**FOOD DELIVERY SERVICE**

**Professor:**

Dr.Mutlu Mete

**Project Members:**

Srivinusha Janga (50267673)

Lakshmi Chinnam (50263055)

Purushotam Pradeep Kumar Reddy Desireddy (50256934)

Whose Oracle Account is used for this project ? Srivinusha Janga

**Group ID:** **6**

Step 1: Create an imaginary scenario. Your scenario should satisfy following conditions:

This project (Food Service) involves below entities:

1) Delivery\_Service

2) Restaurant

3) Feedback

4) Employee

5) Customer

6) Menu

7) Delivery\_Vehicle

8) Order\_List

9) Booking

10) Offer

11) Payment

12)Emp\_Delivery

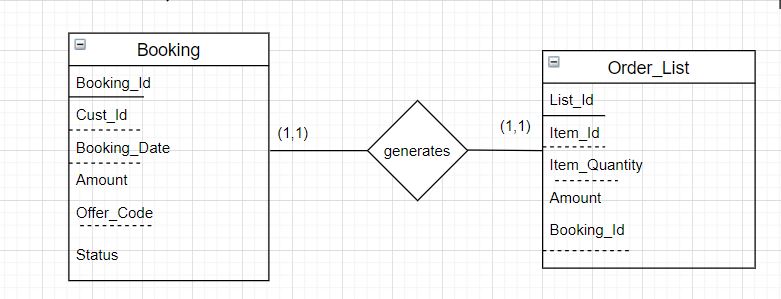
13)Nutrition\_Fact

This project involves below relationships between entities.

1. A Scenario including at least two one-to-one binary relationships.

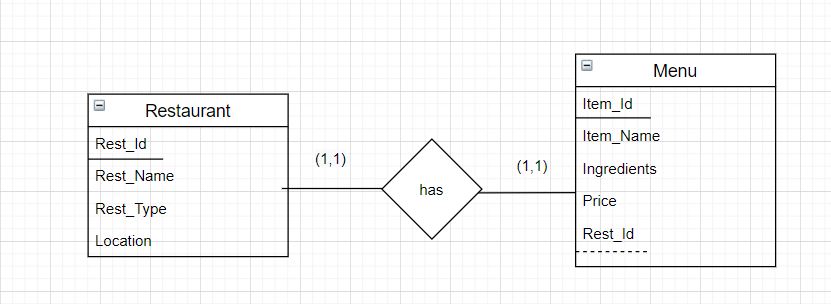
Order\_List - Booking:

This project involves each Booking generating only one Order.For one Booking\_ID only one Order\_List\_ID must be generated that can be written as at least one and at most one.



Restaurant-Menu:

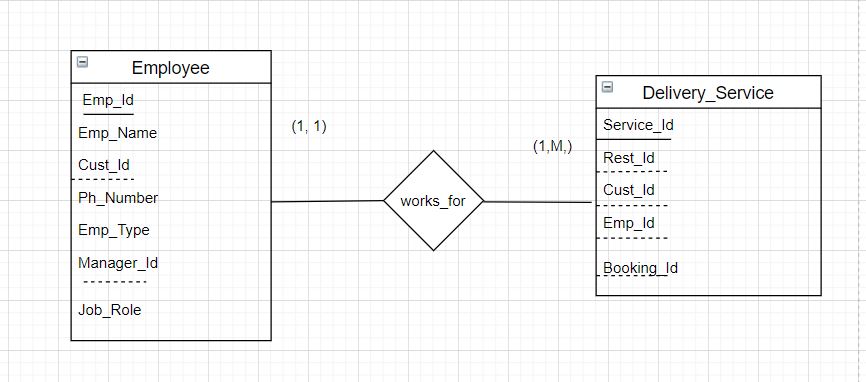
This project involves Restaurant having a one-to-one relationship with Menu. This Relationship shows Each Restaurant must contain one Menu.



2. A Scenario including at least two one-to-many binary relationships

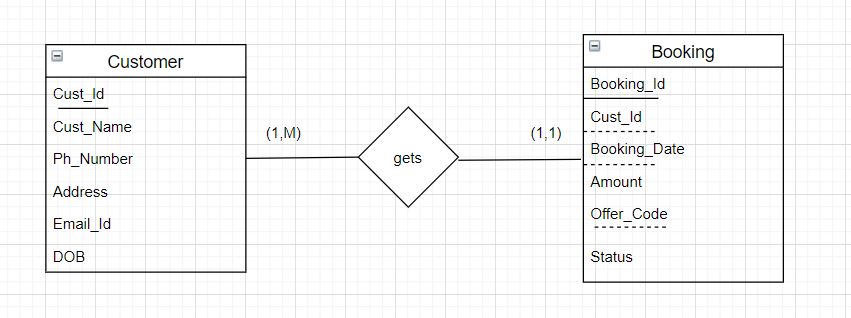
Employee-DeliveryService :

This project involves atleast one Employee at most many Employees working for Delivery\_service having atleast one atmost one Service.



Booking-Customer:

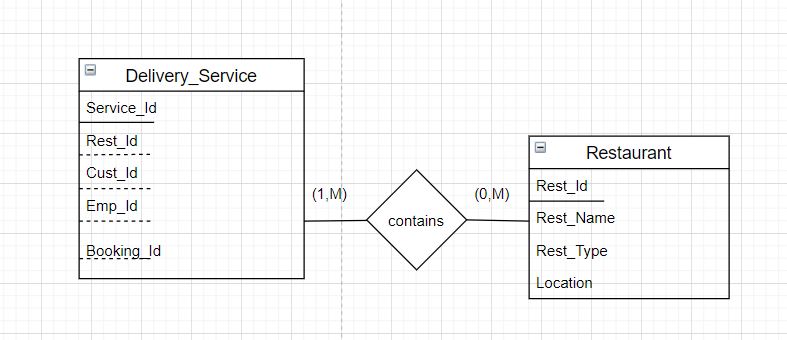
Customers get one or more Booking but Booking\_Id is unique for every Customer i.e., Invoice generates atleast one at most one for atleast one atmost many Customers.



3. A Scenario including at least two many-to-many binary relationships

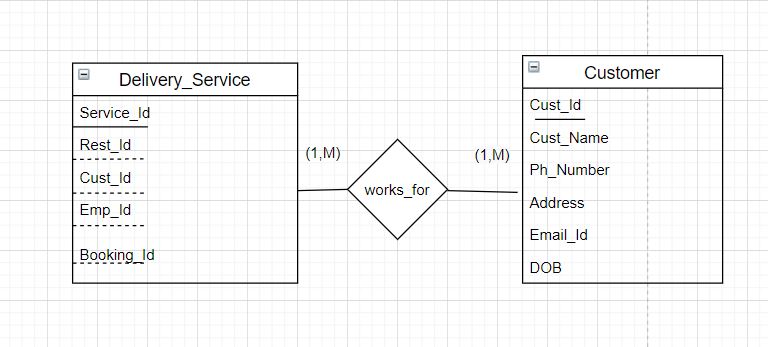
DeliveryService-Restaurant:

Delivery\_Service contains atleast one atmost many Restaurant and each Restaurant may not group with Delivery Service Systems or may join with Many Delivery Service Systems.



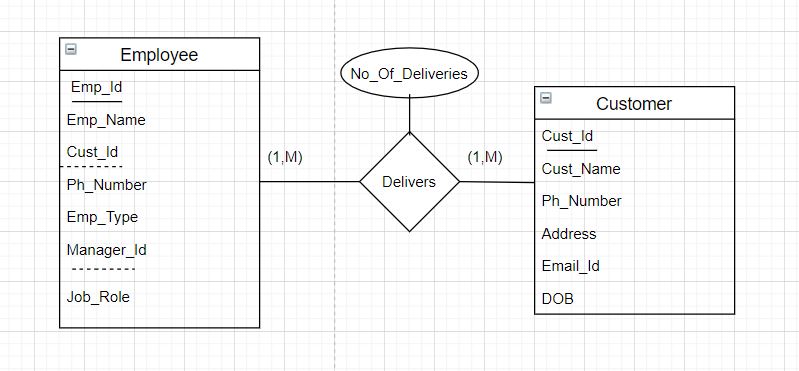
Delivery\_Service - Customer :

Customer can use many Delivery Services so,Customer can order from minimum one or maximum many delivery services in the same way Delivery\_Service is available for minimum one maximum many Customers.



4. A scenario including at least one intersection data over a many-to-many relationships

This project involves a many-to-many relation between Employee and Customer which further gives an intersection data . We have a Booking table which contains the BookingID from Booking table and Cust\_ID from Customer table to give us the Deliveries count received by the customer and Deliveries with the help of BookingID and Deliveries\_Number multivalued composite attribute of EMPLOYEE called Emp\_Delivery with CUSTOMER .



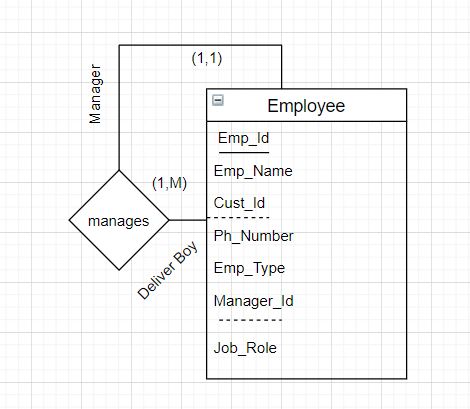
5. A scenario including at least one one-to-many unary relationships

Employee:

The Manages relationship type relates an employee to a Manager, where both employee and Manager entities are members of the same EMPLOYEE entity set. Hence, the EMPLOYEE entity type

participates twice in Manages: once in the role of Manager and

once in the role of food delivery person. Each relationship instance in Manages associates two different employee entities e1 and e2 one of which plays the role of Manager and the other the role of food delivery person.

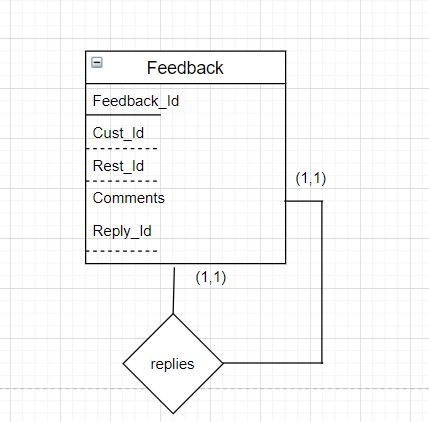


6. A scenario including at least one one-to-one unary relationships

Feedback :

This project involves for Every Feedback comment there must be a reply from Restaurant Organization do reply for every comment in Feedback(minimum one reply maximum one reply).

Here, for one comment will allow only one reply to that comment which is recursive, Each reply can also be a comment.



Step 2: Explain the story behind the scenario, and all your assumptions, which are required to support relationships given above.

Food Delivery Service :

* Nowadays people are faced with a situation within which they are doing not have time to cook or prepare food.Access to a good sort of foods one among the foremost important advantages that these services provide is that the people may order from an oversized menu.
* These days a spread of local eateries and national restaurants participate in online food ordering by partnering with delivery companies like Uber Eats, Postmates, Grubhub and Waitr.
* These services allow customers to pursue a bigger selection of foods and restaurants and order food through a convenient online page or app.having it delivered to their exterior door,is helping these people get their work done typically seniors.
* Each Delivery\_Service have sheer number of Restaurants offered.
* Adding delivery service to the restaurant will help you distinguish yourself from the competition,and it also attract Customers.
* It should have a good Food Delivery Staff(Employee) ,Delivery drivers have a their own vehicle(Emp\_vehicle),Each Delivery person will be managed by the manager.
* Each employee will be trained on the best way to place the food in their cars so no spills occur and how to speak to the customer when they deliver the food.

**Story behind our mini world:**

This project involves

1) Food Delivery Service includes many offers

For e.g., the list of offers the project contains are;

* Senior Citizen Discount
* Volume Discount
* Trial Discount
* Seasonal
* Happy Hours
* Free delivery

Each OfferCode could also be available for under Booking\_ID

3)This project provides below payment methods:

* Credit Card
* Debit Card
* PayPal
* Cash

4)Employee have two roles i.e Manager and Food Delivery Person.Manager who manages Many employees.

5)A Customer will have a chance to share feedback to the Restaurant and the Restaurant must give a reply to the comment.

Assumptions:

1) A Customer will have a chance to share feedback to the Restaurant and Restaurant must give a reply to the comment.

Step 3: Show your entities and their attributes. Each entity should have at least three attributes.

1) Delivery\_Service

* \*Service\_ID
* Rest\_ID
* Cust\_ID
* Emp\_ID
* Booking\_ID

2) Restaurant

* \*Rest\_ID
* Rest\_Name
* Rest\_Type
* Location

3) Feedback

* \*Feedback\_ID
* Cust\_ID
* Rest\_ID
* Comments
* Reply\_ID

4) Employee

* \*Emp\_ID
* Emp\_Name
* Cust\_ID
* Ph\_Number
* Emp\_Type
* Manager\_ID
* Job\_Role

5) Customer

* \*Cust\_Id
* Cust\_Name
* Ph\_Number
* Address
* Email\_ID
* DOB

6) Menu

* \*Item\_ID
* Item\_Name
* Ingredients
* Price
* Rest\_ID

7) Delivery\_Vehicle

* \*Vehicle\_ID
* Emp\_ID
* Vehicle\_Type

8) Order\_List

* \*List\_ID
* Item\_ID
* Item\_Quantity
* Amount
* Booking\_ID

9) Booking

* \*Booking\_ID
* Cust\_ID
* Booking\_Date
* Amount
* Offer\_Code
* Status

10) Offer

* \*Offer\_Code
* Offer\_Type
* Disc\_Amount
* Validity

11) Payment

* \*Pay\_ID
* Pay\_Type
* Amount
* Booking\_ID

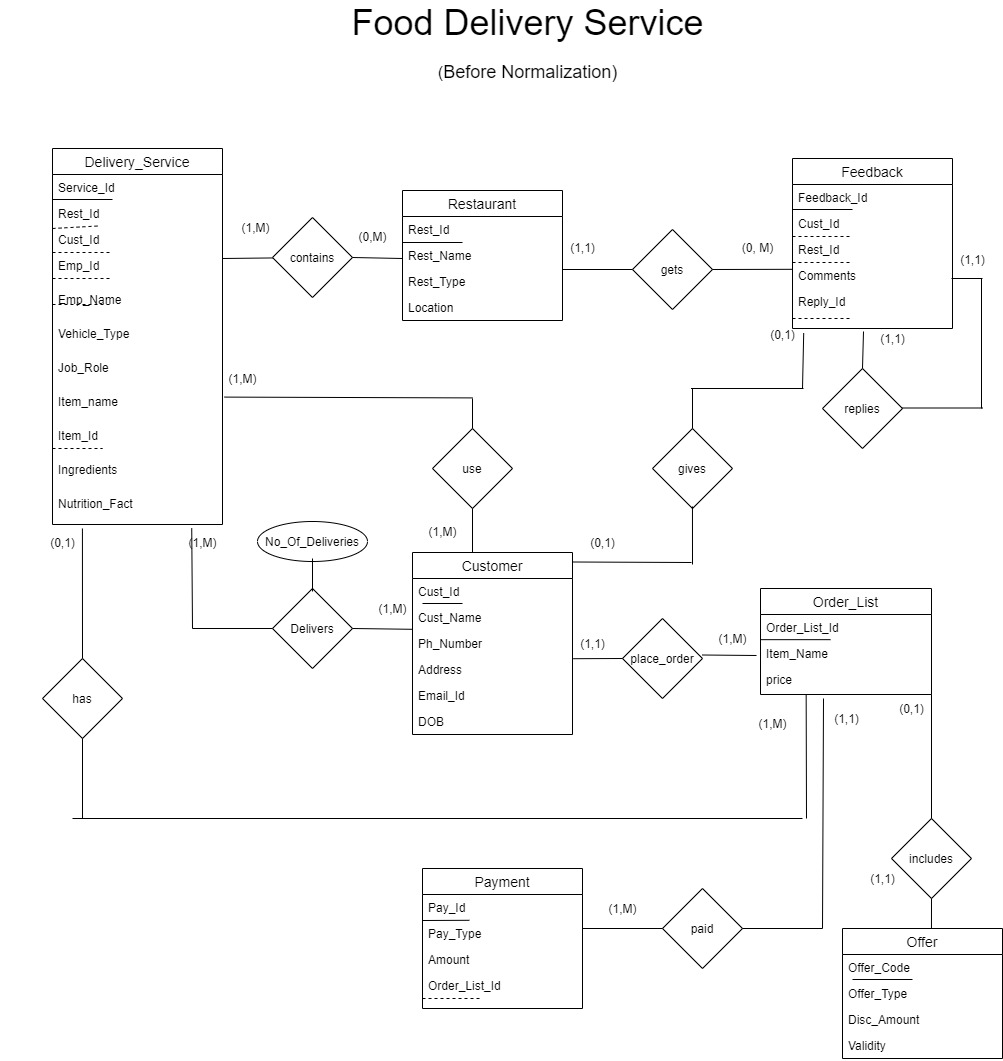
12)Emp\_Delivery

* Emp\_ID
* Service\_ID
* Deliveries

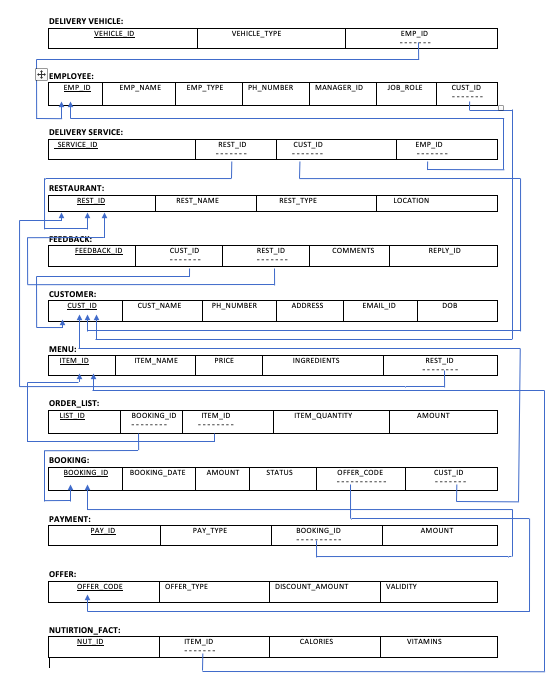
13)Nutrition\_Fact

* \*Nutrition\_ID
* Item\_ID
* Calories
* Vitamin

Step 4: Draw your initial ER diagram with min-max notation

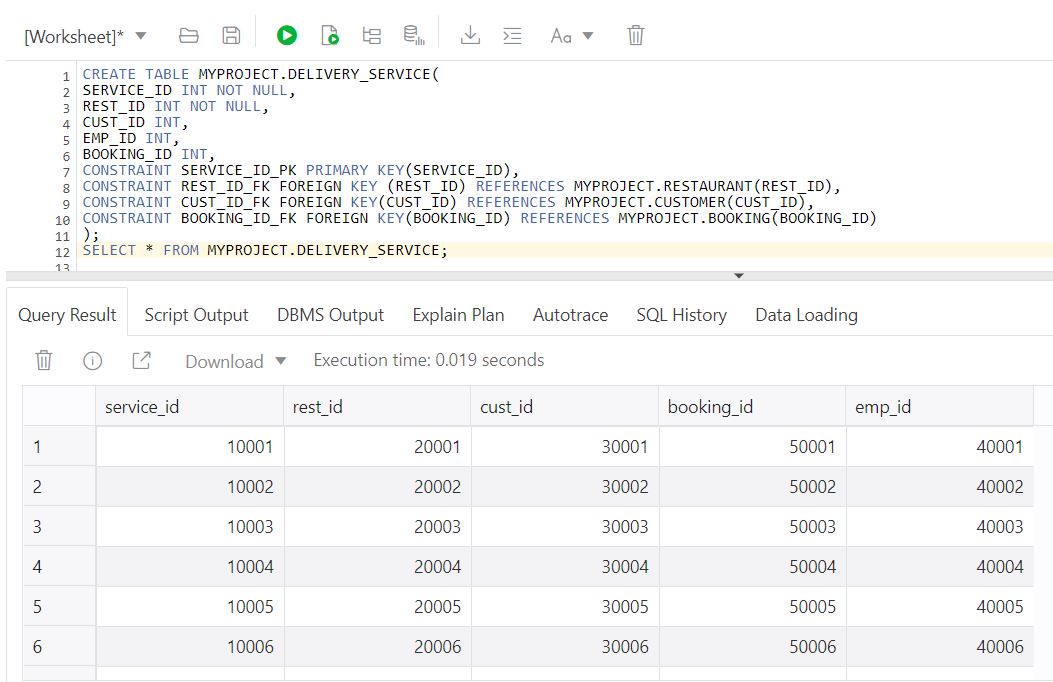


Step 5: Show your referential integrity rules for your scenario(FK->PK)



6) Convert the ER diagram to tables. Show your tables with primary keys

Table 1:Delivery\_Vehicle

Table 2: Restaurant

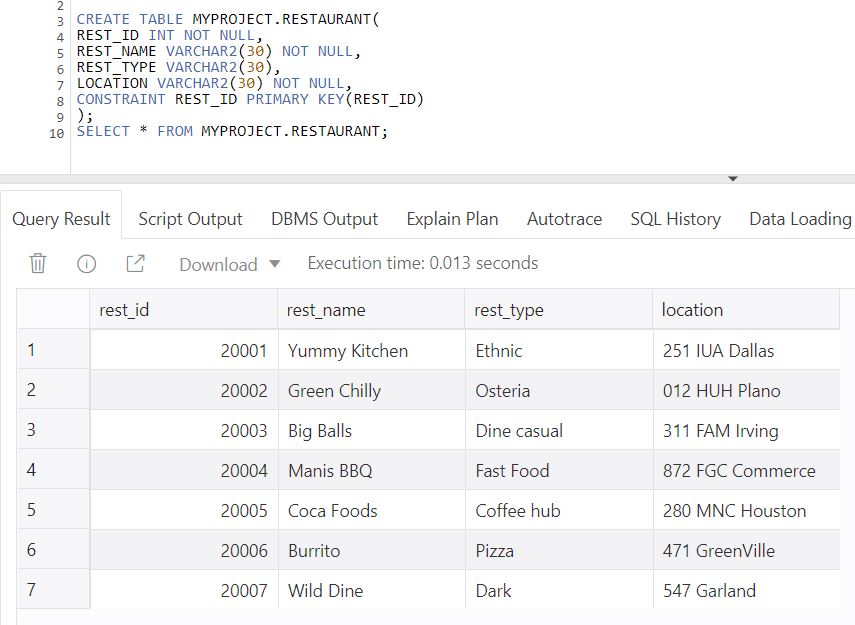


Table 3:Feedback

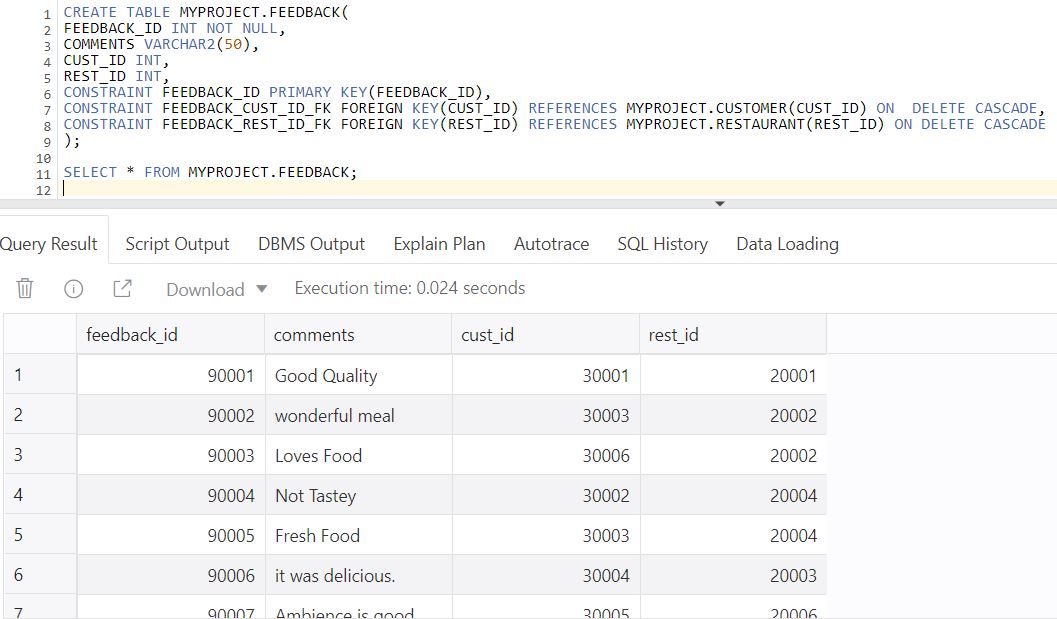


Table 4:Employee

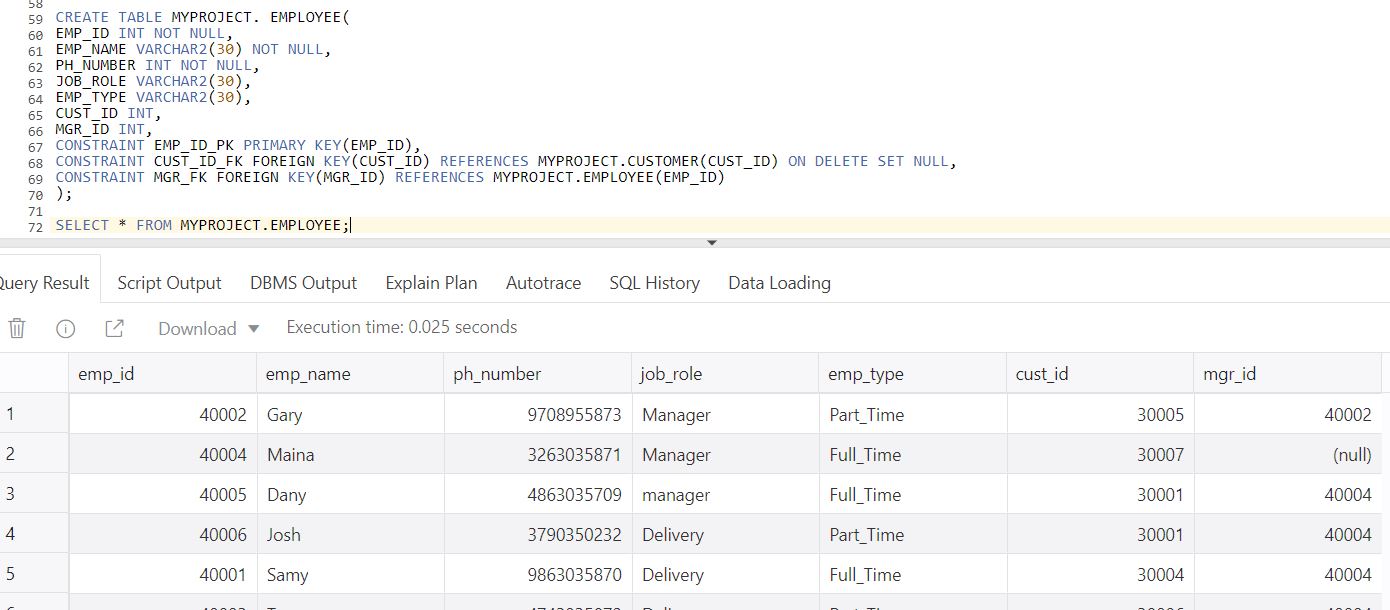


Table 5:Customer

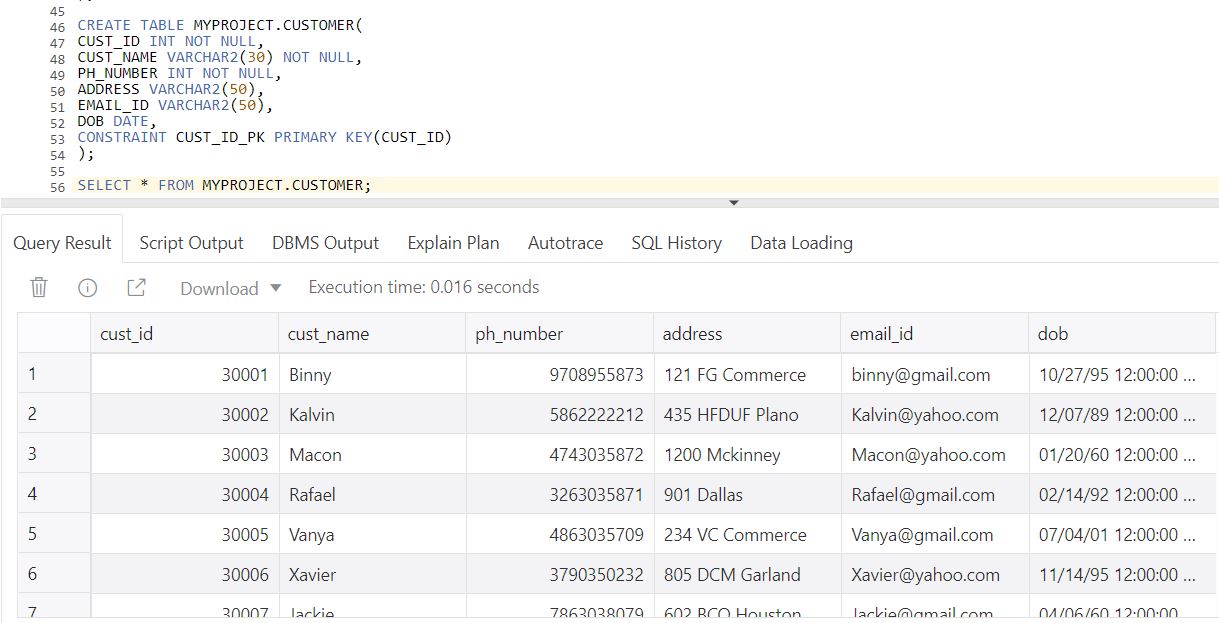


Table 6:Menu

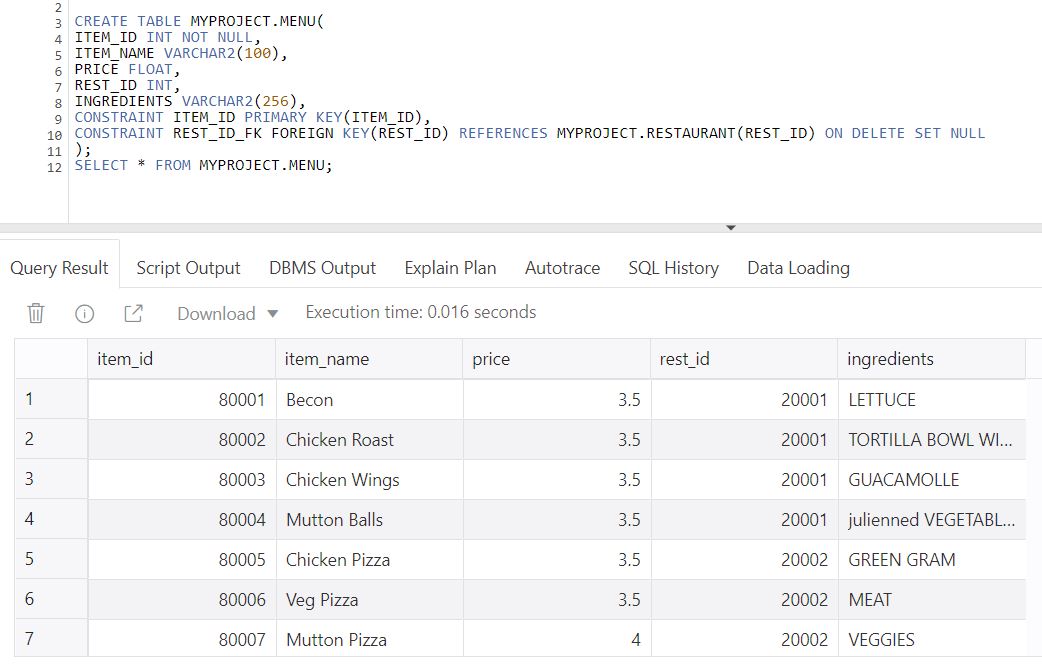


Table 7:Delivery\_Vehicle

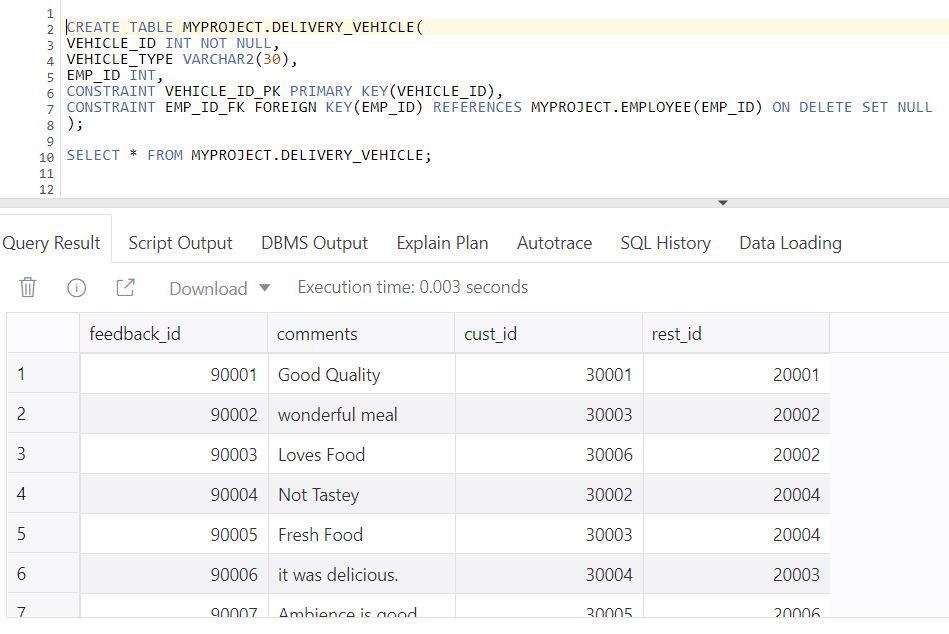


Table 8:Order\_List

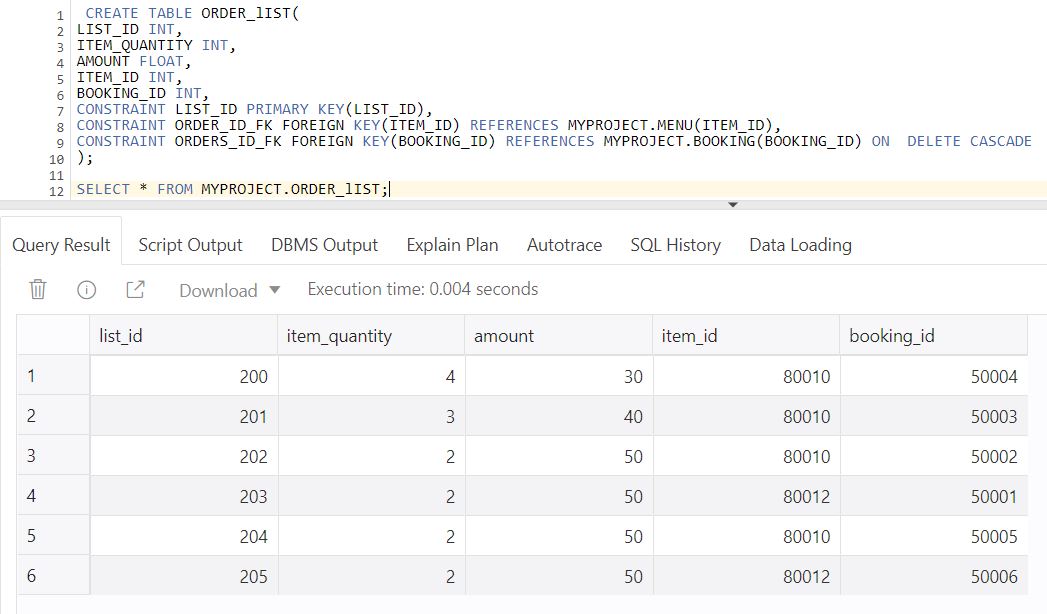


Table 9:Booking

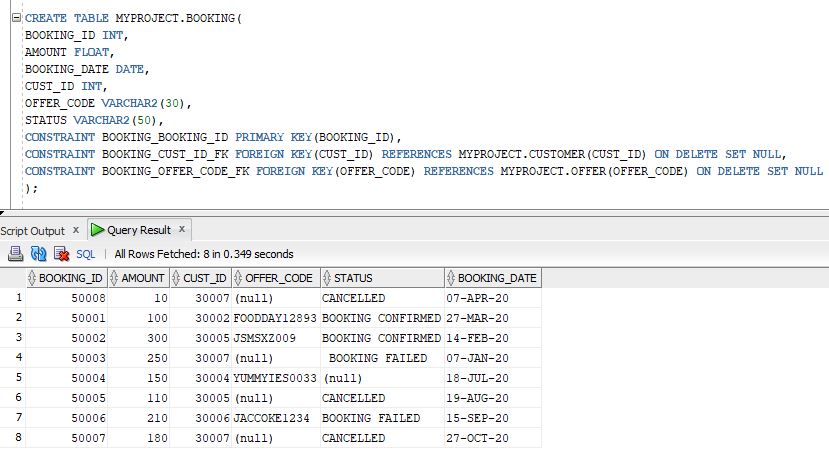


Table10 :Offer

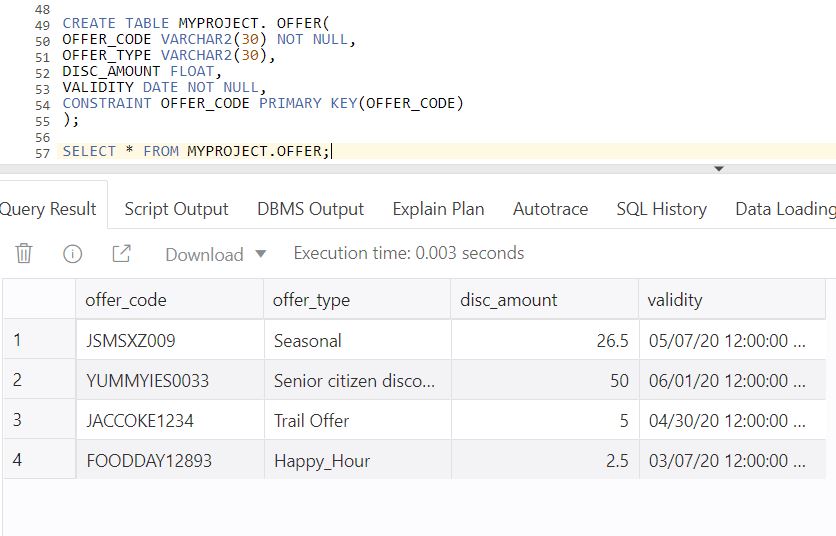


Table 11:Payment

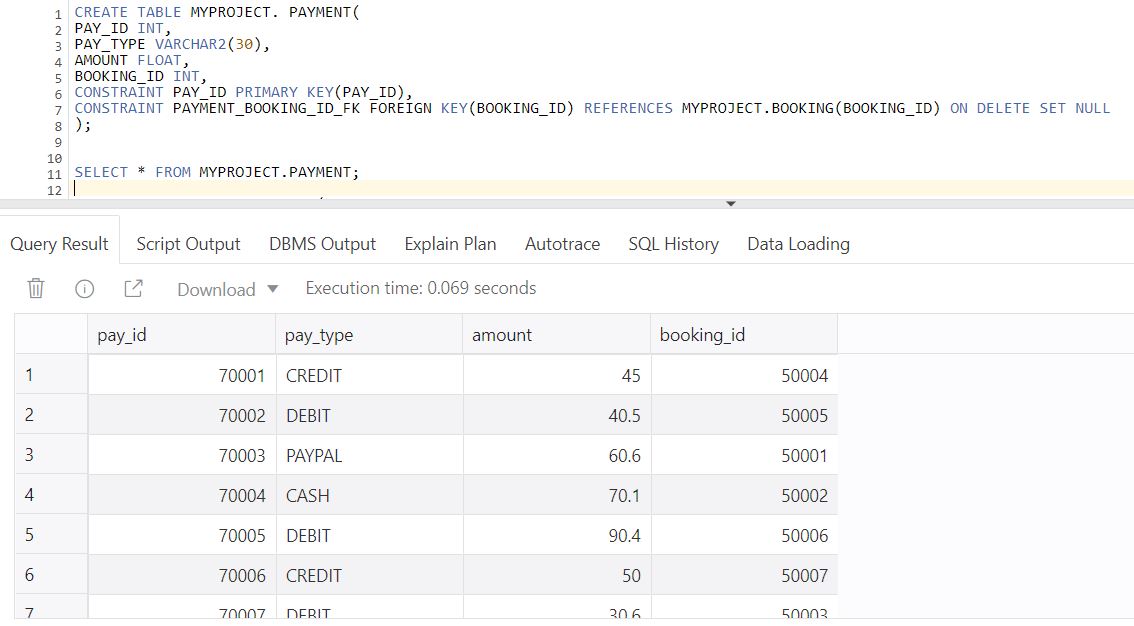


Table 12:Emp\_Delivery

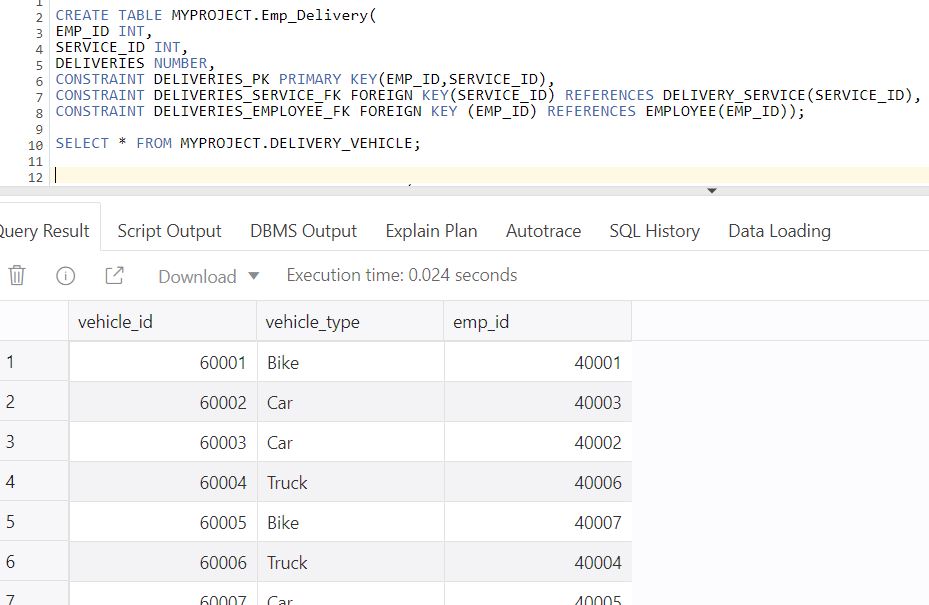
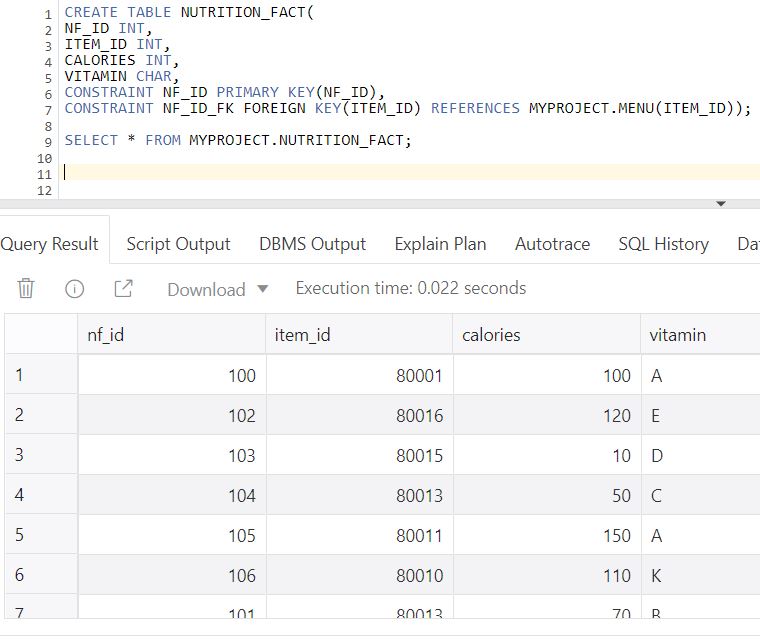


Table 13:Nutrition\_Fact



7) Discuss database normalization rules on your tables. Do not explain what the rules are. Just check if each of the tables satisfies 1NF, 2NF, 3NF and BCNF. If not, normalize your tables.

Database Normalization rules on Tables:

This project violates Normalization rules in the table Delivery\_service.Booking ,Vehicle information is repeated for each Employee.This violates 1st Normal Form .Primary key must uniquely identify attribute value .So, Booking and Vehicle Tables were formed.If we delete some critical information we may lose.

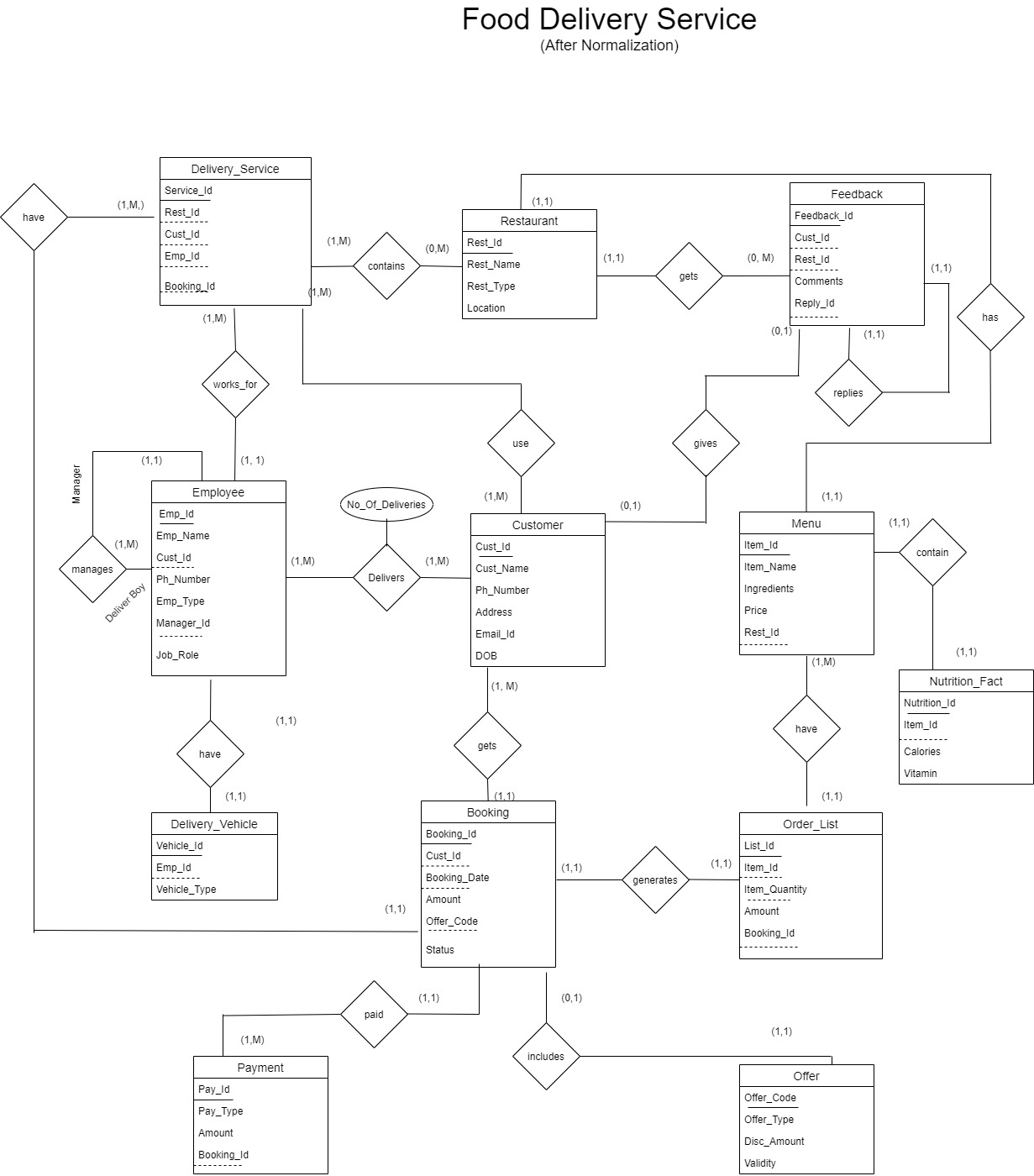
Emp\_name is not fully dependent on Vehicle\_Id,Booking\_Id 2nd Normal Form violated. So ,an Identified new table Employee is formed.

Ingredients,Nutrition\_Fact dependent on Item\_Name non-prime attribute dependent on non-prime attribute violates third normal form,in order to remove the Transitive dependency Menu table was formed.Updating Item\_Name forms inconsistent in data. New table with removed dependency was created.

Category Table we see that not all attributes are fully dependent on the primary key.The only attribute that is fully dependent is Item\_Name.

Amount is Fully dependent on Item\_Quality ,Ingredients,Nutition\_Fact is fully dependent on Item\_Name and Item\_Quality.These attributes were non-prime attributes So Menu Table is formed.Menu Table each non-key attribute i.e., Nutrition\_Fact is not fully dependent on Primary Key(Item\_Id). Here,the Menu table violates BCNF as every attribute should be a candidate key, so a new table (Nutition\_Fact) is identified.

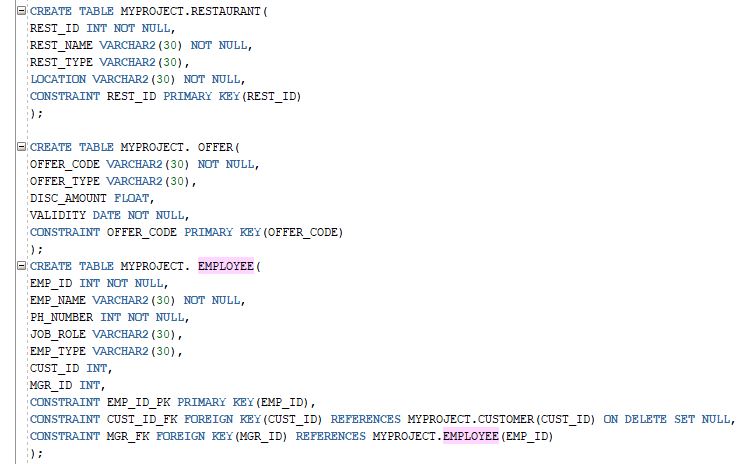
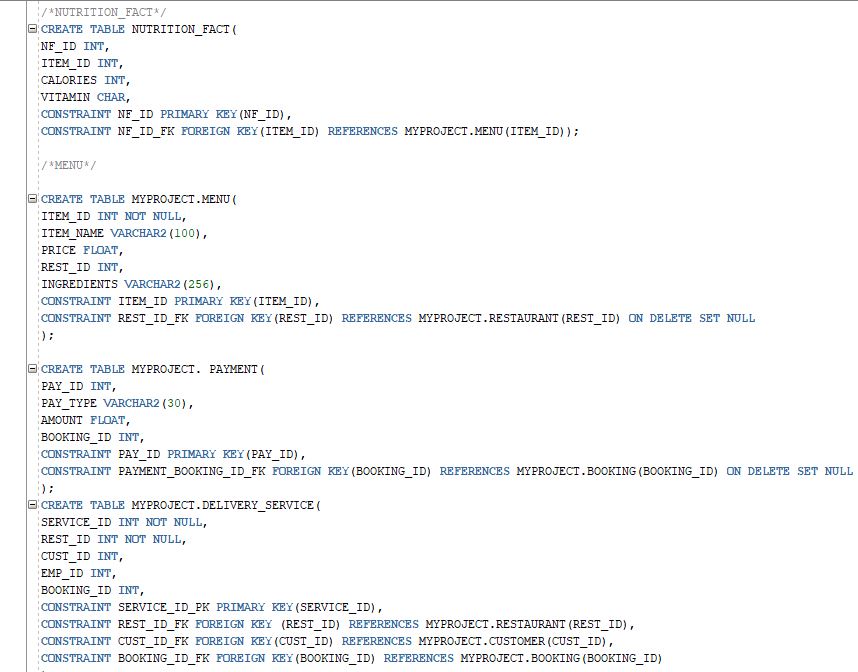
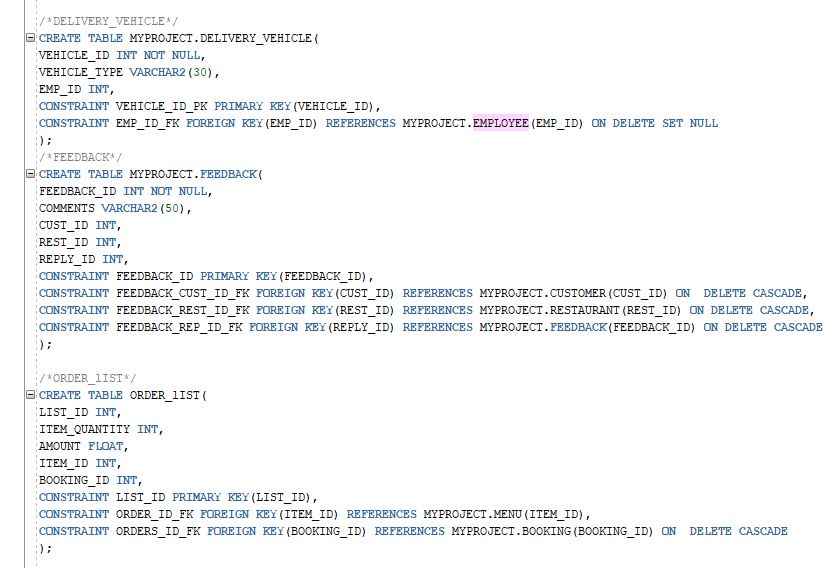
8) Show the latest table/ER design after normalization.

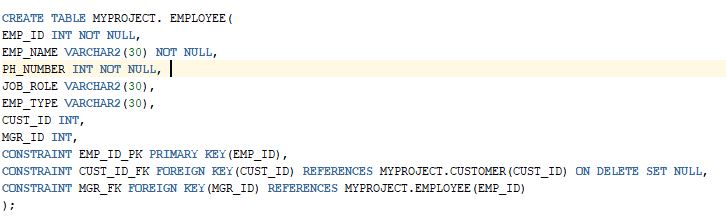


9) State an imaginary cascade deletion rule on any table(s). (Not a SQL query required, only give an imaginary example, and describe the reason behind it.)

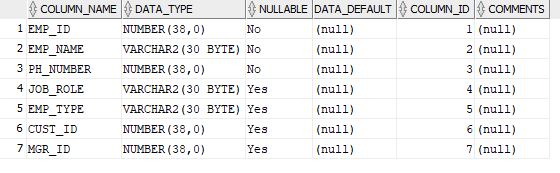
If Restaurant is removed due to some loss the corresponding feedback comments related to Restaurant will be deleted.i..e In This project If the Restaurant Primary Key is deleted then Corresponding Feedback Id will be deleted because Rest\_Id is foreigh key in Feedback table.

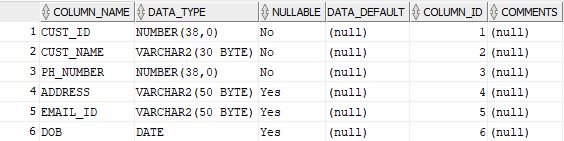
10) Create tables using SQL commands. Think about the nature of your scenario to decide about NULL attributes.



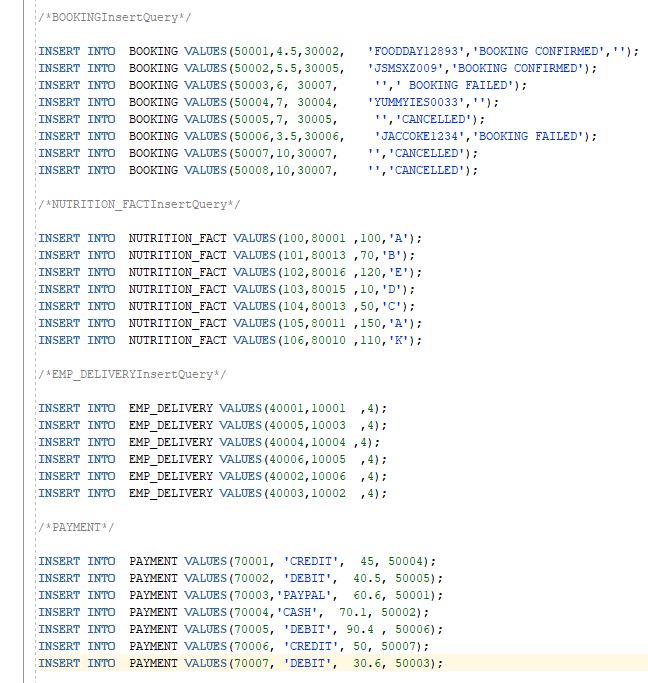
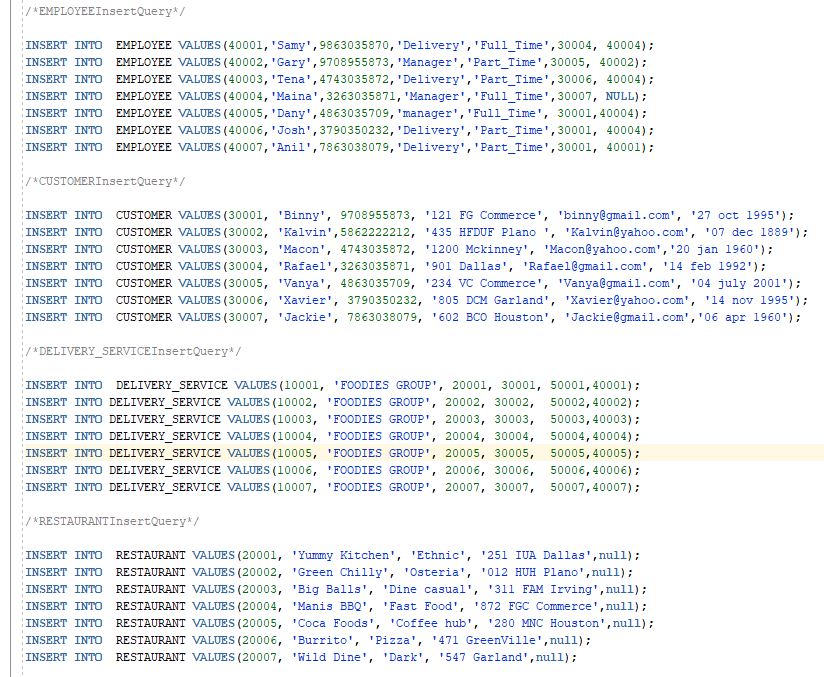


Here few Tables that accepts null:

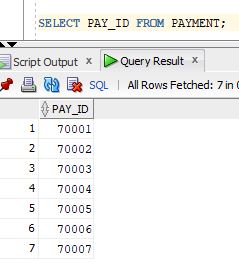
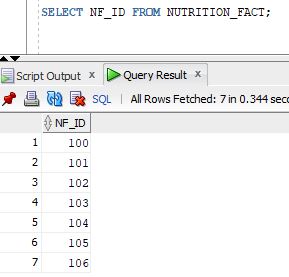
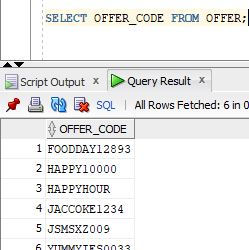


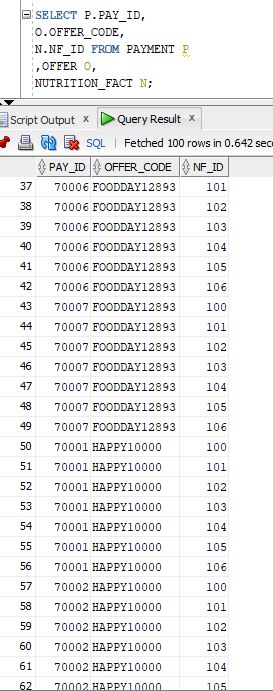


11) Insert at least five (5) records into each table.



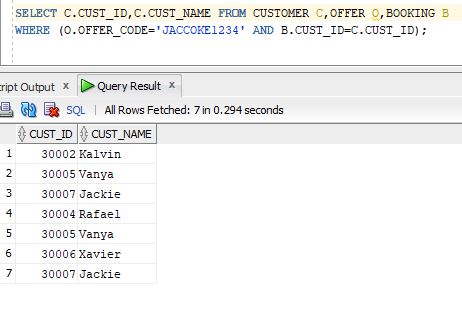
12) List only primary keys values for three different tables. (Select command on three different tables. No join operation is required.)

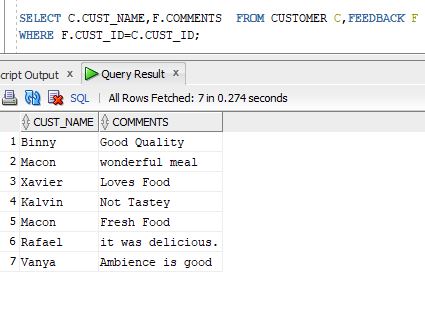




13) Demonstrate two (2) SELECT commands with WHERE statement.

Listing all the Customers ID’s and Names who uses offer code ‘JACCOKE1234’

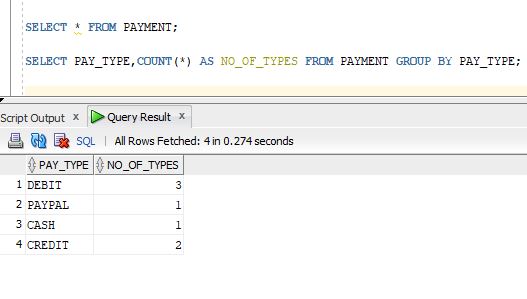




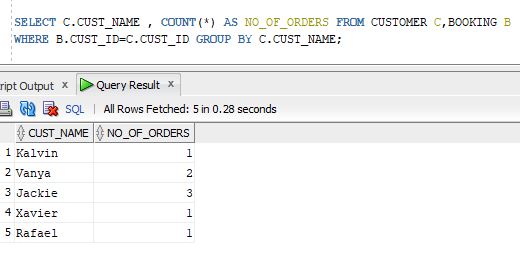
Listing all the Customer name along with their Comments to the

14) Demonstrate two (2) SELECT commands with GROUP BY statement.

* Payment Type count used by Customers.
* Show number of Payment\_Types used by Customers.

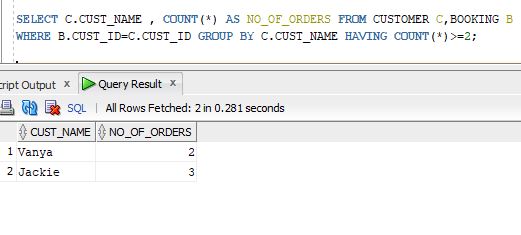


* Number of orders placed by each customer.

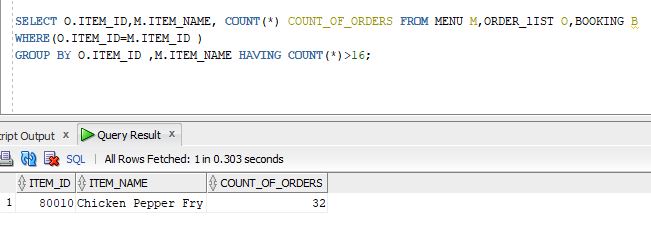


15) Demonstrate two (2) SELECT commands with HAVING statement.

* To know which customer has more than or equal to two orders.
* Showing Item\_Id along with Item\_Name with more than 16 orders to know the item which have placed more than 16 times.

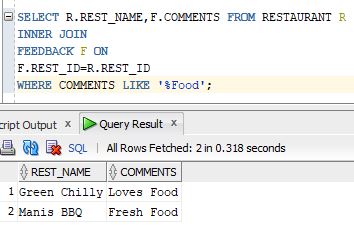


* Item which have the highest number of orders.
* Listing Customers who have placed orders more than or equal to 2



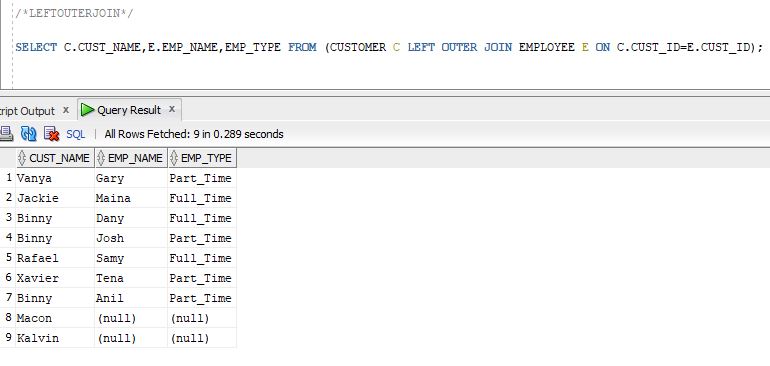
16) Using two related tables (meaning logically connected with primary-key and foreign-key pairs), run an inner join statement to show matching rows. For instance, assume that Table A and Table B have 4 and 5 attributes respectively. Also, assume that Table A’s primary key is seen as foreign key in Table B. Use join operations to show matching rows whose primary key and foreign key is the same.

Showing all Restaurant Names who has comments which includes Food in it.



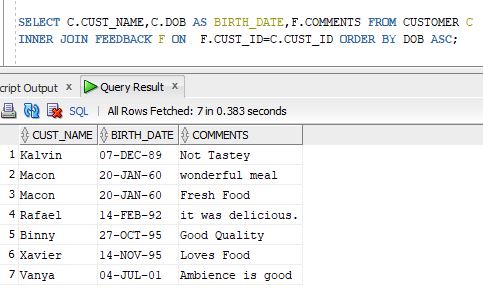
17) Demonstrate a left join statement.

List all the Customers who received orders from Employees.

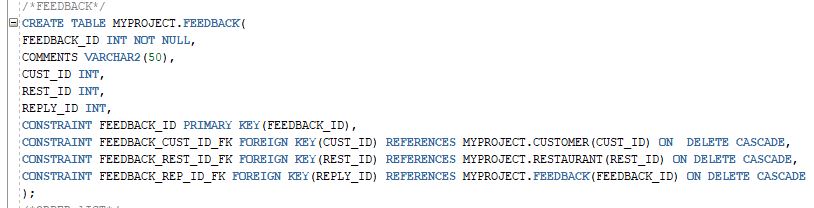


18) Demonstrate ORDER BY statement to order inner join operation according to foreign key. (Either ascending or descending is acceptable)

Listing all Customers name, Date of birth who have commented



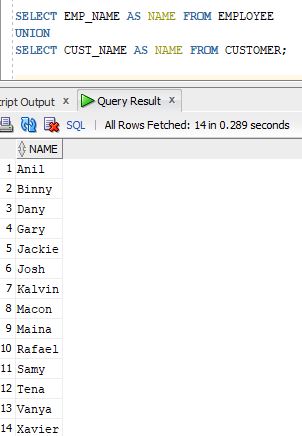
19) Create a "cascade delete" SQL statement over two tables.



If the Rest\_Id is deleted then the corresponding feedback will be deleted.Similarly, if the Cust\_Id is deleted then the corresponding feedback will be deleted.

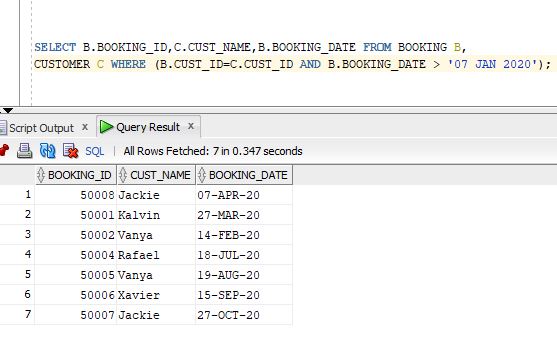
20) Demonstrate UNION statement.

Get the name list of Employee and Customer

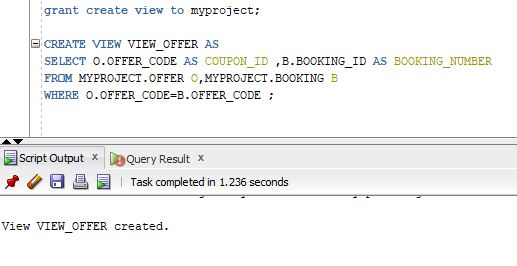


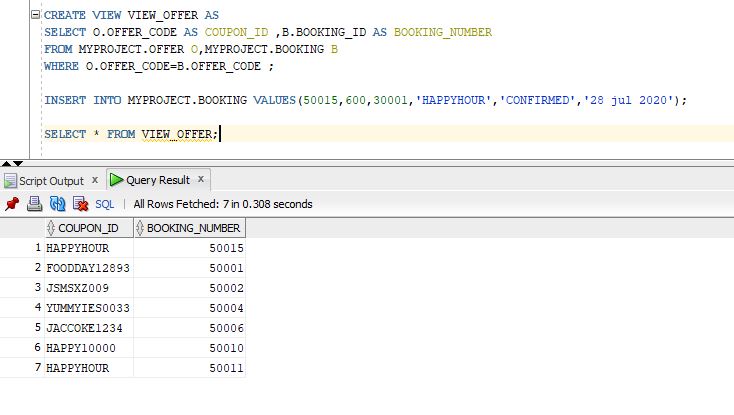
21) Demonstrate a SQL statement in which a DATE data type is subject of where statement (such as, select ... from ... where birthday > DATE)

List all the Customers who Booked after 7th January 2020

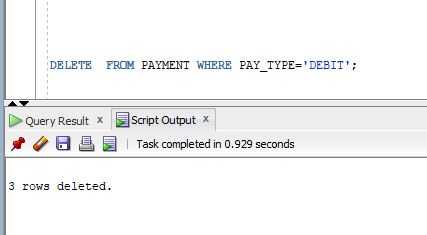


22) Demonstrate CREATE VIEW statement.





23) Delete three rows from a table.



24) Delete all rows from a table, then delete the empty table from database.

