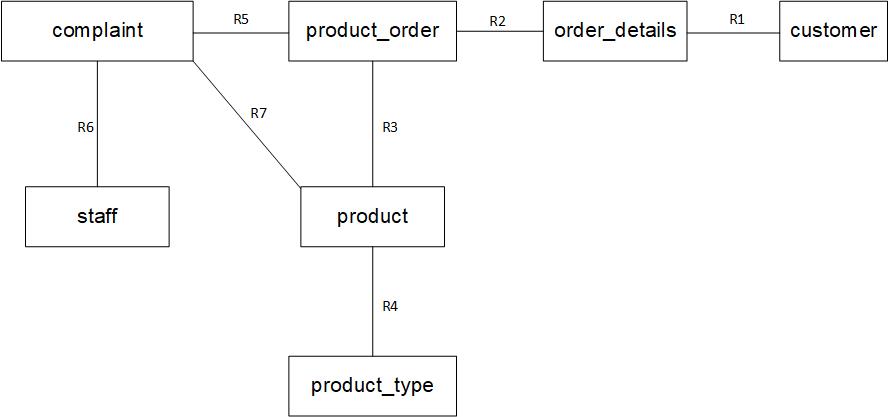
Introduction:

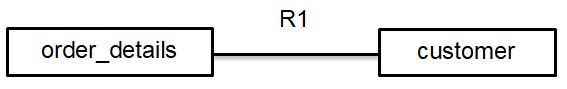
Plato Toys is a toy manufacturer and retailer they want to sell their products online. The company wants design and implements a database that meets the requirements. The purpose of this assessment is to provide information about how a normalized database has been designed and implemented to meet the requirements for Plato toys Manufacturer Company. The assignment also includes a task based assessment that provides discussion on how the work has been done to complete the requirements of this Company.

Task-1

1. Entity-Relationship model for Plato Toys database system.

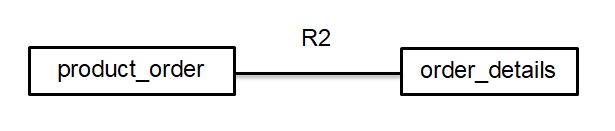


**Entity-Relationship model description**



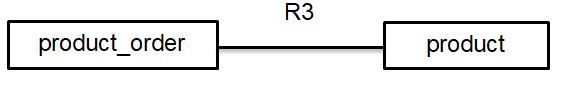
**Relationship type:** One to many

**Relationship description:** One customer can have one or more order and one order must have one customer.



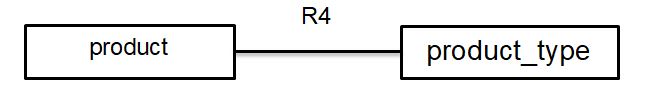
**Relationship type:** One to many

**Relationship description:** One or more product order can have one order and one product order must have one order.



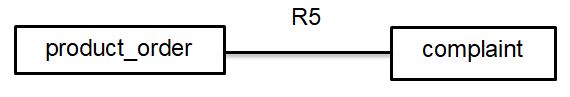
**Relationship type:** One to many

**Relationship description:** One or more product order can have one product and one product must have product order.



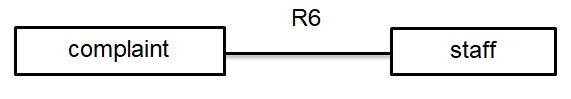
**Relationship type:** One to many

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



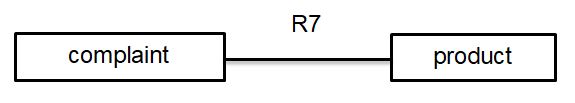
**Relationship type:** One to many

**Relationship description:** One or more product complaint can have one product order and one complaint must have one product order.



**Relationship type:** One to many

**Relationship description:** One or more complaint can have one staff and one complaint must have one staff.



**Relationship type:** One to many

**Relationship description:** One or more complaint can have one product and one complaint must have one product.

1. Relation schema ER model from identify primary and foreign key.

customer

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Customer\_Id | varchar | 10 | Yes |  |  |
| Customer\_Name | varchar | 50 |  |  |  |
| Customer\_Address | varchar | 100 |  |  |  |
| Customer\_Phone | varchar | 15 |  |  |  |

Order\_details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Order\_Id | varchar | 10 | Yes |  |  |
| Order\_Date | varchar | 10 |  |  |  |
| Order\_Total | float | 5 |  |  |  |
| Customer\_Id | varchar | 10 |  | Yes | customer |

staff

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Staff\_Id | varchar | 5 | Yes |  |  |
| Staff\_Name | varchar | 25 |  |  |  |

product\_type

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Product\_Type\_Id | int | 3 | Yes |  |  |
| Product\_Type\_Name | varchar | 100 |  |  |  |
| Product\_Type\_Quantity | int | 5 |  |  |  |

product

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Product\_Id | varchar | 6 | Yes |  |  |
| Product\_Name | varchar | 20 |  |  |  |
| Product\_Type\_Id | int | 3 |  | Yes | product\_type |
| Product\_Type\_Price | float | 5 |  |  |  |
| Product\_Type\_Totalprice | float | 5 |  |  |  |

product\_order

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Product\_Order\_Id | int | 5 | Yes |  |  |
| Order\_Id | varchar | 10 |  | Yes | order\_details |
| Product\_Id | varchar | 6 |  | Yes | product |

complaint

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes | Data Type | Length | Primary Key | Foreign Key | Reference |
| Complaint\_Id | varchar | 10 | Yes |  |  |
| Complaint\_Description | varchar | 255 |  |  |  |
| Complaint\_Solution | varchar | 255 |  |  |  |
| Product\_Id | varchar | 6 |  | Yes | product |
| Product\_Order\_Id | int | 5 |  | Yes | product\_order |
| Staff\_Id | varchar | 5 |  | Yes | staff |

1. Database management system many to many relationships in relational schema database design can’t do it. So, many to many relationship breakdowns to create a one to many relationships in entity relationship model.

**Data integrity:** In the database data store system accuracy, consistency and reliability maintain for predefined set of rules, as determined by the database administrator or application developer. Error checking and validation routines use a database when it is designed and authenticated.

**Example:** Maintained whole database system data integrity maintains database numeric columns where has unique value or primary key in database system.

**Referential integrity:** In relational database concept relational table must always be consistent with the primary key where reference table foreign key has. For the database table organize they contain information. So, database concept primary key and foreign key used for database relationship maintain between the tables.

**Example:** Relational database system each table has one primary key which can be reference on another table primary key against another attribute. It’s called a foreign key.

**Functional dependence:** Functional dependency is maintained one table to another table relationship that exists one attribute uniquely determines another attribute. A database server as a constraint between two sets of attributes which is important part of the relational database design aspect normalization.

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

Task-2

Data Entry and Data Manipulation for Plato Toys database system.

1. Table and database create.

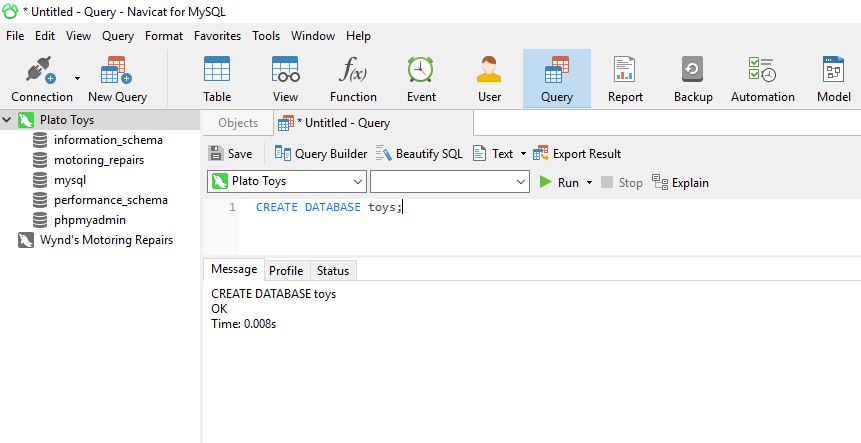


Figure-1: Create database.

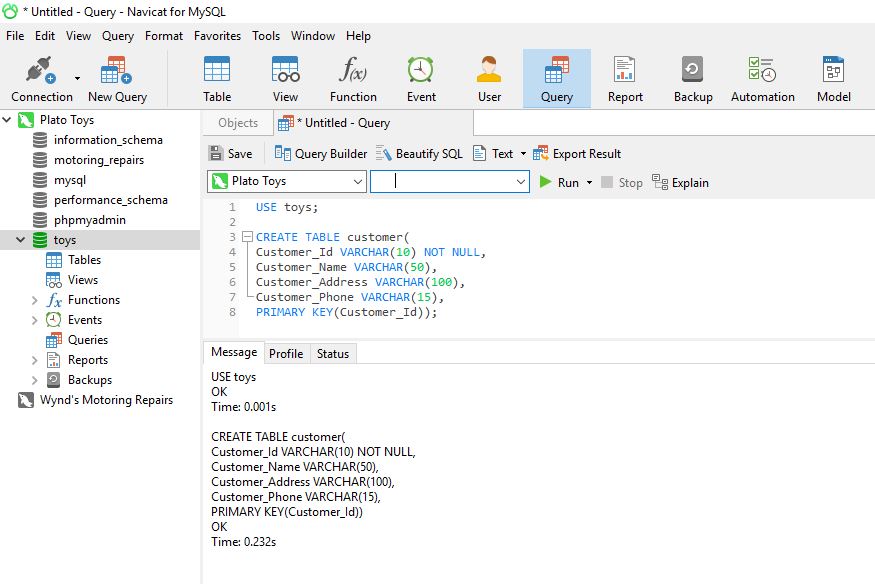


Figure-2: Create customer table.

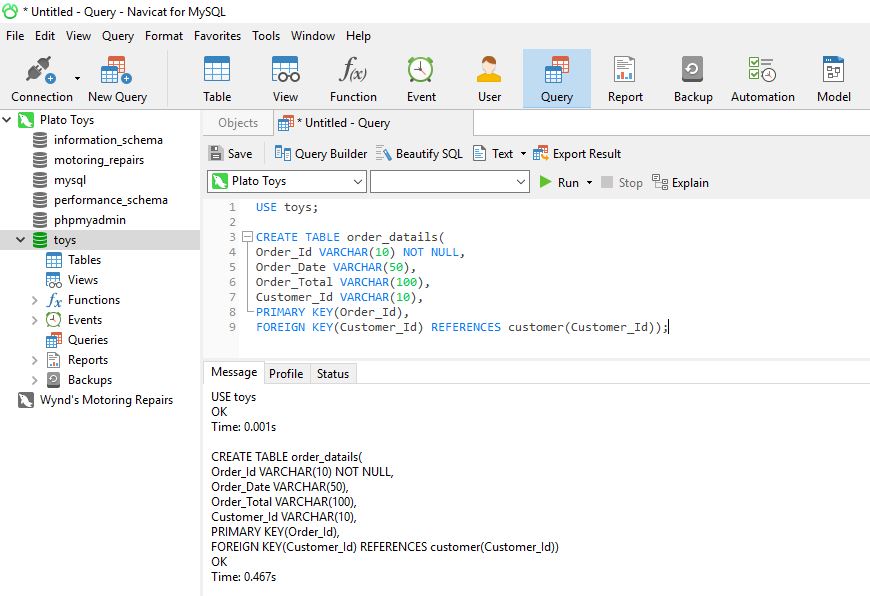


Figure-3: Create order details table.

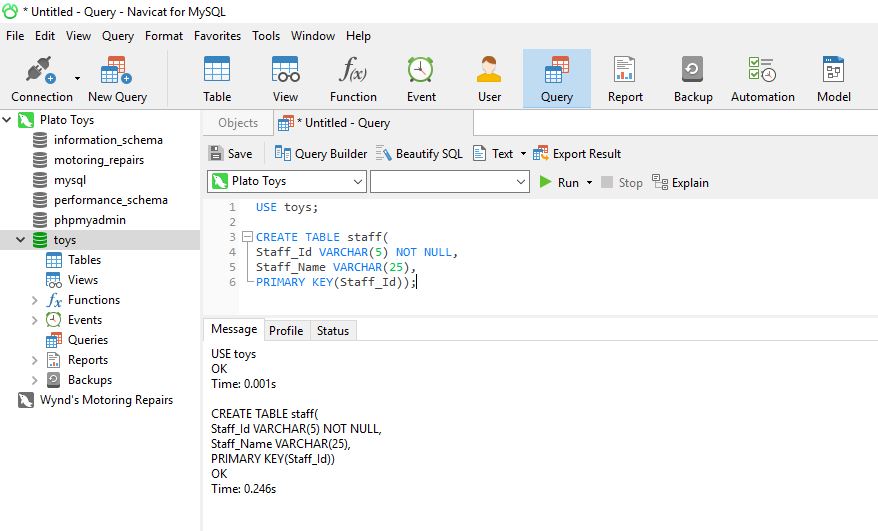


Figure-4: Create staff table.

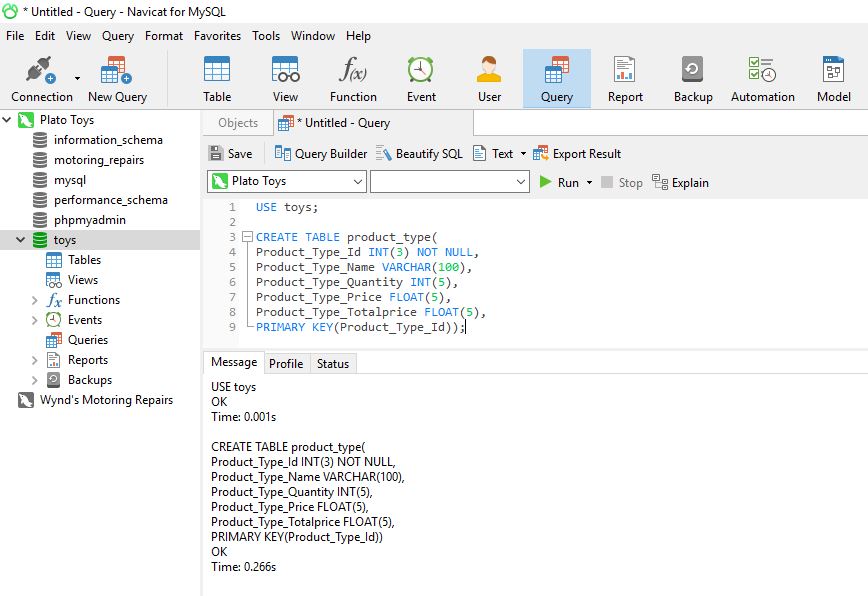


Figure-5: Create product type table.

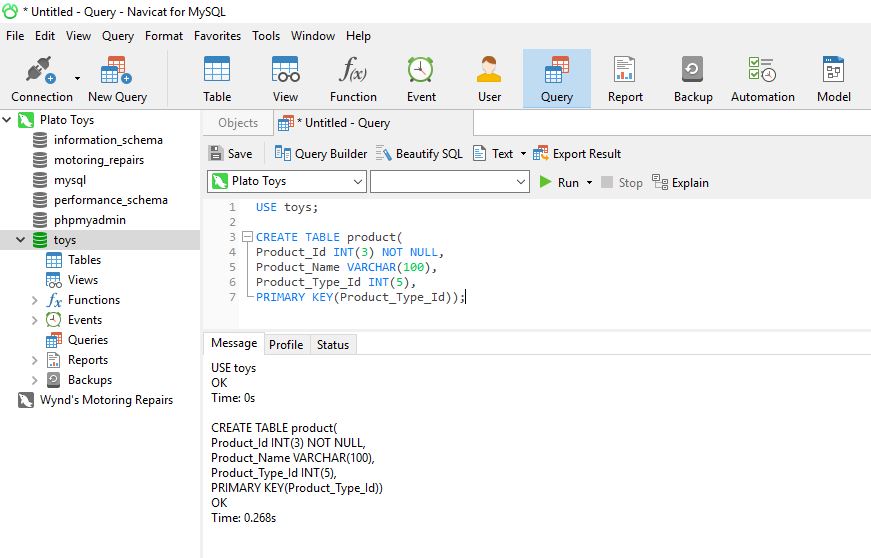


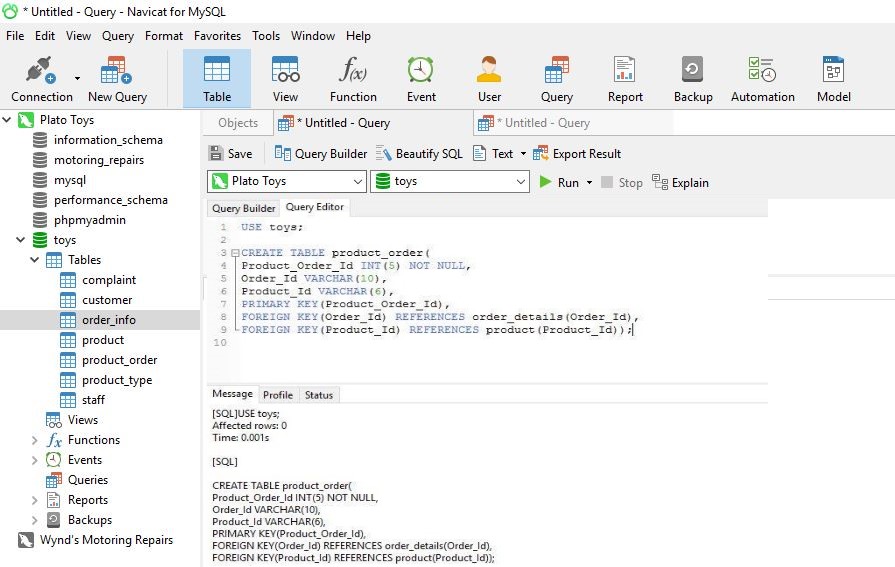
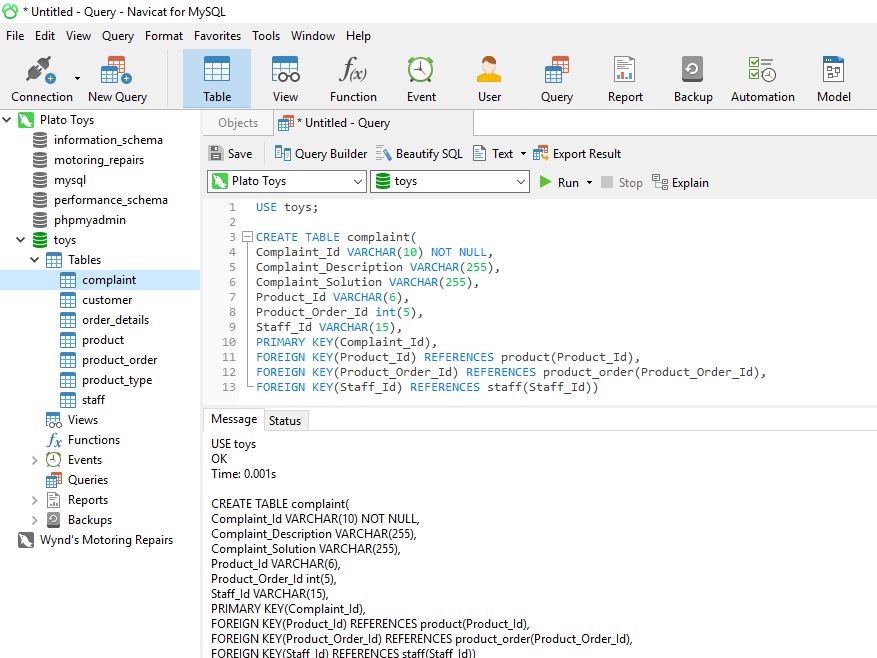
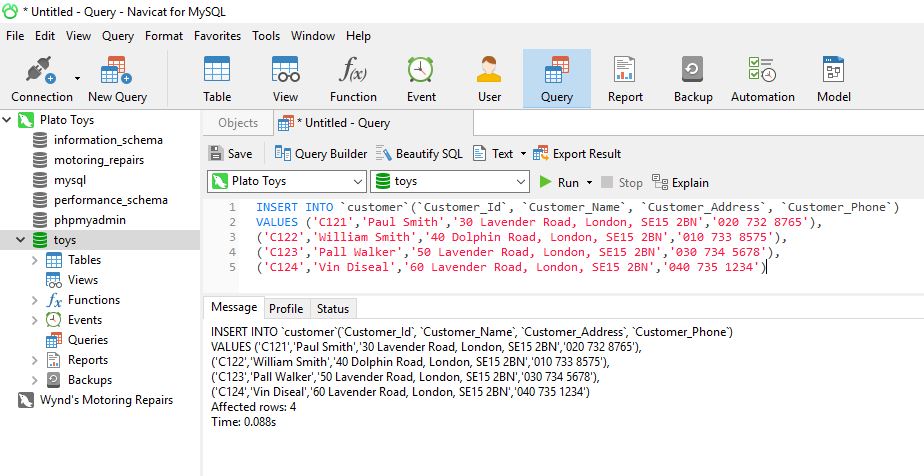
Figure-6: Create product table. 

Figure-7: Create product order table.

 Figure-8: Create complaint table.

1. Data insert and show in customer table.  Figure-9: Data insert.

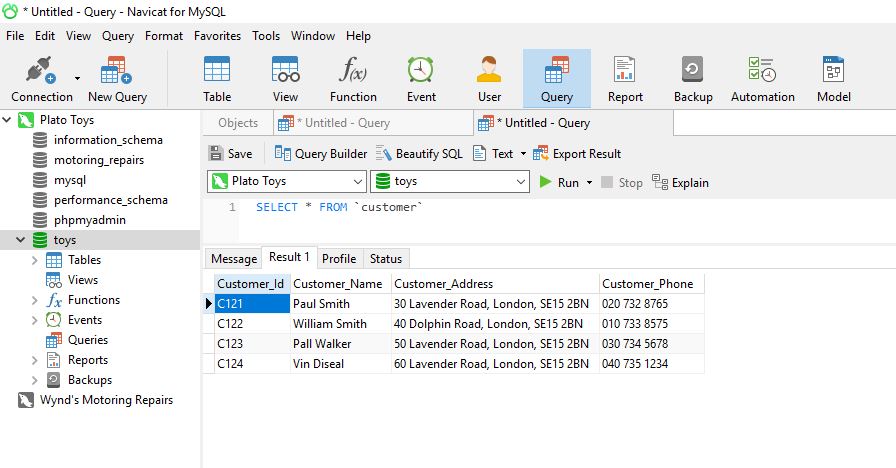


Figure-10: Data Show.

1. Data insert and show in all order and products.

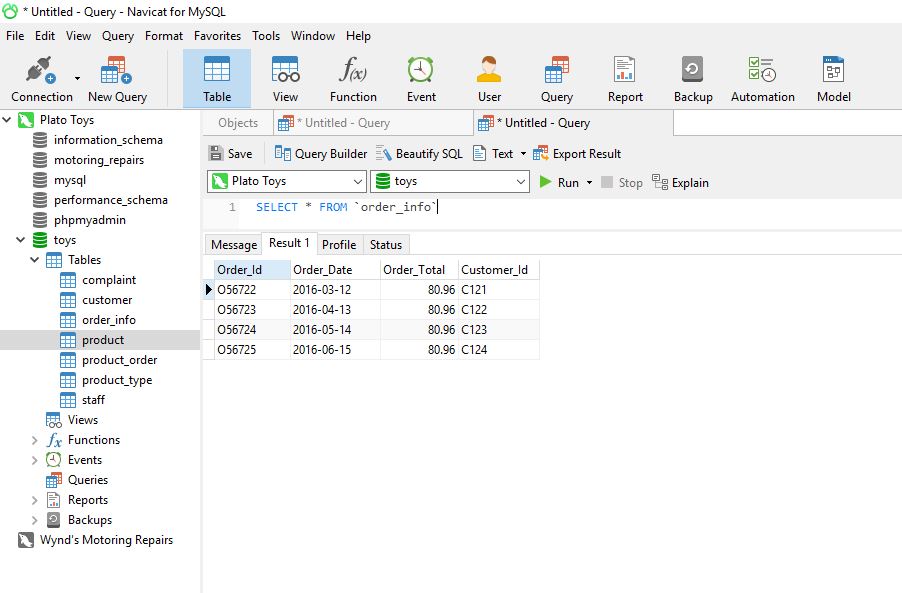
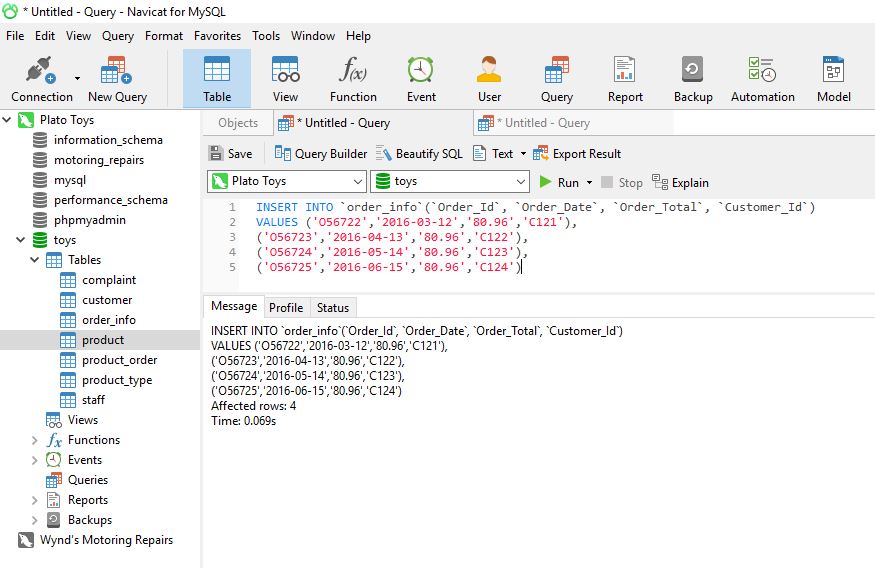
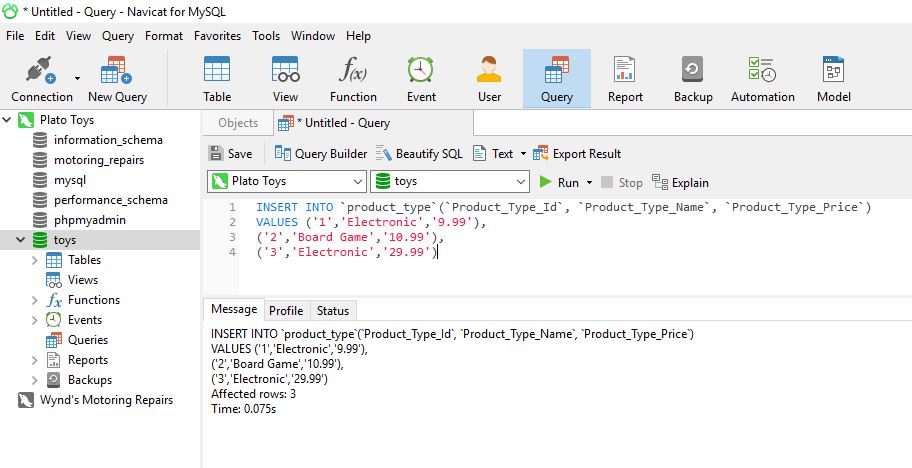
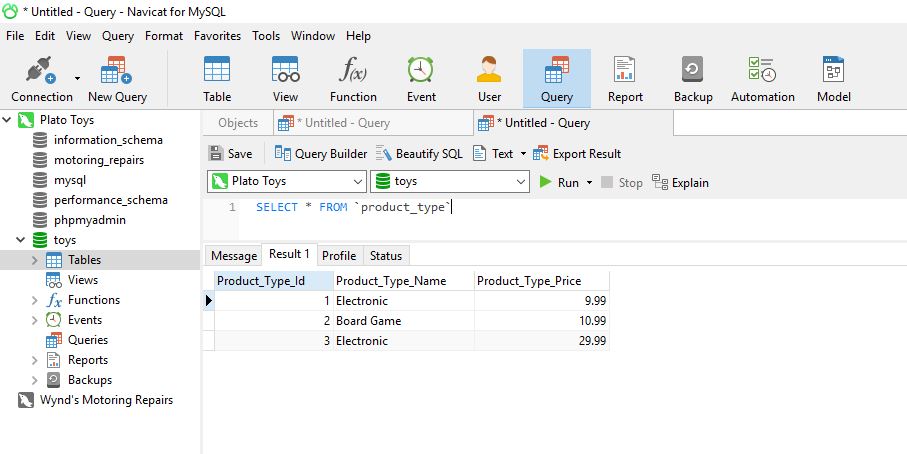
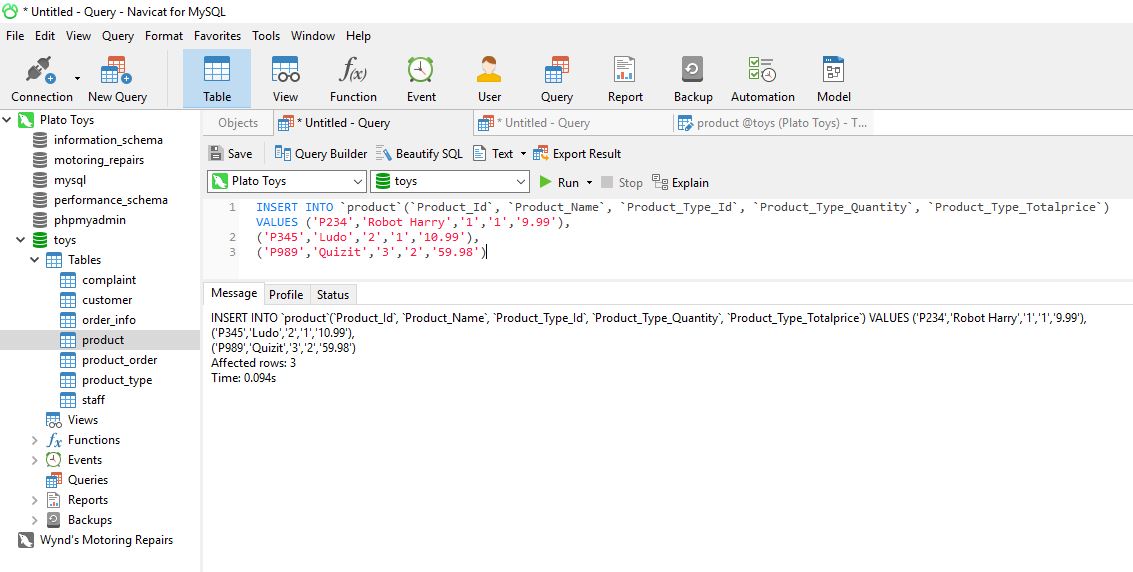
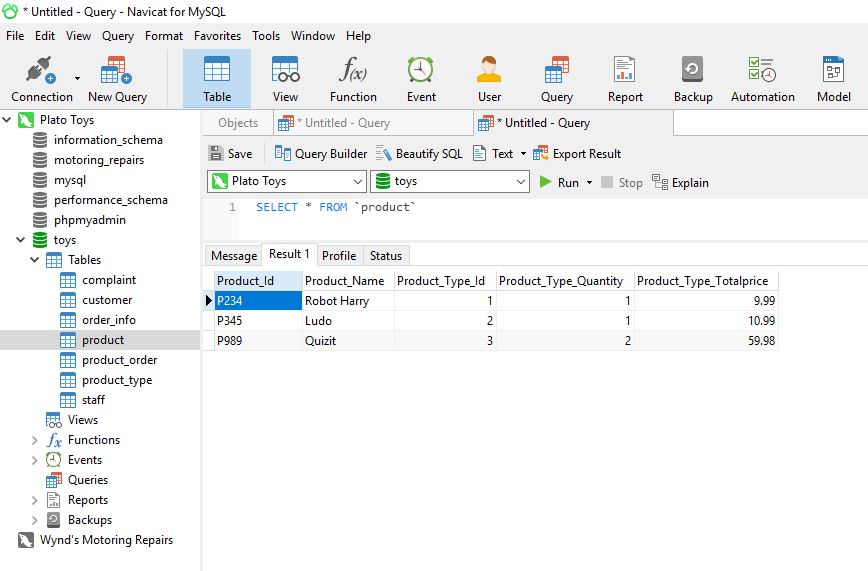
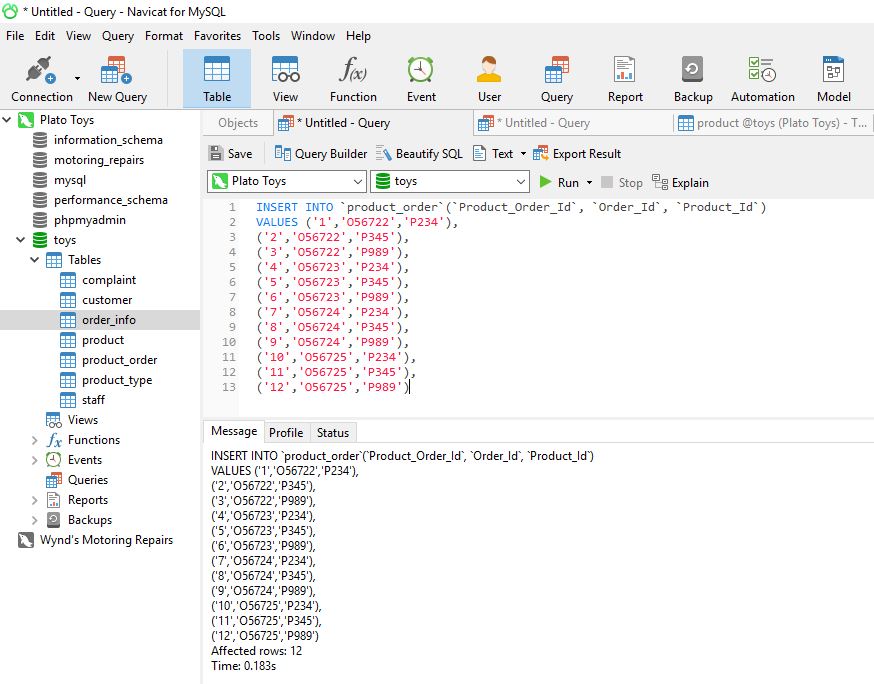


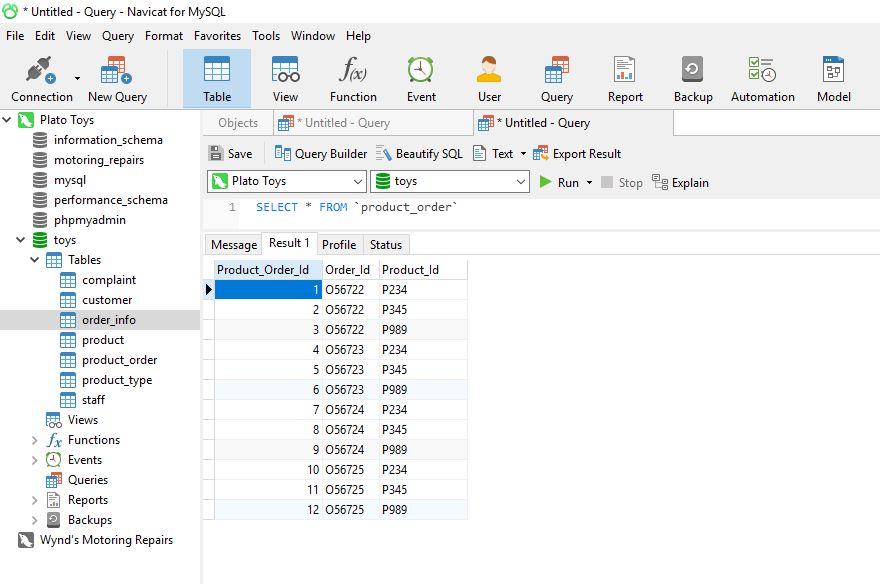
Figure-11: Insert Order info data Figure-12: Show Order info data

 Figure-13: Insert product type data.

 Figure-14: Show product type data.  Figure-15: Insert product data.

 Figure-16: Show product data.

 Figure-17: Insert product order data.

 Figure-18: Show product order data.

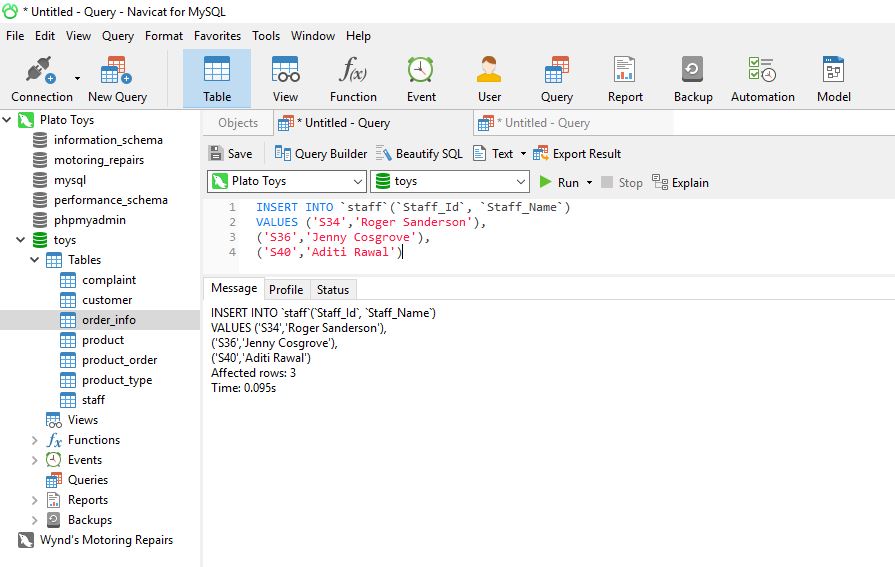
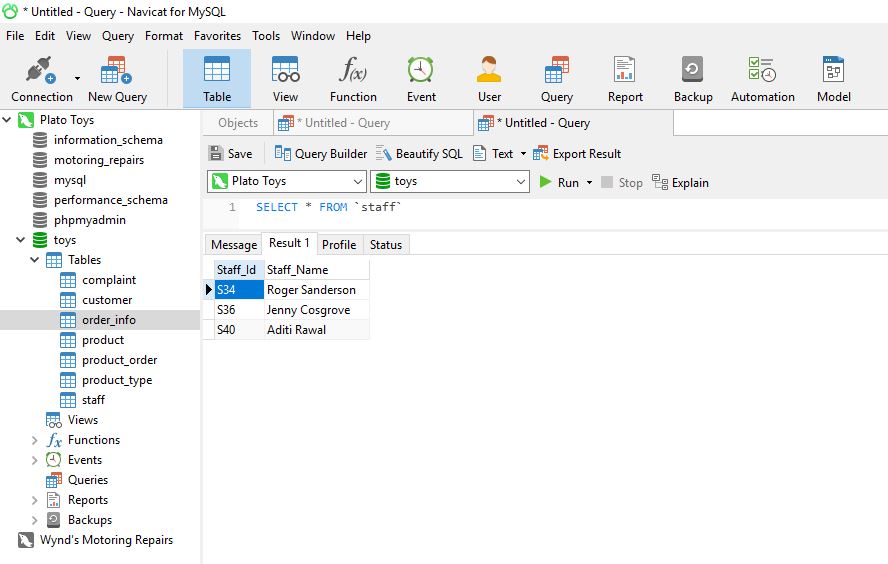
1. Data insert and show for all staff. 

Figure-19: Insert staff data.

 Figure-20: Show staff data.

1. Data insert and show for all complaint.

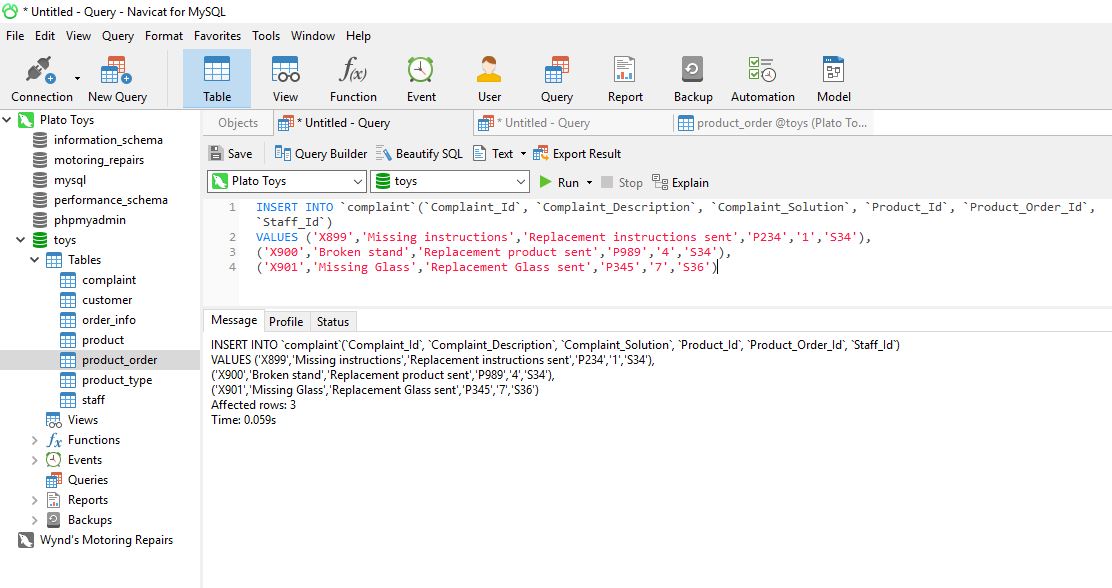
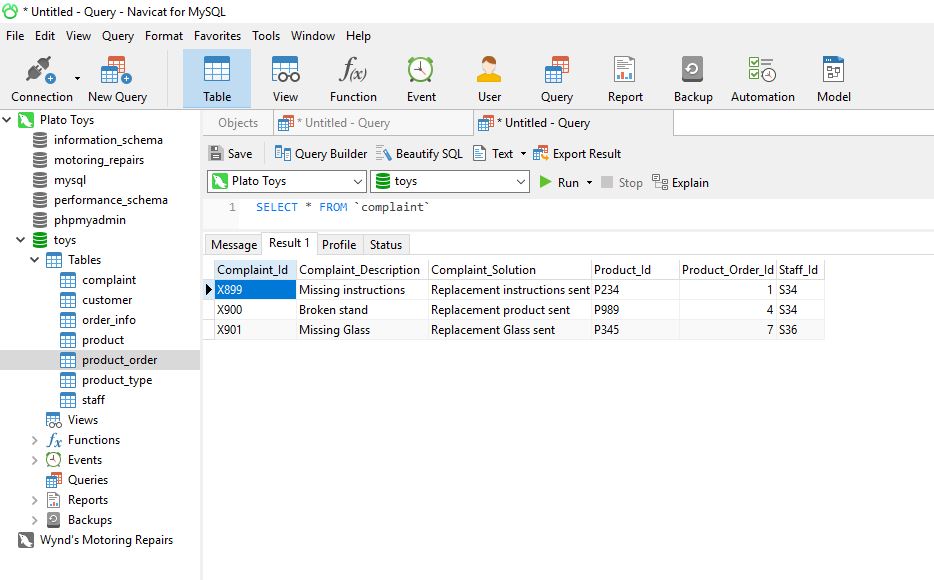
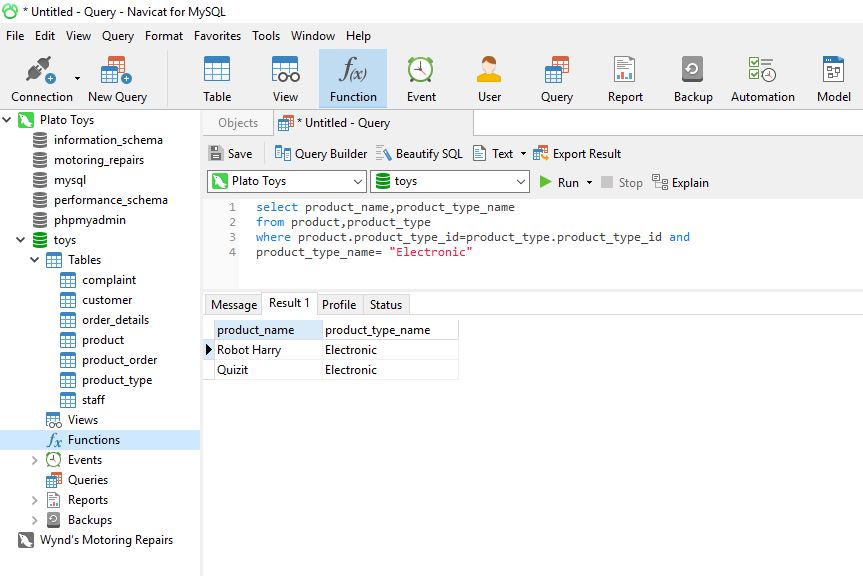
 Figure-21: Insert complaint data. 

Figure-22: Show complaint data.

1. Selects all ‘Electronic’ products type.

Figure-23: Show Electronic product type name.

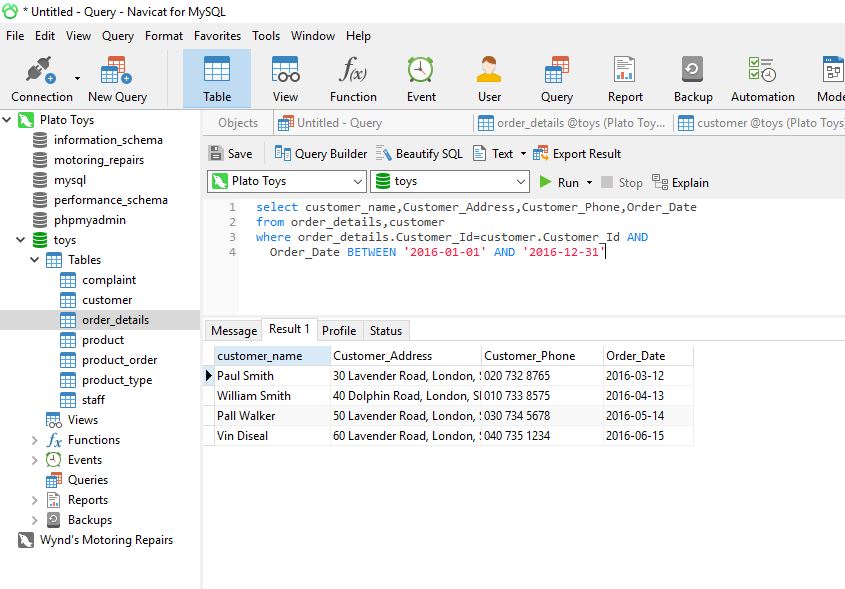
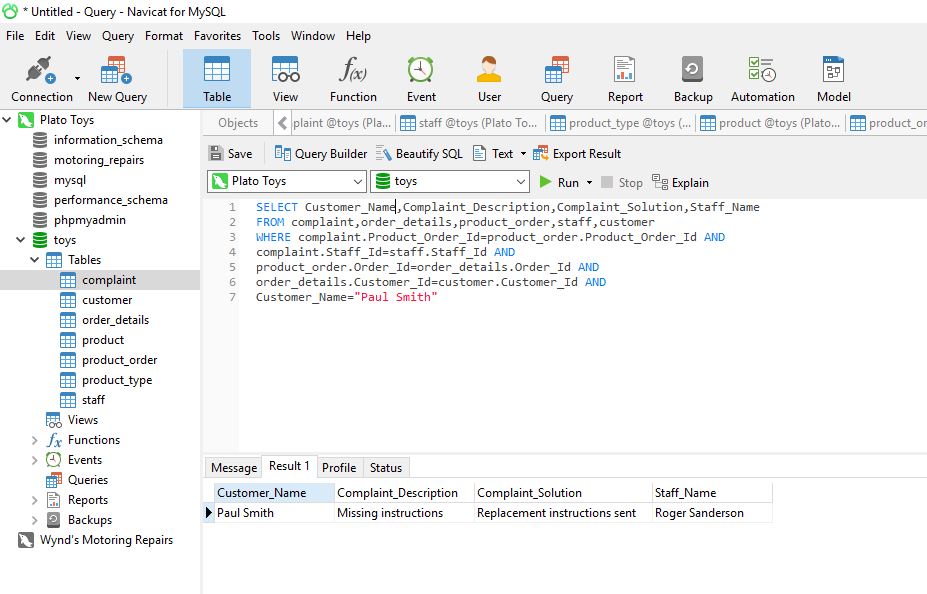
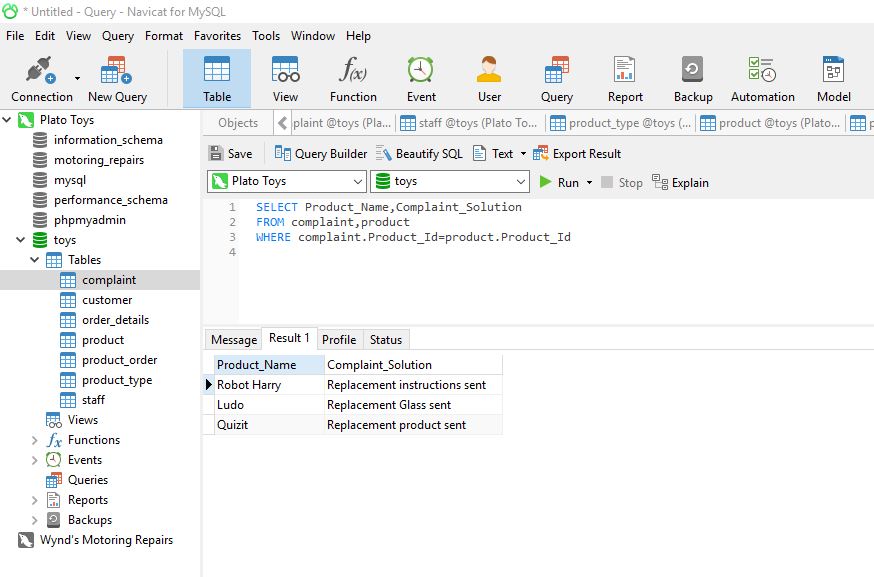
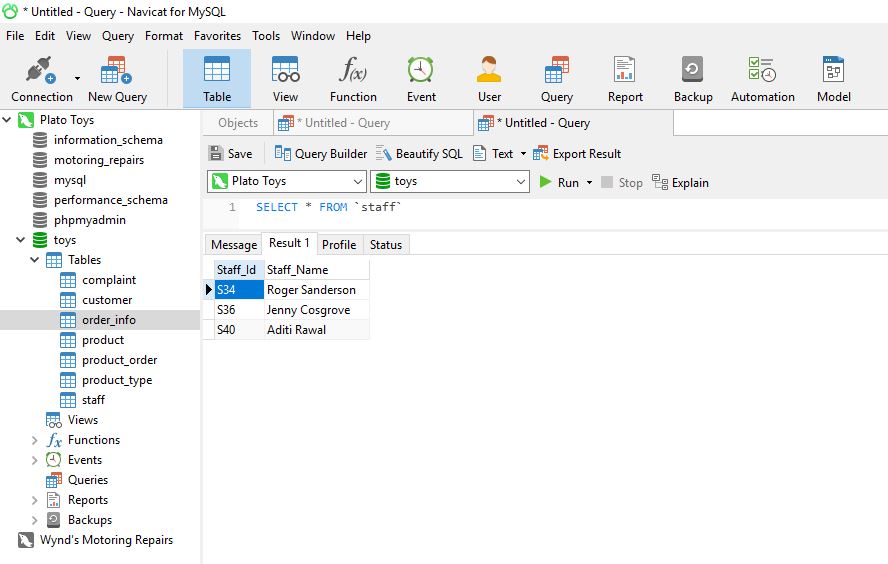
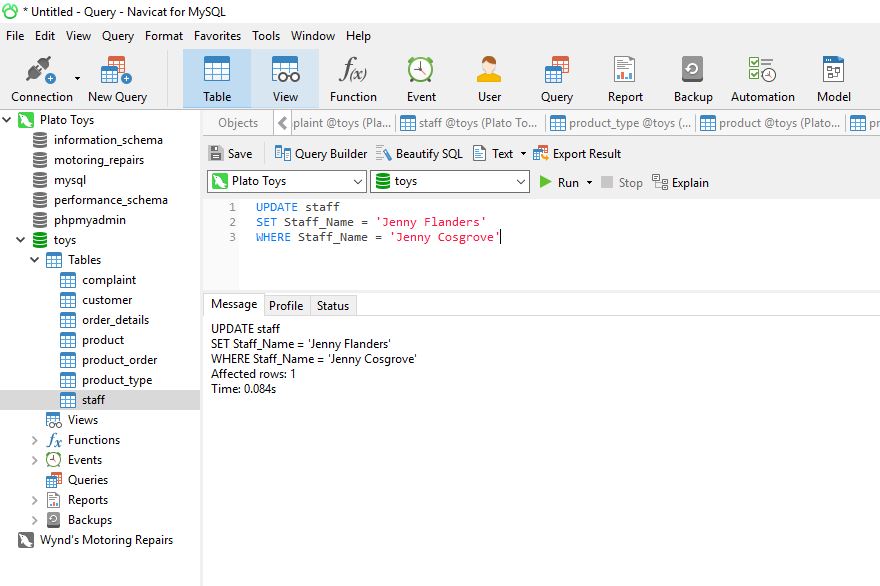
1. Selects all the customer details from customer who has made an order in the year 2016.  Figure-24: Show Customer details who has made an order in the year 2016.
2. Customer ‘Paul Smith’ complaints and the name of staff who is dealing.  Figure-24: Show customer ‘Paul Smith’ complaints and the name of staff
3. All products name where a complaint has resulted in a replacement product being sent. 

Figure-25: All product name with replacement details.

1. Update the database to change the name of the member of staff ‘Jenny Cosgrove’ to ‘Jenny Flanders’.

 Figure-26: Update before.

 Figure-27: When update.

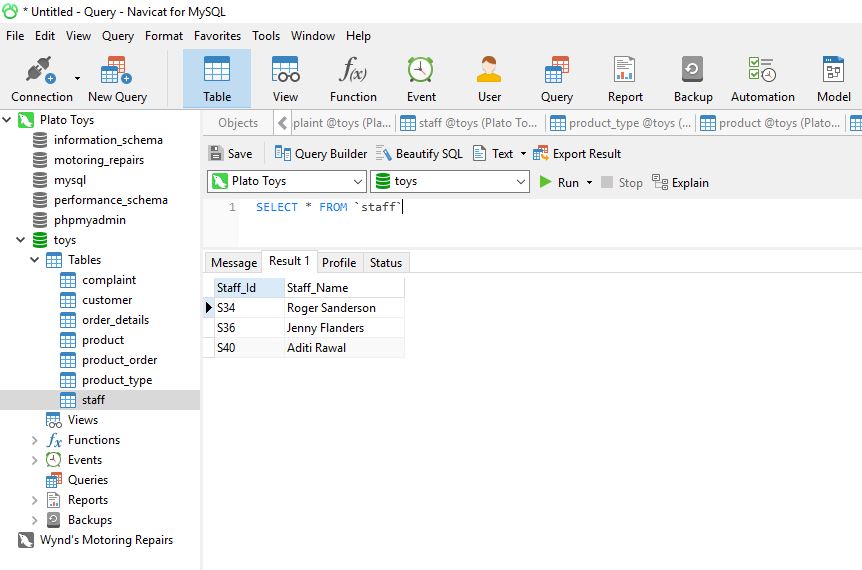


Figure-28: Update After.

**Task-3**

|  |  |  |  |
| --- | --- | --- | --- |
| **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 | Reducing and avoided Data duplicity must be. | Normalization processes Many to many relationship breakdowns to create one to many relationships. | Ensured that reduce data duplicity. |
| 3 | Standardization processes data organized properly. | Primary key used for uniquely identified data and foreign key used for share and access of data. | All the data is organized and identify in standard way. |
| 4 | New data and table 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 Customer\_Id, Staff\_Id,  Product\_Type\_Id  etc. |
| 5 | They want to yearly report with products complaint which is staff dealing. | To create their yearly report for used between functions create a yearly and complaint report sheet. | Yearly data and complaint data with staff name updated individual and totally. |

Conclusion:

This assessment has covered discussions on the assignment including was done to meet their requirements and assumptions that had to be taken while completing the tasks. So, this assignment helped me learn using SQL to create databases. It also helped me acquire knowledge about how a database can be designed and implemented with ERD model and data dictionary in a professional way.