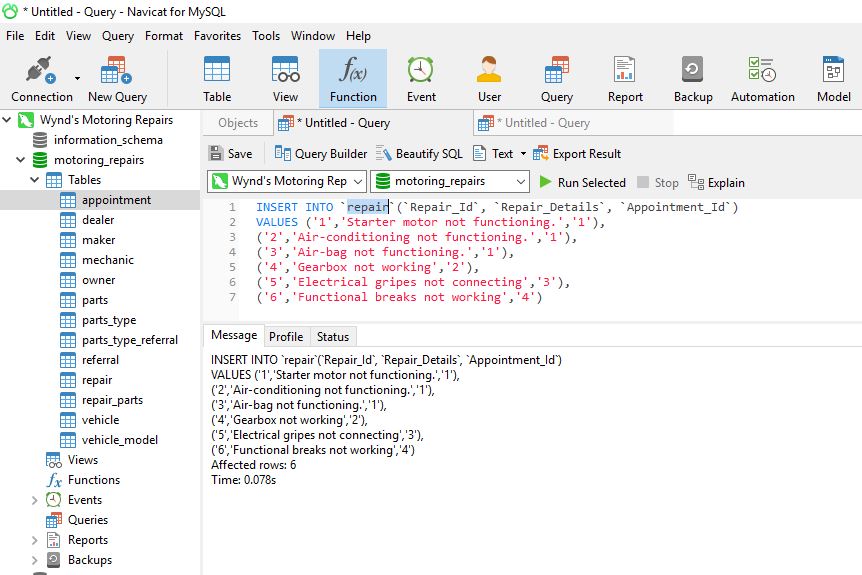
Figure-9. 

Figure-10

1. Appointments, repairs and referrals with all table data insert and show in figure 11 to 26 for Appointments.

Figure-11

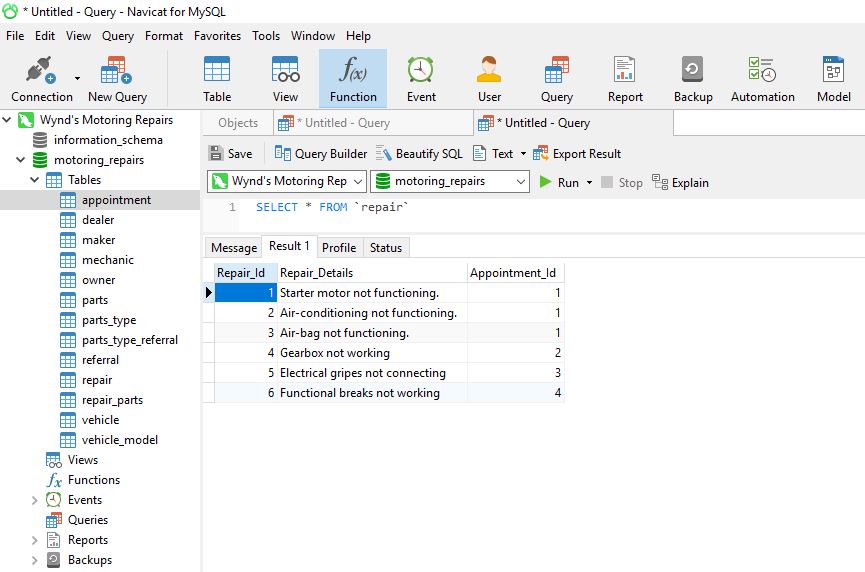
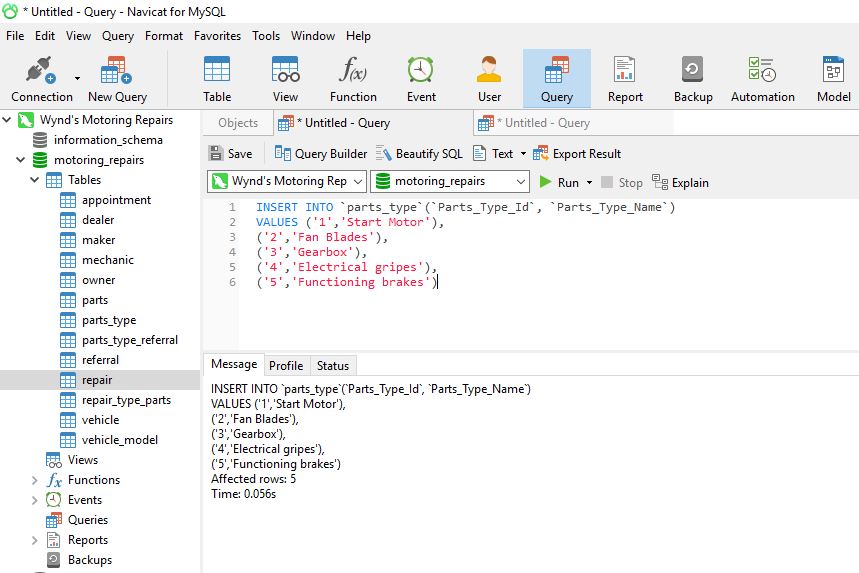
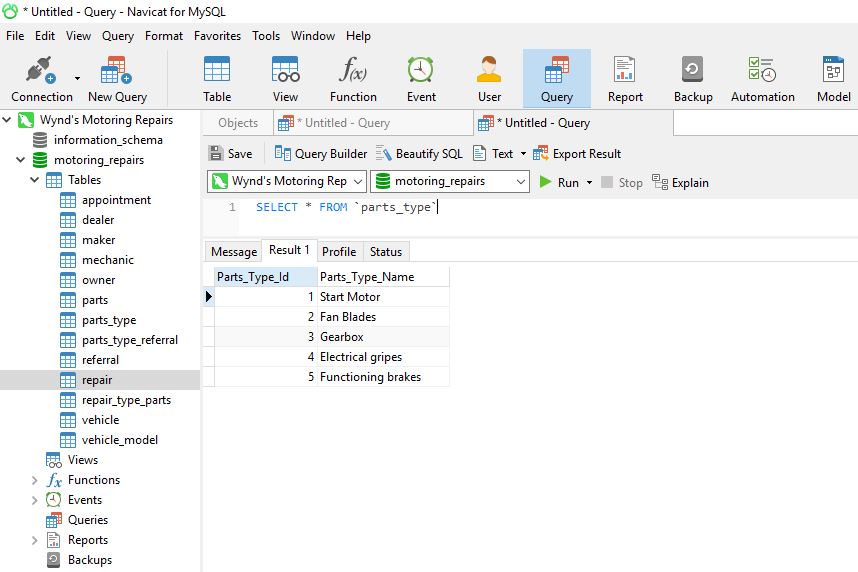
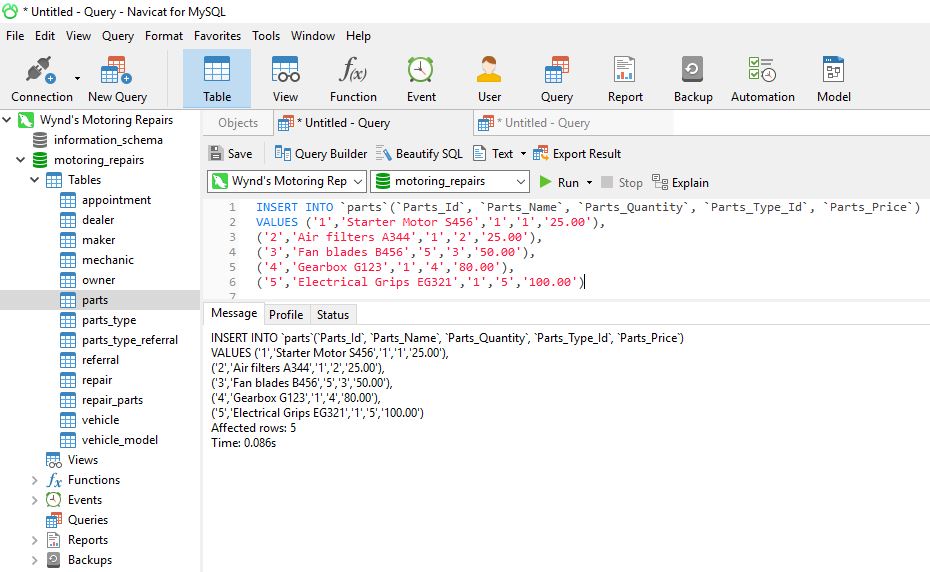
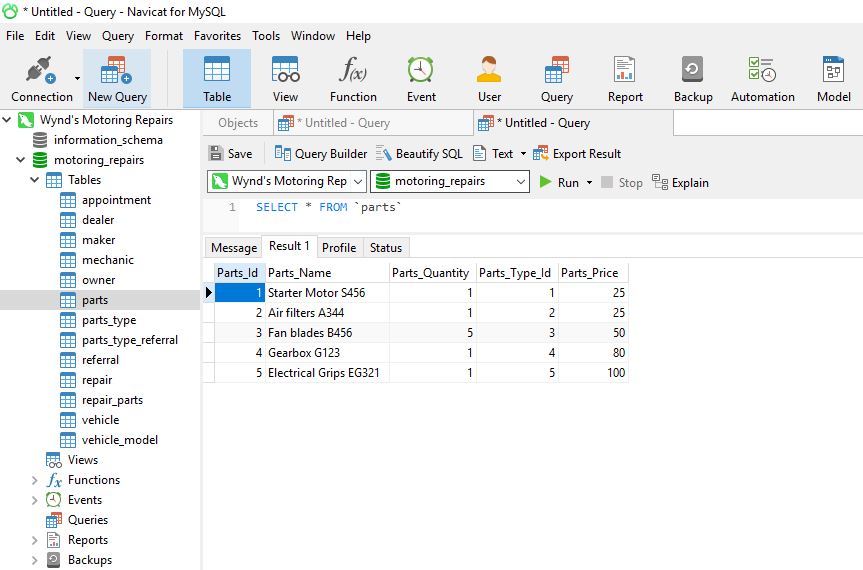
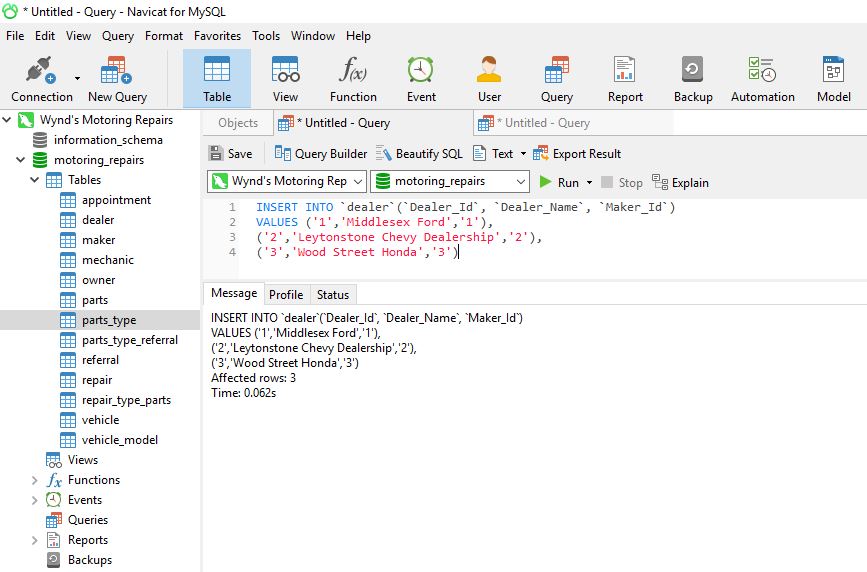


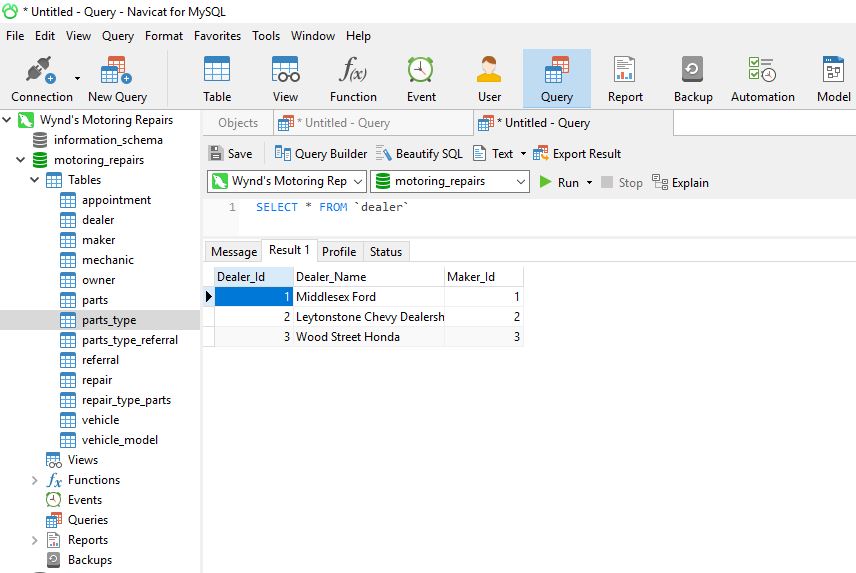
Figure-12

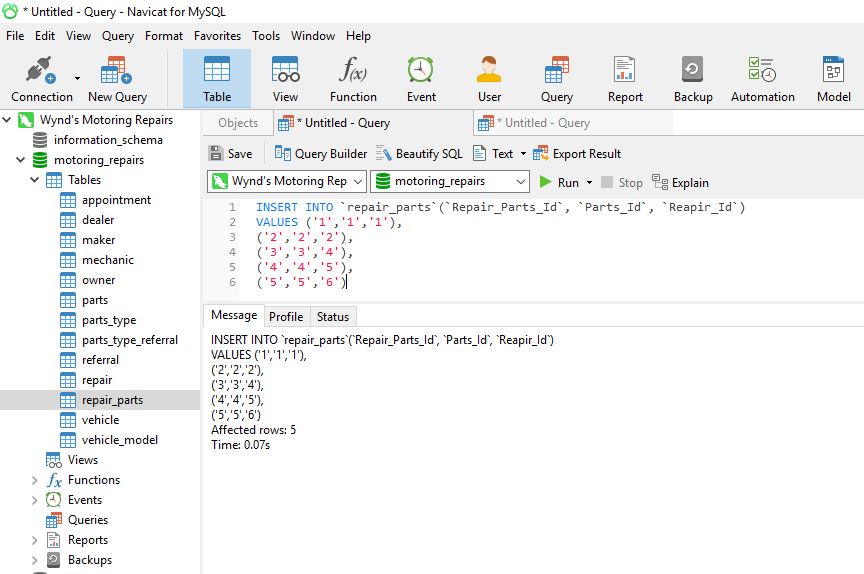
 Figure-13

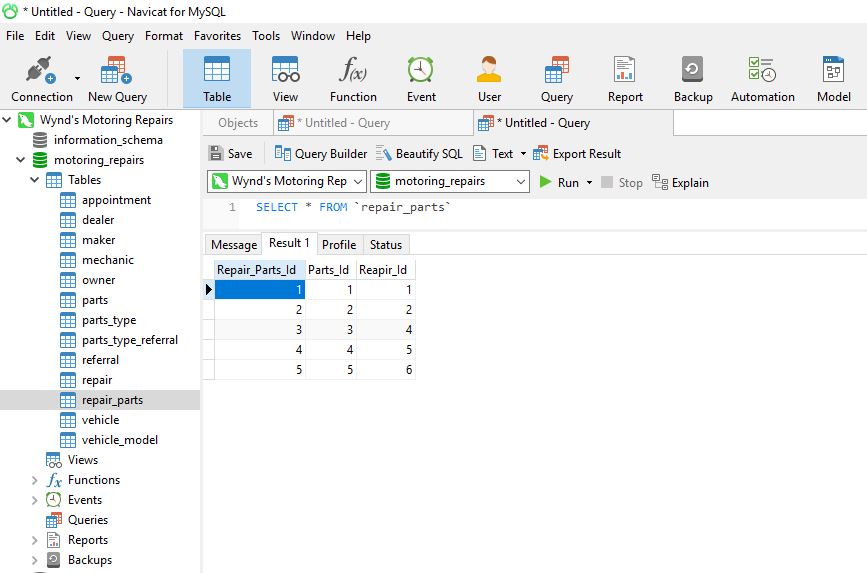
Figure-14

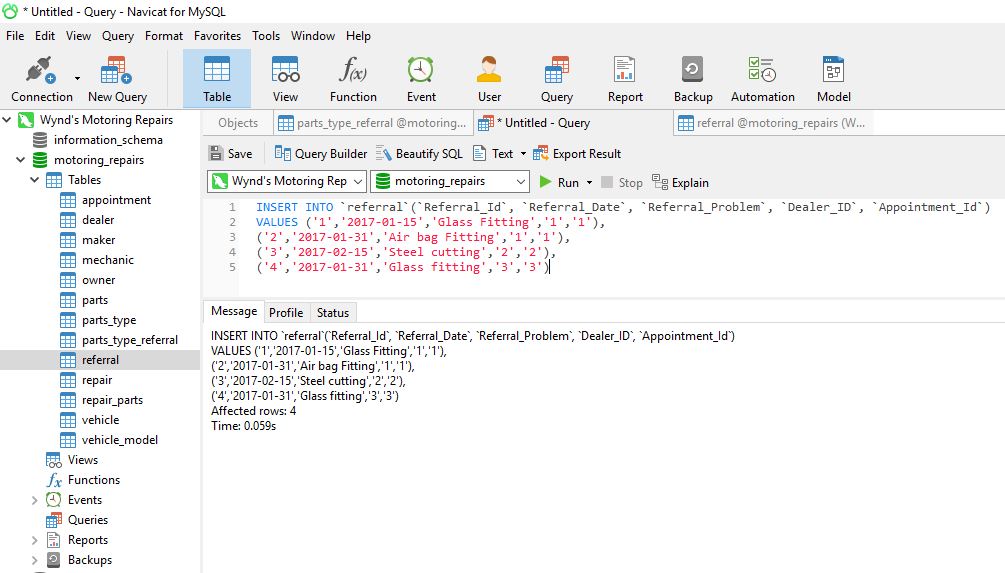
 Figure-15 Figure-16

Figure-17

 Figure-18

Figure-19

Figure-20

Figure-21

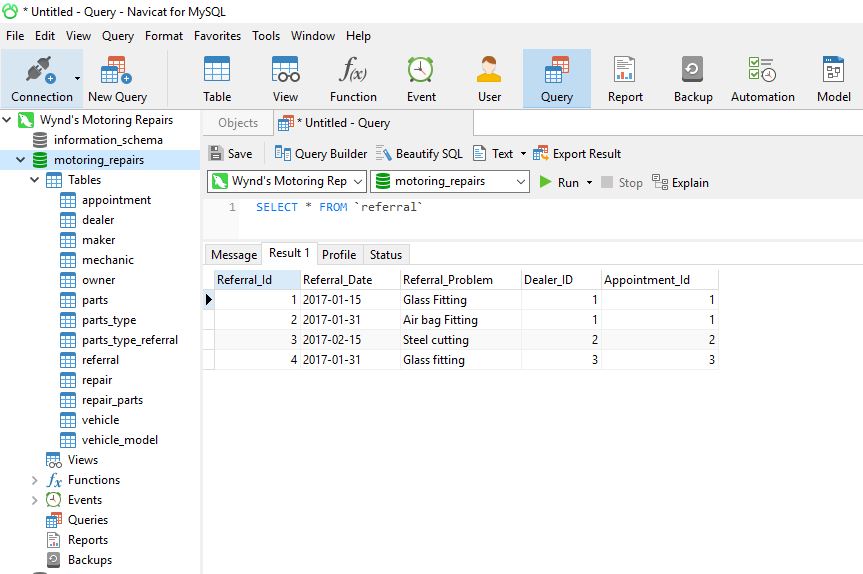
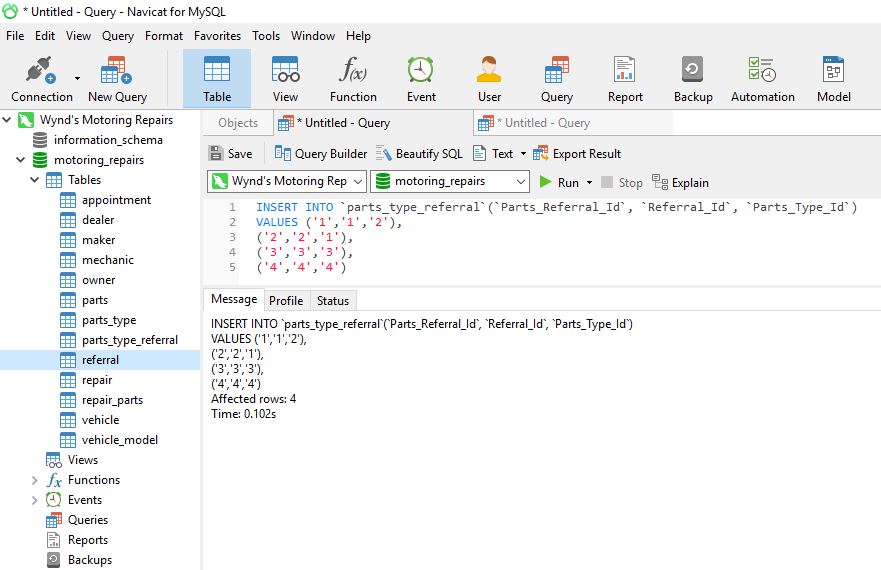
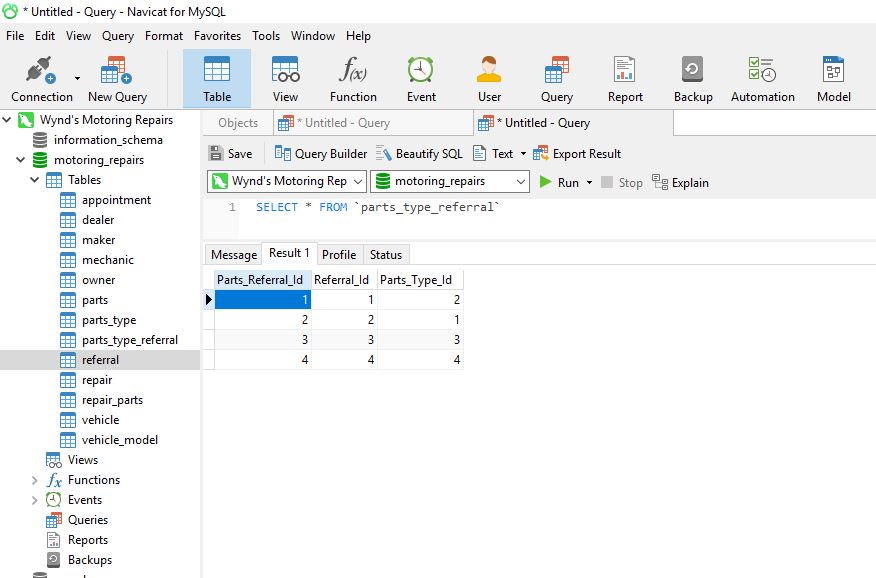
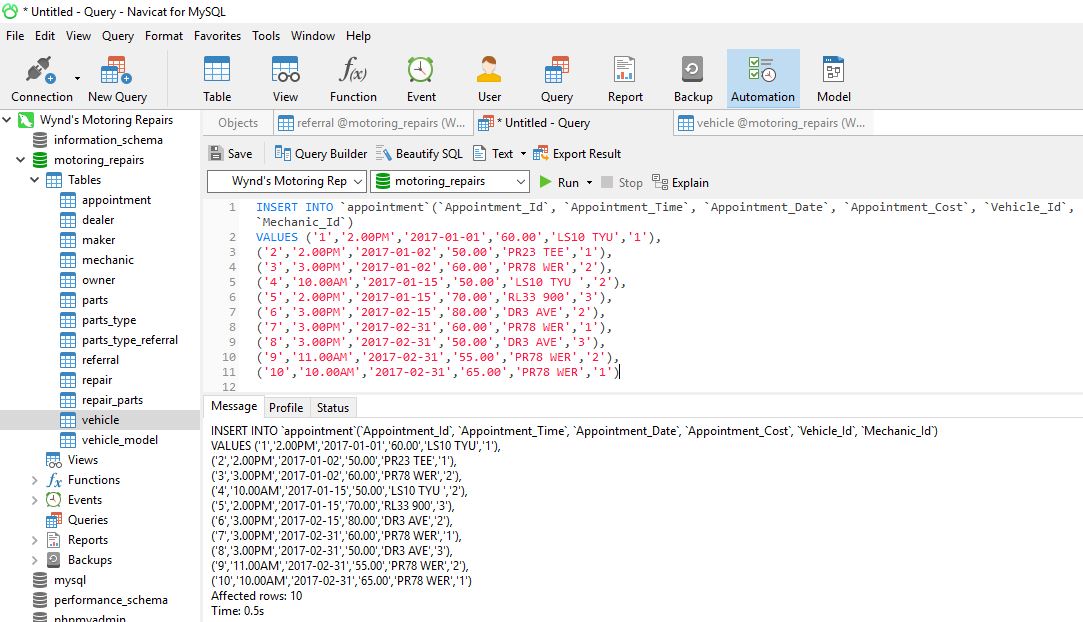
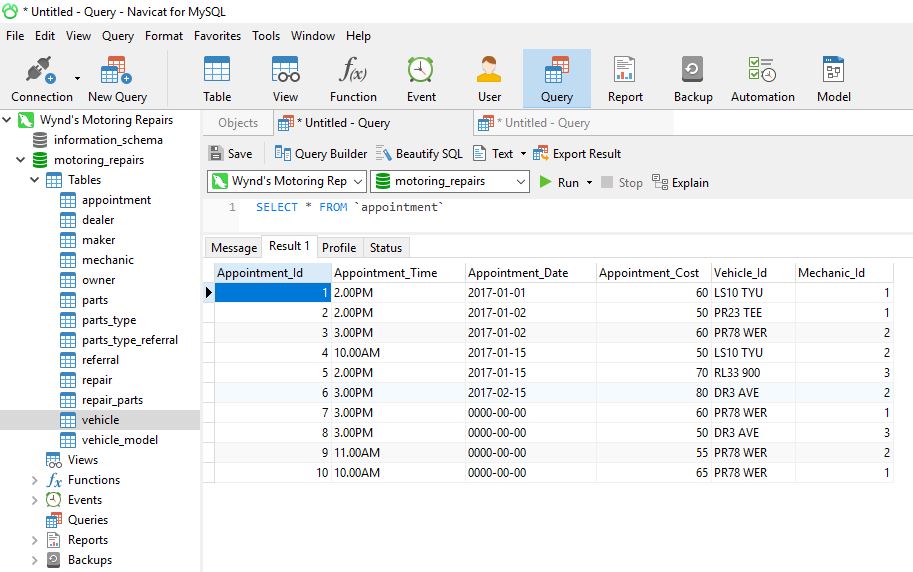


Figure-22

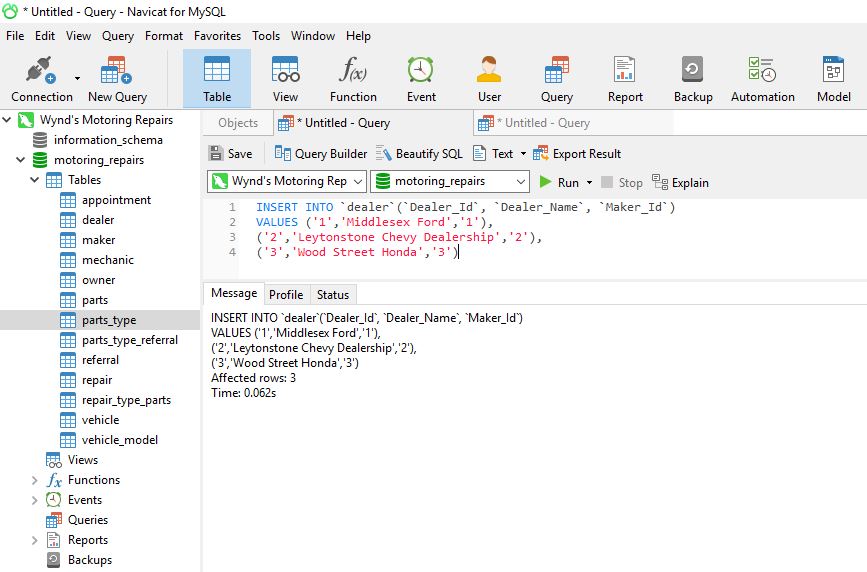
Figure-23

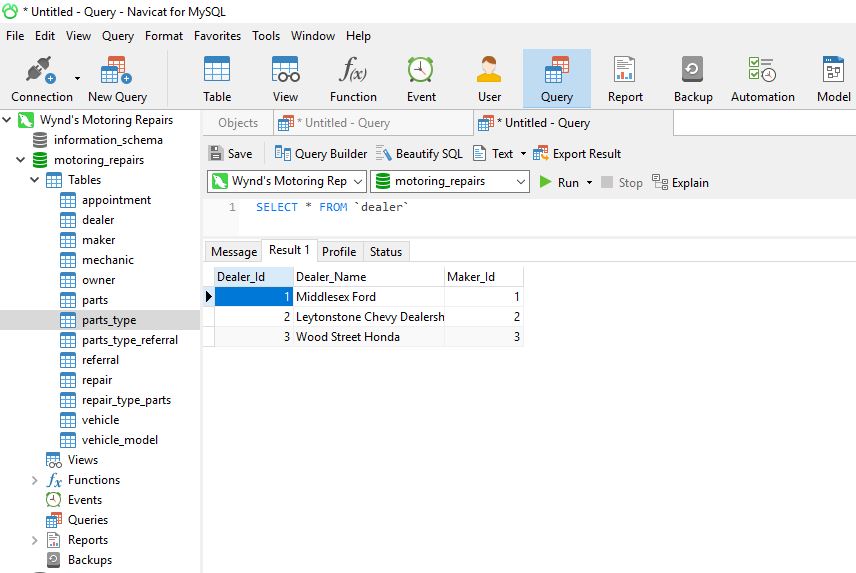
Figure-24

Figure-25

Figure-26

1. Specialist dealers data insert and show figure 27 to 28.

Figure-27

Figure-28

1. Select all owner data.

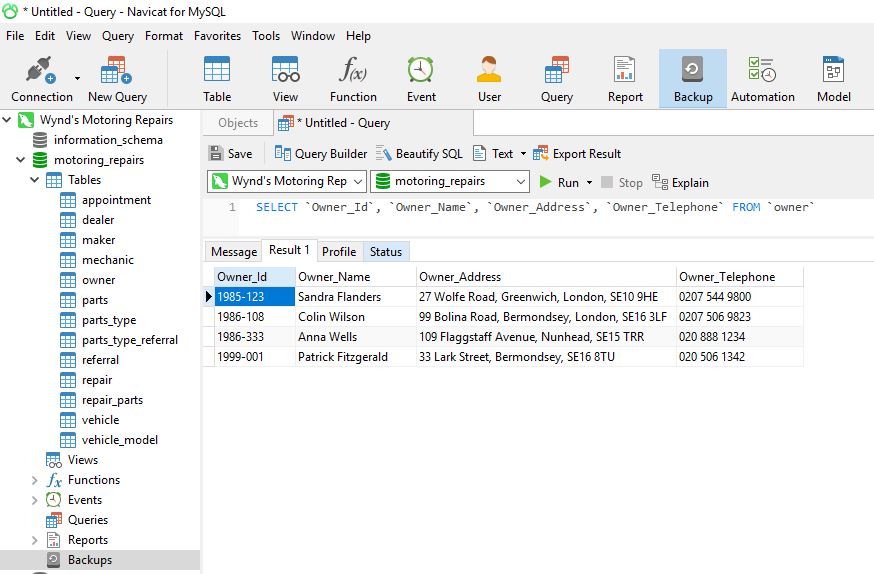
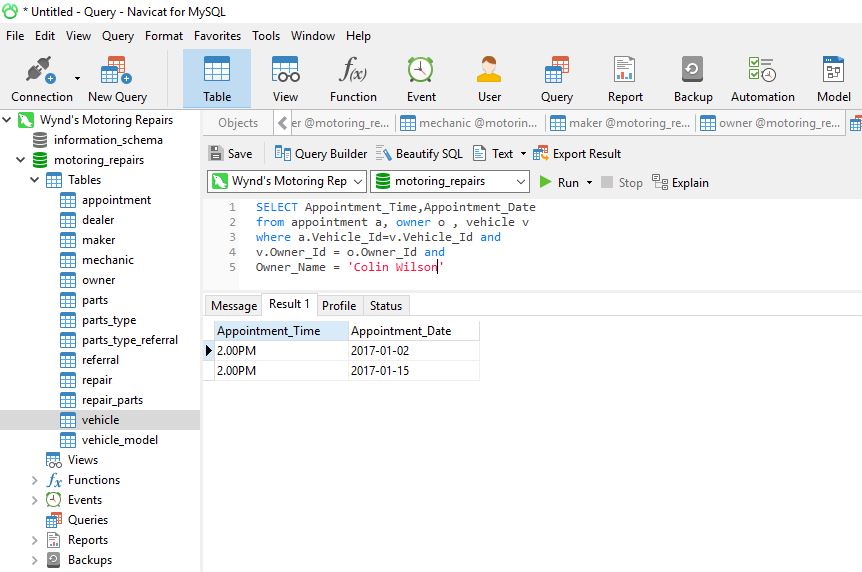
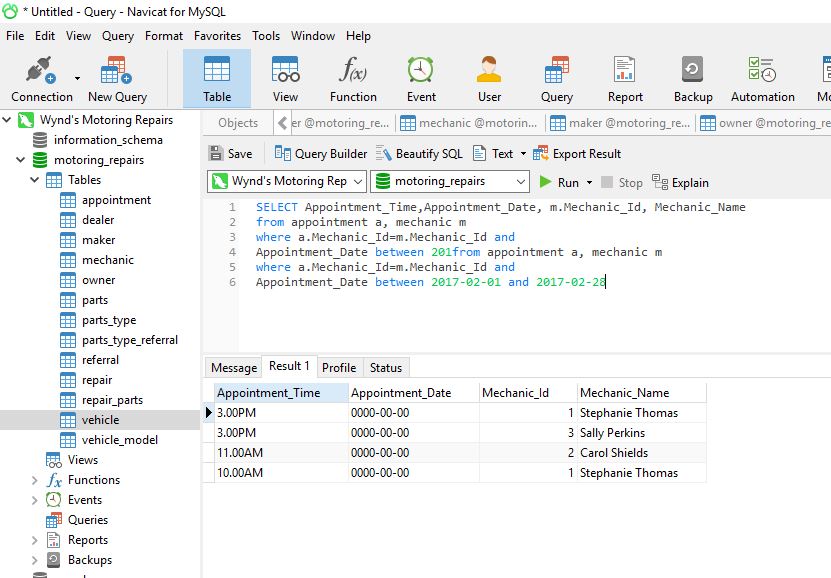
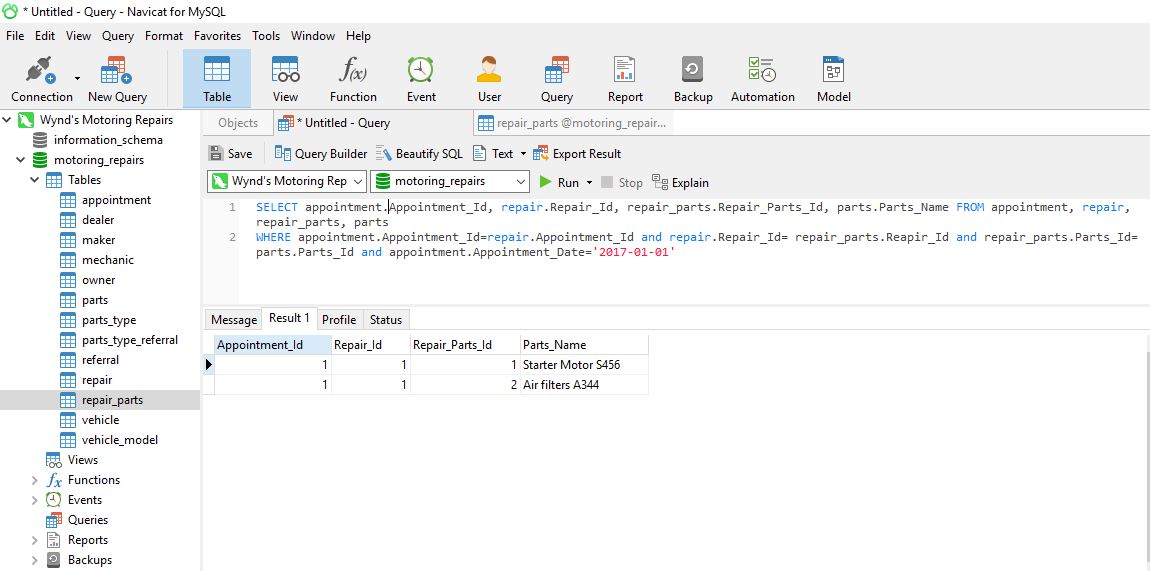
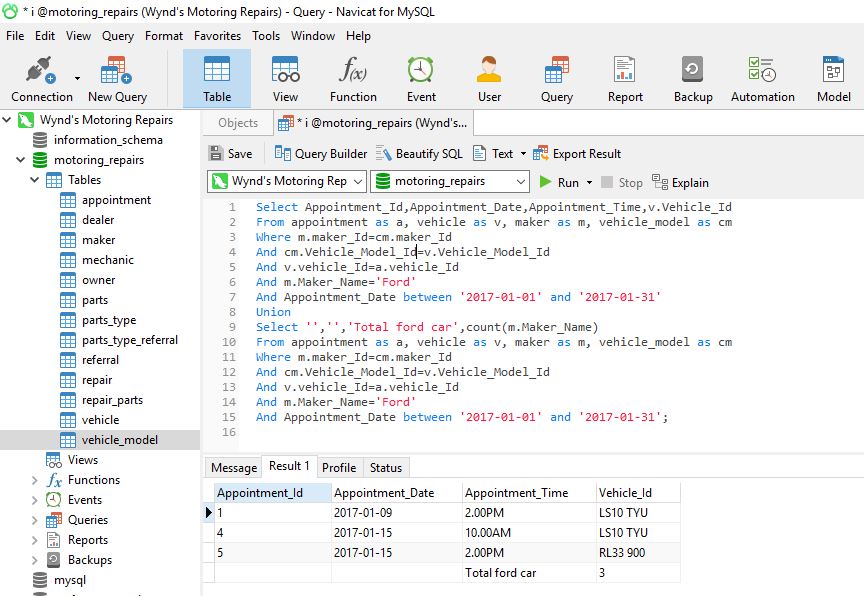
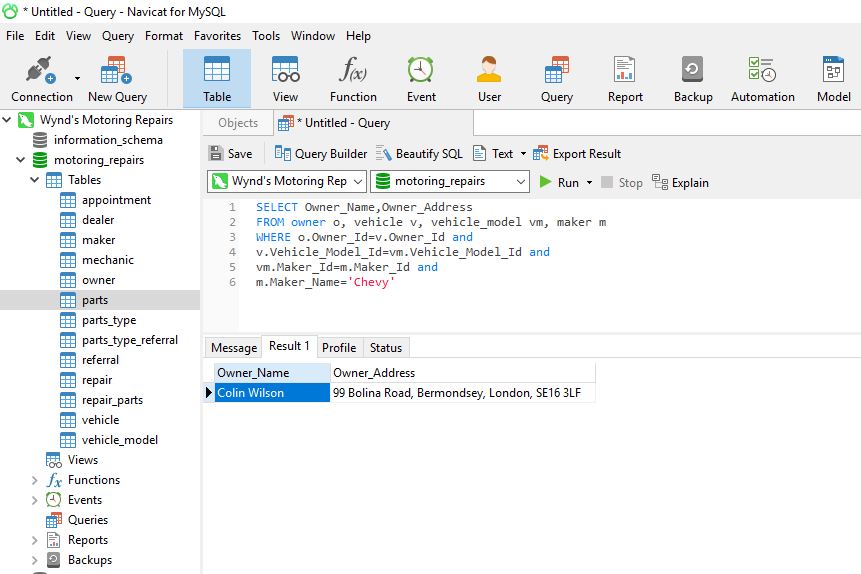
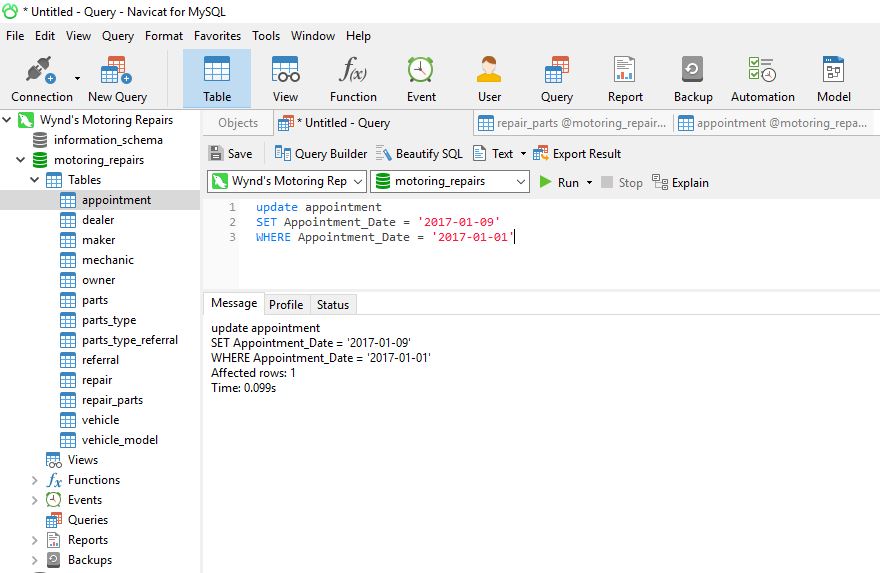


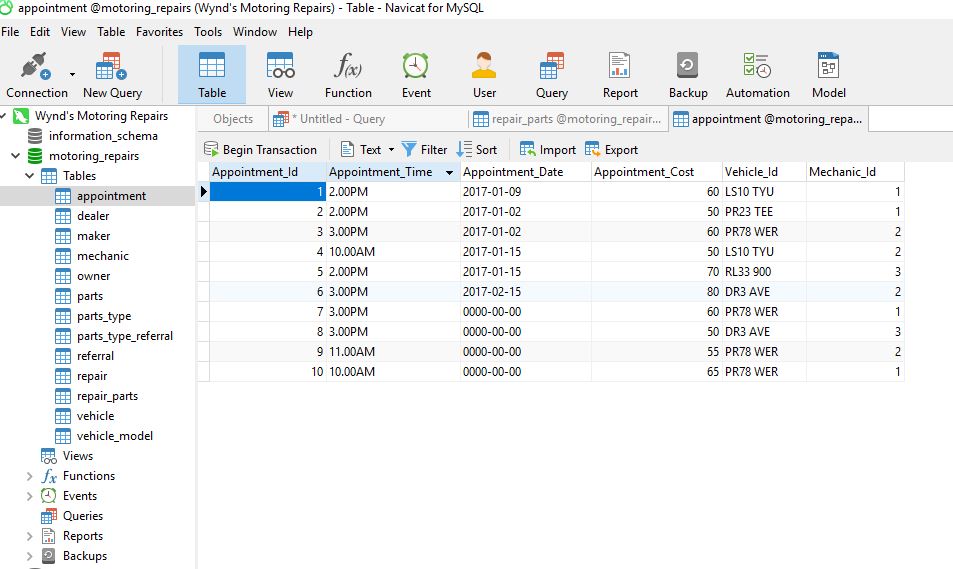
Figure-29

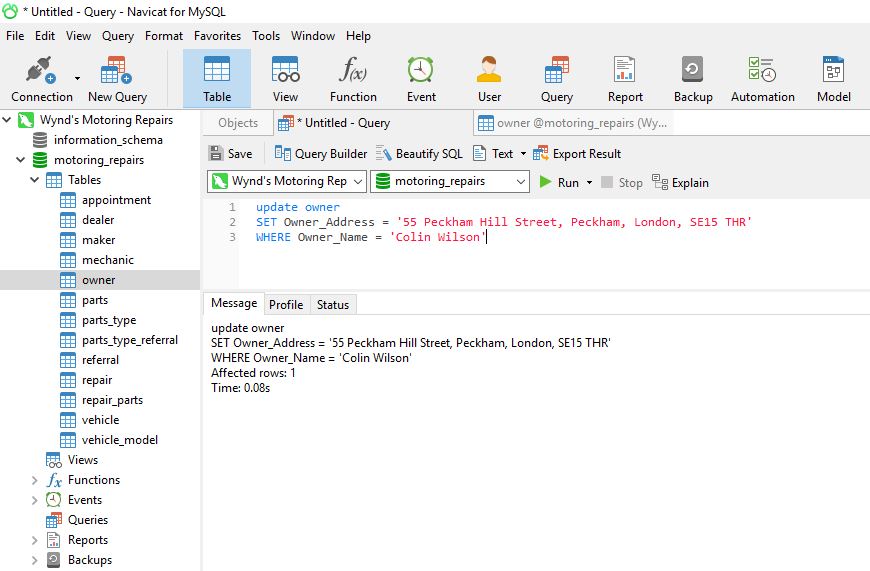
1. Selects dates and times for the appointments for vehicles of Colin Wilson.
2. Selects all dates, times and mechanics names for appointments in February 2017.
3. Shows all the parts used in appointments on 1 January 2017.
4. Counts all the Ford cars with appointments in January 2017. 
5. Shows all owners’ names and addresses for those owners who have cars made by Chevy.

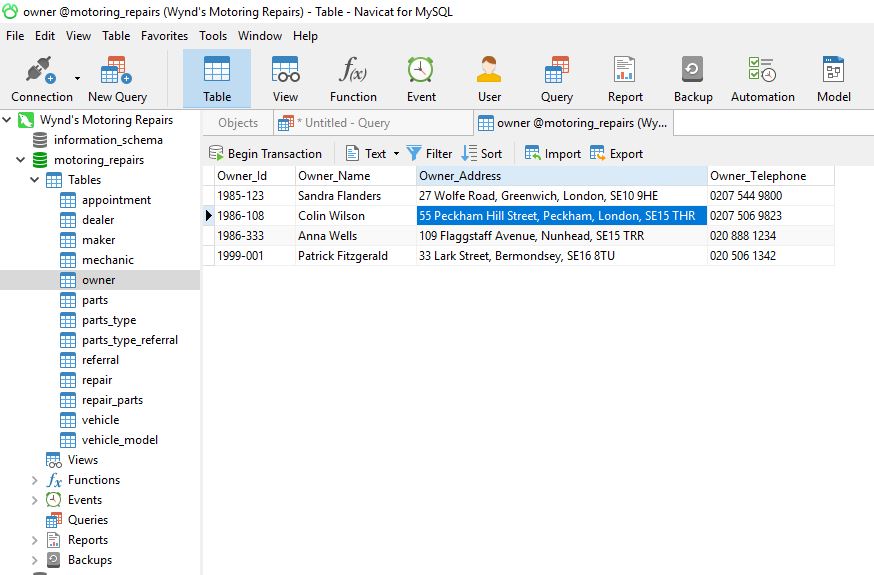


1. Update the appointments so that all appointments on 1 January 2017 are now on 9 January 2017 at the same times.



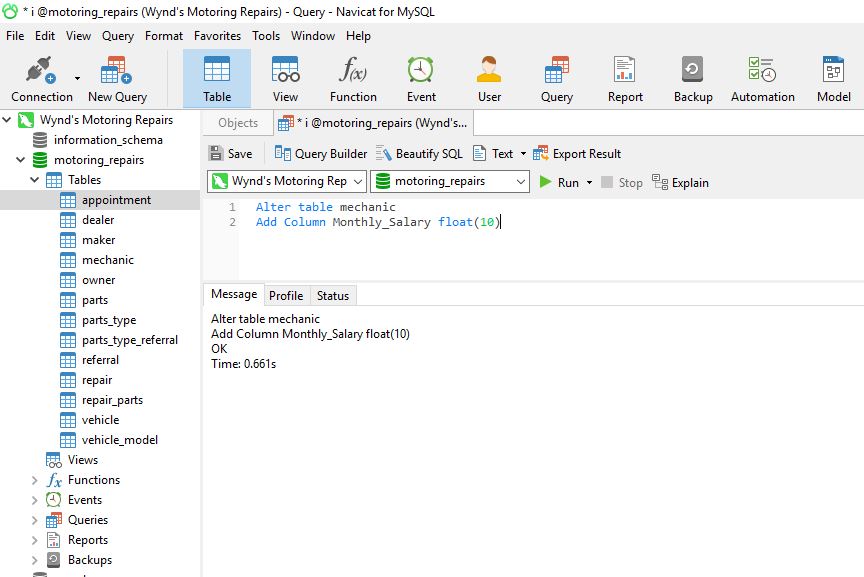


1. Update Colin Wilson’s address.

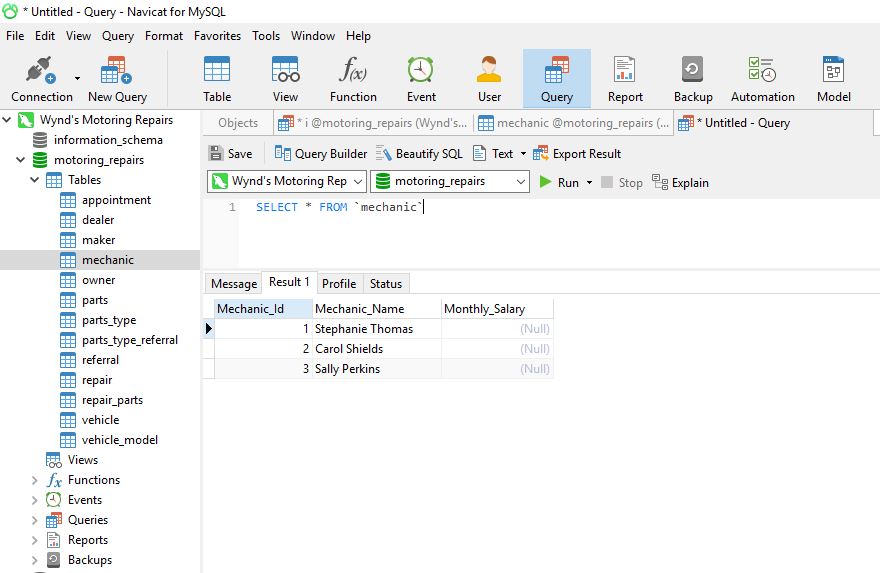


Task-3

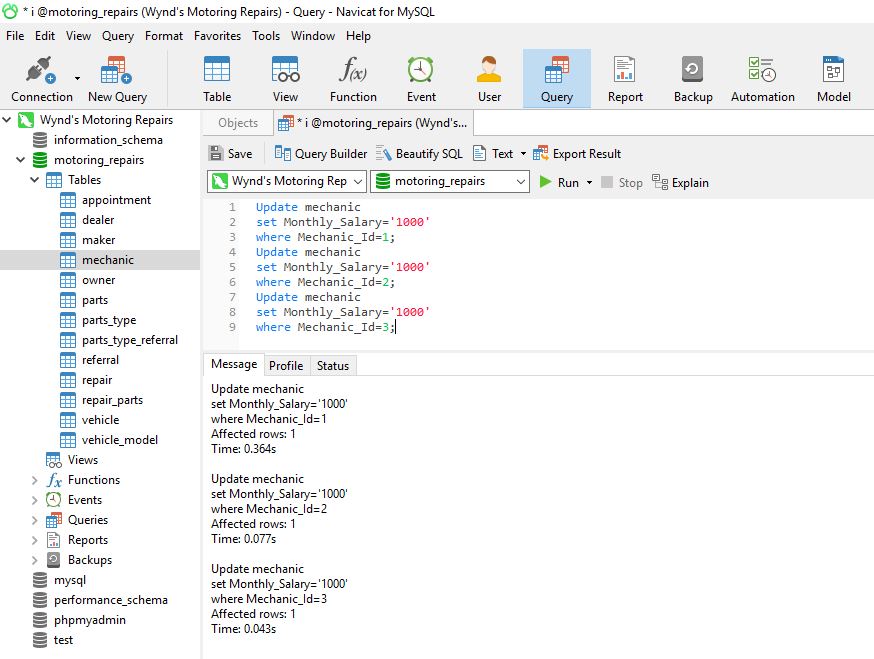
Derived Data function and query.



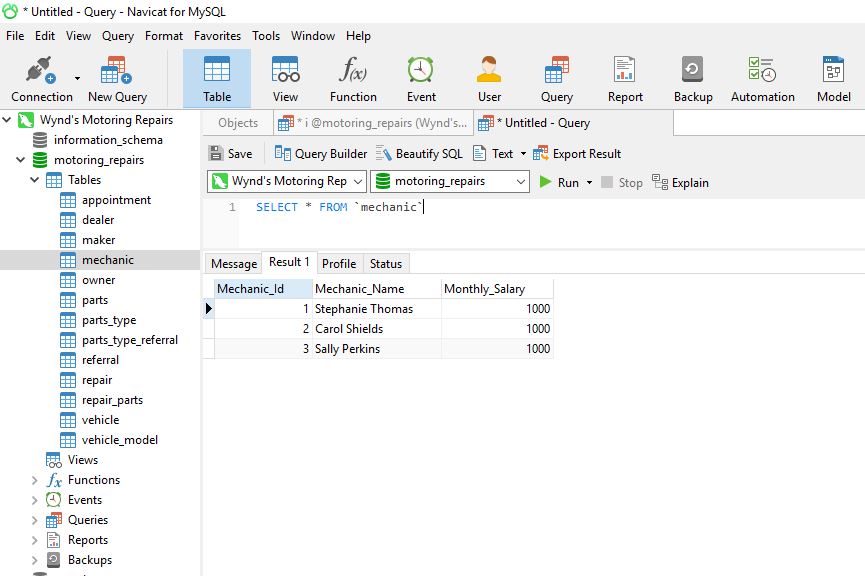
Add column Monthly\_Salary.



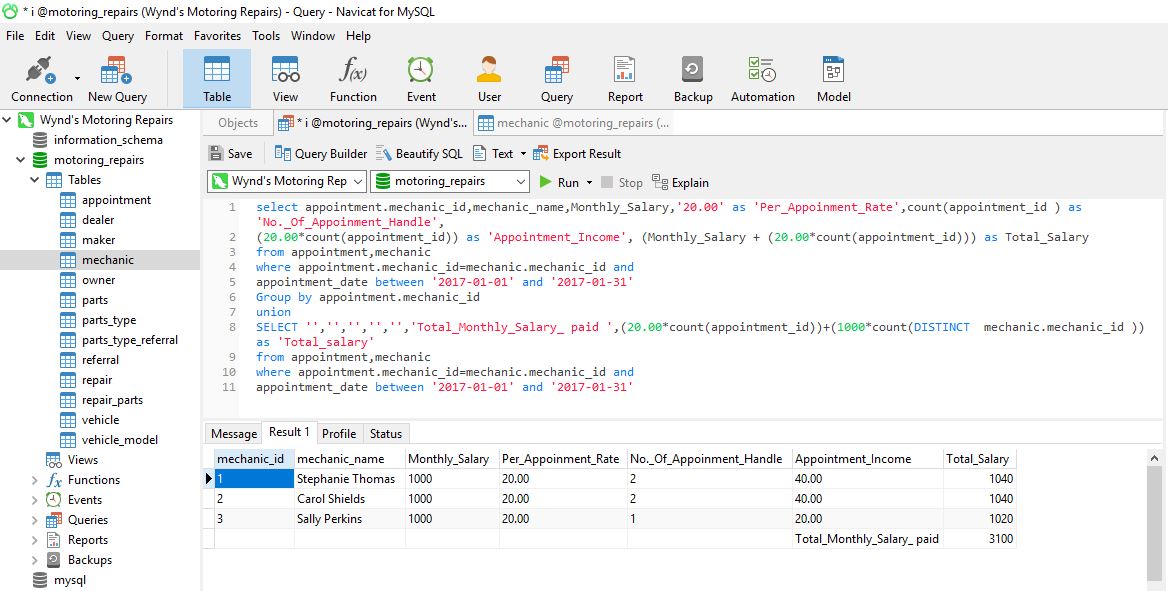
Show Monthly\_Salary field add.



Update Mechanic monthly salary field data.



Mechanic table all data shown.



Mechanic table total salary derived Data query and result.

Count: Appointment table unique data select for used count.

Union: Two or more select statement for used union.

Group by: One or more columns result set for used group by function with sum function.

**Task-4**

Winds Motoring Repairs future business enhances current database system must be implemented on Distributed Database Management System (DDBMS). So, their business practices, organisational structure and technological developments how changes are discussed on there.

1. **Business Practices:** Every branch data must be stored and manage individually in their local database system. Every branch data can send their central database system with data security at the end of the business hour for backup purpose. A single branch data can’t be accessed directly another branch data but any information is needed it can be accessed from the central database with data security.
2. **Organisational Structure:** A main office that has a central database system besides every branch must be connected to the central database system. A single branch must store their local data on daily basis in a central database system.
3. **Technological Developments:** A main office database system and hardware component must be more powerful and securable from another branch database system. The branch label database system don’t need more powerful from central database system because central database system store more data from the branch label database system.

Task-5

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Original Requirements** | **Initial Requirements** | **Overall Assessments** |
| 1 | All data can be stored in central based database system. | Entity Relationship Model processes data store in one place. | Database system store properly with using link entity system. |
| 2 | Data duplicity must be reducing and avoided. | Normalization processes Many to many relationship breakdowns to create one to many relationships. | Ensured that reduce data duplicity. |
| 3 | Data organized properly with standardization. | Primary key used for uniquely identified data and foreign key used for share and access of data. | All the data is organized identify in standard way. |
| 4 | New data can be inserting and query can be displayed. | Data insert more than their given data for displayed more query result. | Some attributes are required for uniquely identification like Appointment ID, Mechanic Id etc. |
| 5 | They want to monthly salary report for their staff. | To create their salary report for used derive function design a monthly salary sheet. | Monthly salary sheet show all of data with individual and totally. |
| 6 | Wynd’s Motoring Repairs wants to all mechanic full salary report with final calculation. | Derive data concept with aggregate function implemented on hare. | So, the salary report was generated successfully. |

Conclusion: At last it can be said that, Database Design and Development produce instruction on how to design and development more reliable and robust database system in business development processes.