American International University-Bangladesh

Faculty of Science and Technology
Department of Computer Science

Pastry Shop
Management System

A project in

Introduction to Database [L]

Prepared By

ID	**NAME**
20-43804-2	TONOY CHANDRA SARKER

Supervised By

JUENA AHMED NOSHIN
Faculty of Science and Technology
American International University-Bangladesh
Dhaka, Bangladesh.

Contents

Page No

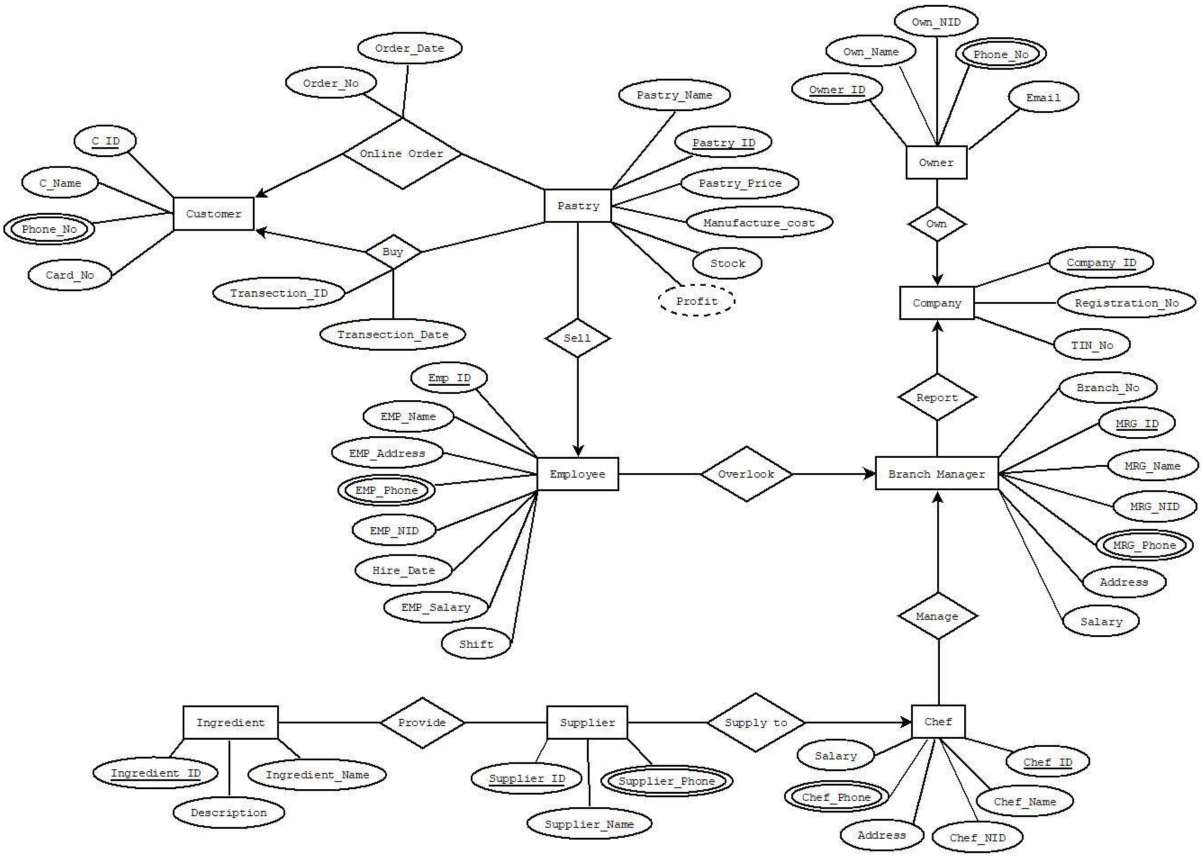
1. Introduction

This project Pastry shop Management System for Pastry Company is developed by Oracle SQL. The Pastry shop Management system is an independent application which is based on ordering and selling the pastry and generating the bill. The main principle behind the need of Pastry shop Management System is easy supervision of shop. It has user friendly & modular approach. Data storing is easier. Every information will be authentic and secured in this database system. The pastry shop Management system customer can easily order a pastry. The record of each customer is stored that is customer's name and contact details are added for reference. The employees can easily access the database and check for availability of the items. At the end of the day, report is generated to calculate the payment for each user in eachday. The report will be examined by the company. This Pastry shop Management System willreduce time consumption. This project is prepared to help the company to maintain the day- to-day operations.

2. Scenario Description

In a pastry shop, a customer may buy or order many pastries. Each customer has a unique customer Id. Also, customer has name, phone number, card no. A customer can have multiple phone numbers. A pastry can only be sold to or ordered online only one customer. While buying pastry a unique transaction Id and data is stored. When ordering online an order no and order date is stored. In the pastry shop, the pastries are identified by their names. There is pastry name, pastry Id, price, manufacture cost, stock details, profit are also stored in the management system. The profit is calculated from the cost price and sell price. Pastries are sold by employees. Where each pastry can be sold by one employee, but one employee cansell many pastries. Each employee is identified by their own unique employee Id. Also, each employee will have some personal information such as name, address, phone number, salary, shift, hire-date, NID are stored in the database. Each employee work under only one branch manager and the manager overlooked all employees. The manager has own unique Id. Also, manager data such as name, address, branch no, NID, salary are also stored in the database. Here manager and employee can have multiple phone number. Branch manager can manage many chefs in the pastry shop. Chef can be managed by only one manager. Each chef has unique ID just for identifying them. They also have name, NID, address, salary, and multiple phone number. In the pastry shop, some ingredient can be provided by multiple suppliers. Ingredient has a unique ingredient Id. Also has description and ingredient name. The supplier can supply the ingredient to only one chef. To identify the supplier, they have own unique supplier Id , name, and multiple phone number. There will be a branch manager who will report stock details, profit, amount to the company. The company has a single account. In the accounts daily profit, net profit and balance are stored. The company has a unique company ID. Also has TIN number and registration number. There are multiple owners in the company.Among the owners there is a founder. Each of the owners are identified by their owner ID. Other data such as name, NID, phone number and email are also stored in the database. The owners can have multiple phone number.

3. ER Diagram


Order_Date
Order_No
Online Order
Pastry_Name
Pastry_ID
Pastry_Price
Manufacture_cost
Stock
Profit
Pastry
C_ID
C_Name
Phone_No
Card_No
Customer
Buy
Transection_ID
Transection_Date
Sell
Own_NID
Own_Name
Phone_No
Owner_ID
Email
Owner
Own
Company
Company_ID
Registration_No
TIN_No
Report
Branch Manager
Branch_No
MRG_ID
MRG_Name
MRG_NID
MRG_Phone
Address
Salary
Manage
Emp_ID
EMP_Name
EMP_Address
EMP_Phone
EMP_NID
Hire_Date
EMP_Salary
Shift
Employee
Overlook
Ingredient
Ingredient_ID
Description
Ingredient_Name
Provide
Supplier
Supplier_ID
Supplier_Name
Supplier_Phone
Supply to
Chef
Salary
Chef_ID
Chef_Phone
Chef_Name
Address
Chef_NID

4. Normalization

Online Order

UNF

Online order (C_ID, C_name, Phone number, card no., Order_Date, Order_no, Pastry name, Pastry_ID, Pastry_Price, Manufacture_cost, stock)

1NF

Phone number is multivalued.

1. C_ID, C_name, Phone number, card no., Order_Date, Order_no Pastry name, Pastry_ID, pastry_price, Manufacture_cost, stock.

2NF

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Order_Date, Order_no.
3. Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.

3NF

There is no transitive dependency.

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Order_Date, Order_no
3. Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.

Table creation

1. C_ID, C_name, Phone number_1, Phone number_2, card no.,
2. Order_Date, Order_no., **Pastry_ID**, **C_ID**
3. Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.

Buy

UNF

Online order (C_ID, C_name, Phone number, card no., Transection_ID, Transection_Date, Pastry name, Pastry_ID, pastry_price, , stock)

1NF

Phone number is multivalued.

1. C_ID, C_name, Phone number, card no., Transection_ID, Transection_Date Pastry name, Pastry_ID, pastry_price, , stock.

2NF

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Transection_ID, Transection_Date
3. Pastry_ID, Pastry name, pastry_price, , stock.

3NF

There is no transitive dependency.

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Transection_ID, Transection_Date
3. Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.

Table creation

1. C_ID, C_name, Phone number_1, Phone number_2, card no
2. Transection_ID, Transection_Date, **Pastry_ID, C_ID**
3. Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.

Sell

UNF

Sell (Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock, EMP_ID, EMP_Name, EMP_Address, EPM_Phone, EPM_NID, Hire_Date, EMP_Salary, Shift)

1NF

EPM_Phone is multivalued.

1. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock, EMP_ID, EMP_Name, EMP_Address, EMP_phone_1, EMP_phone_2, EPM_NID, Hire_Date, EMP_Salary, Shift.

2NF

1. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock
2. EMP_ID, EMP_Name, EMP_Address, EMP_phone_1, EMP_phone_2, EPM_NID, Hire_Date, EMP_Salary, Shift.

3NF

1. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock
2. EMP_ID, EMP_Name, EMP_Address, EMP_phone_1, EMP_phone_2, EPM_NID, Hire_Date
3. EMP_Salary, Shift

Table creation

1. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock, **EMP_ID, EPM_Sal_ID.**
2. EMP_ID, EMP_Name, EMP_Address, EMP_phone_1, EMP_phone_2, EPM_NID, Hire_Date,
3. EPM_Sal_ID, EMP_Salary, Shift

Overlook

UNF

Overlook(EMP_ID, EMP_name, EMP_NID, EMP_address, EMP_phone, EMP_salary, hire-date, shift, MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone, MRG_address, salary)

1NF

EMP phone and MRG_phone is a multivalued attribute.

1. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone, EMP_salary, hire-date, shift, MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone, MRG_address, salary

2NF

1. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone_1, EMP_phone_2, EMP_salary, hire-date, shift
2. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary

3NF

1. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone_1, EMP_phone_2, hire-date.
2. EMP_salary, shift.
3. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary

Table Creation

1. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone_1, EMP_phone_2, hire-date, **MRG_ID, EPM_Sal_ID.**
2. EPM_Sal_ID, EMP_salary, shift.
3. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary

Report

UNF

Report(Branch_No, MRG_ID, MRG_Name, MRG_NID, MRG_Phone, Address, Salary, Company_ID , Registration_No, TIN_No,)

1NF

Phone is a multivalued attribute.

1. Branch_No, MRG_ID, MRG_Name, MRG_NID, MRG_Phone, Address, Salary, Company_ID , Registration_No, TIN_No.

2NF

1. Branch_No, MRG_ID, MRG_Name, MRG_NID, MRG_Phone_1, MRG_Phone_2, Address, Salary.
2. Company_ID ,Registration_No, TIN_No.

3NF

There is no transitive dependency.

1. Branch_No, MRG_ID, MRG_Name, MRG_NID, MRG_Phone_1, MRG_Phone_2, Address, Salary.
2. Company_ID, Registration_No, TIN_No.

Table creation

1. Branch_No, MRG_ID, MRG_Name, MRG_NID, MRG_Phone_1, MRG_Phone_2, Address, Salary, **Company_ID_**.
2. Company_ID, Registration_No, TIN_No.

Own

UNF

Own(Owner_ID, Owner_Name, Owner_NID, Phone_No, Email, Company_ID, Registration_No, TIN_No)

1NF

Phone is a multivalued attribute.

1. Owner_ID, Owner_Name, Owner_NID, Phone_No, Email, Company_ID, Registration_No, TIN_No.

2NF

1. Owner_ID, Owner_Name, Owner_NID, Phone_No_1 , Phone_No_2, Email.
2. Company_ID, Registration_No, TIN_No.

3NF

There is no transitive dependency.

1. Owner_ID, Owner_Name, Owner_NID, Phone_No_1, Phone_No_2, Email.
2. Company_ID, Registration_No, TIN_No.

Table creation

1. Owner_ID, Owner_Name, Owner_NID, Phone_No_1, Phone_No_2, Email, **Company_ID**.
2. Company_ID, Registration_No, TIN_No.

Manage

UNF

Manage(MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone, MRG_address, salary, Chef_ID, Chef_name, Chef_NID, address, Chef_Phone, Salary)

1NF

MRG phone and Chef phone numbers are multivalued.

1. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone, MRG_address, salary, Chef_ID, Chef_name, Chef_NID, address, Chef_Phone, Salary

2NF

1. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary.

2. Chef_ID, Chef_name, Chef_NID address, Chef_Phone_1, Chef_Phone_2, Salary.

3NF

There is no transitive dependency.

1. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary.

2. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary.

Table creation

1. MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary.

2. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary, **MRG_ID.**

Supply to

UNF

Supply to (Supplier_ID, Supplier_Name, Supplier_Phone, Chef_ID, Chef_name, Chef_NID, address, Chef_Phone, Salary)

1NF

MRG phone and Chef phone numbers are multivalued.

1. Supplier_ID, Supplier_Name, Supplier_Phone, Chef_ID, Chef_name, Chef_NID, address, Chef_Phone, Salary

2NF

1. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.
2. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary.

3NF

There is no transitive dependency.

1. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.
2. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary.

Table creation

1. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2, **Chef_ID**.
2. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary.

Provide

UNF

Provide(Ingredient_ID, Ingredient_Name, Description, Supplier_ID, Supplier_Name, Supplier_Phone)

1NF

Chef phone numbers is multivalued.

1. Ingredient_ID, Ingredient_Name, Description, Supplier_ID, Supplier_Name, Supplier_Phone.

2NF

1. Ingredient_ID, Ingredient_Name, Description.
2. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.

3NF

There is no transitive dependency.

1. Ingredient_ID, Ingredient_Name, Description.
2. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.

Table creation

1. Ingredient_ID, Ingredient_Name, Description.
2. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.
3. **Ingredient_ID, Supplier_ID.**

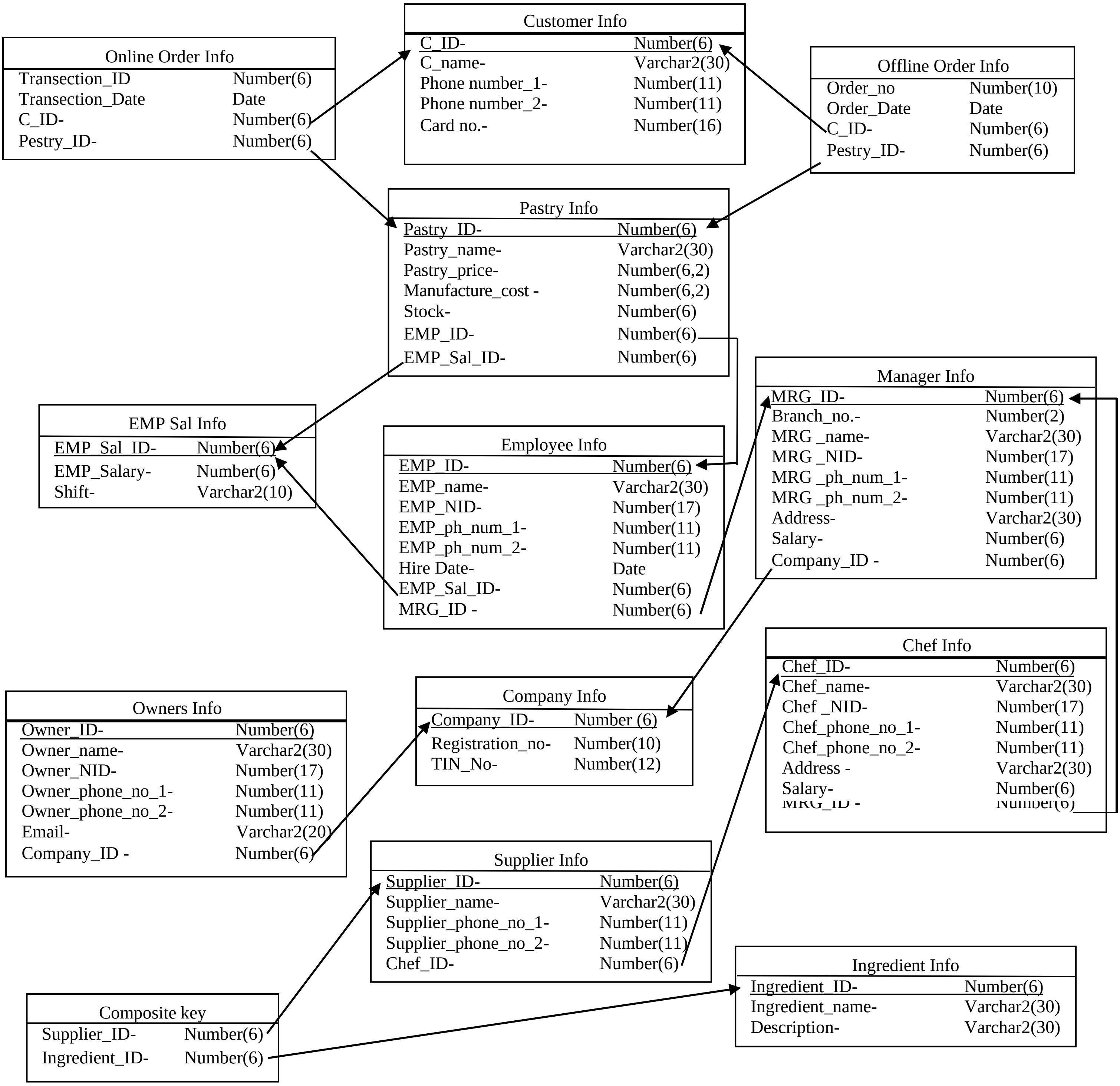
Temporary Table

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Order_Date, Order_no., **Pastry_ID**, **C_ID**
3. ~~Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.~~
4. ~~C_ID, C_name, Phone number_1, Phone number_2, card no.~~
5. Transection_ID, Transection_Date, **Pastry_ID, C_ID**
6. ~~Pastry_ID, Pastry name, pastry_price, Manufacture_cost, stock.~~
7. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock, **EMP_ID, EPM_Sal_ID.**
8. ~~EMP_ID, EMP_Name, EMP_Address, EMP_phone_1, EMP_phone_2, EPM_NID, Hire_Date~~
9. EPM_Sal_ID, EMP_Salary, Shift
10. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone_1, EMP_phone_2, hire-date, **MRG_ID, EPM_Sal_ID.**
11. ~~EPM_Sal_ID, EMP_salary, shift.~~
12. ~~MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary.~~
13. MRG_ID, Branch_No, MRG_Name, MRG_NID, MRG_Phone_1, MRG_Phone_2, Address, Salary, **Company_ID**.
14. Company_ID, Registration_No, TIN_No.
15. Owner_ID, Owner_Name, Owner_NID, Phone_No_1, Phone_No_2, Email, **Company_ID**.
16. ~~Company_ID, Registration_No, TIN_No.~~
17. ~~MRG_ID, Branch no., MRG_name, MRG_NID, MRG_phone_1, MRG_phone_2, MRG_address, salary.~~
18. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary, **MRG_ID.**
19. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2, **Chef_ID**.
20. ~~Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary.~~
21. Ingredient_ID, Ingredient_Name, Description.
22. ~~Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2.~~
23. **Ingredient_ID, Supplier_ID.**

Final Table

1. C_ID, C_name, Phone number_1, Phone number_2, card no.
2. Order_Date, Order_no., **Pastry_ID**, **C_ID.**
3. Transection_ID, Transection_Date, **Pastry_ID, C_ID.**
4. Pastry_ID, Pastry_Name, pastry_price, Manufacture_cost, stock, **EMP_ID, EPM_Sal_ID.**
5. EPM_Sal_ID, EMP_Salary, Shift.
6. EMP_ID, EMP name, EMP_NID, EMP_address, EMP_phone_1, EMP_phone_2, hire-date, **MRG_ID, EPM_Sal_ID**.
7. MRG_ID, Branch_No, MRG_Name, MRG_NID, MRG_Phone_1, MRG_Phone_2, Address, Salary, **Company_ID**.
8. Company_ID, Registration_No, TIN_No.
9. Owner_ID, Owner_Name, Owner_NID, Phone_No_1, Phone_No_2, Email, **Company_ID**.
10. Chef_ID, Chef_name, Chef_NID, address, Chef_Phone_1, Chef_Phone_2, Salary, **MRG_ID.**
11. Supplier_ID, Supplier_Name, Supplier_Phone_1, Supplier_Phone_2, **Chef_ID**.
12. Ingredient_ID, Ingredient_Name, Description.
13. **Ingredient_ID, Supplier_ID.**

5. Schema Diagram



Customer Info

C_ID-	Number(6)
C_name-	Varchar2(30)
Phone number_1-	Number(11)
Phone number_2-	Number(11)
Card no.-	Number(16)

Online Order Info

Transection_ID	Number(6)
Transection_Date	Date
C_ID-	Number(6)
Pestry_ID-	Number(6)

Offline Order Info

Order_no	Number(10)
Order_Date	Date
C_ID-	Number(6)
Pestry_ID-	Number(6)

Pastry Info

Pastry_ID-	Number(6)
Pastry_name-	Varchar2(30)
Pastry_price-	Number(6,2)
Manufacture_cost -	Number(6,2)
Stock-	Number(6)
EMP_ID-	Number(6)
EMP_Sal_ID-	Number(6)

Manager Info

MRG_ID-	Number(6)
Branch_no.-	Number(2)
MRG _name-	Varchar2(30)
MRG _NID-	Number(17)
MRG _ph_num_1-	Number(11)
MRG _ph_num_2-	Number(11)
Address-	Varchar2(30)
Salary-	Number(6)
Company_ID -	Number(6)

EMP Sal Info

EMP_Sal_ID-	Number(6)
EMP_Salary-	Number(6)
Shift-	Varchar2(10)

Employee Info

EMP_ID-	Number(6)
EMP_name-	Varchar2(30)
EMP_NID-	Number(17)
EMP_ph_num_1-	Number(11)
EMP_ph_num_2-	Number(11)
Hire Date-	Date
EMP_Sal_ID-	Number(6)
MRG_ID -	Number(6)

Chef Info

Chef_ID-	Number(6)
Chef_name-	Varchar2(30)
Chef _NID-	Number(17)
Chef_phone_no_1-	Number(11)
Chef_phone_no_2-	Number(11)
Address -	Varchar2(30)
Salary-	Number(6)
MRG_ID -	Number(6)

Owners Info

Owner_ID-	Number(6)
Owner_name-	Varchar2(30)
Owner_NID-	Number(17)
Owner_phone_no_1-	Number(11)
Owner_phone_no_2-	Number(11)
Email-	Varchar2(20)
Company_ID -	Number(6)

Company Info

Company_ID-	Number (6)
Registration_no-	Number(10)
TIN_No-	Number(12)

Supplier Info

Supplier_ID-	Number(6)
Supplier_name-	Varchar2(30)
Supplier_phone_no_1-	Number(11)
Supplier_phone_no_2-	Number(11)
Chef_ID-	Number(6)

Ingredient Info

Ingredient_ID-	Number(6)
Ingredient_name-	Varchar2(30)
Description-	Varchar2(30)

Composite key

Supplier_ID-	Number(6)
Ingredient_ID-	Number(6)

6. Table Creation

User creation**(from system)**:

1.
   ```
   CREATE USER pastry IDENTIFIED BY nightqueen;
   GRANT UNLIMITED TABLESPACE TO pastry;
   GRANT create table, create sequence, create view to pastry;
   ```

Table creation with SQL **(from pastry)**:

1.
   ```
   create table customer_info
   (
     C_ID number(6) PRIMARY KEY,
     C_name varchar2(30) NOT NULL,
     Phone_number_1 number(11) NOT NULL,
     Phone_number_2 number(11),
     Card_no number(16) NOT NULL
   );
   ```

2.
   ```
   create table Pastry_Info
   (Pastry_ID Number(6) primary key,
    Pastry_Name Varchar2(30) not null,
    Pastry_price Number(6,2) not null,
    Manufacture_cost Number(6,2) not null,
    Stock Number(6) not null
   );
   ```

3.
   ```
   CREATE TABLE Employee_info
   (
   EMP_ID number(6) primary key,
   EMP_Name varchar2(30) NOT NULL,
   EMP_NID number(17) NOT NULL,
   ph_num_1 number(11) NOT NULL,
   ph_num_2 number(11),
   hire_date date NOT NULL,
   emp_sal_id number(6) NOT NULL
   );
   ```

4.
   ```
   CREATE TABLE online_Order_Info
   (
   Transection_ID number(6) NOT NULL,
   Transection_Date date NOT NULL
   );
   ```

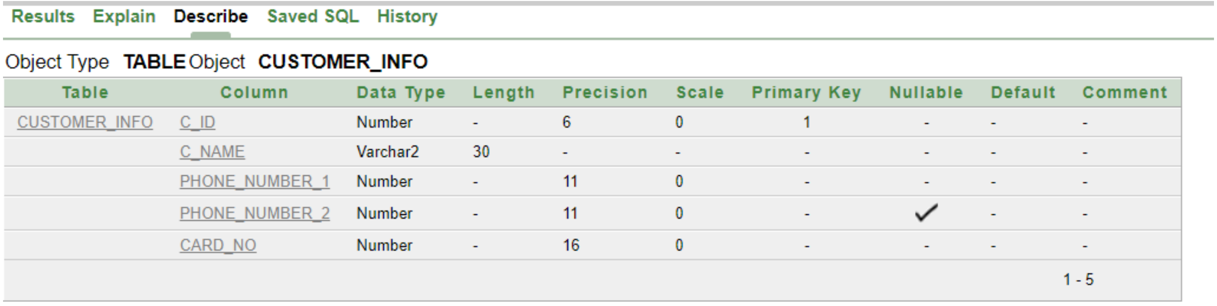
```
5.  CREATE TABLE Offline_Order_Info
    (
    Order_no Number(10) NOT NULL,
    Order_date date NOT NULL
    );
6.  CREATE TABLE EMP_SAL_INFO
    (
    EMP_SAL_ID Number(6) primary key,
    EMP_SAL Number(6) not null,
    shift Varchar2(10)
    );
7.  CREATE TABLE company_info
    (
    company_id Number(6) primary key,
    Registation_No Number(10) not null,
    TIN_NO Number(10)
    );
8.  create table Manager_Info
    (MRG_ID Number(6) primary key,
    Branch_no Number(2) not null,
    MRG_name Varchar2(30) not null,
    MRG_NID Number(17),
     MRG_ph_num_1 Number(11) not null,
     MRG_ph_num_2  Number(11),
     Address Varchar2(30) not null,
     Salary Number(6) not null
    );
9.  CREATE TABLE Owners_Info
    (
    Owner_ID Number(6) primary key,
    Owner_name Varchar2(30) NOT NULL,
    Owner_NID Number(17) NOT NULL,
    Owner_phone_1 Number(11) NOT NULL,
    Owner_phone_2 Number(11),
    Email Varchar2(20) NOT NULL
    );
10. CREATE TABLE Supplier_Info
    (
    Supplier_ID Number(6) primary key,
    Supplier_name Varchar2(30) NOT NULL,
    Supplier_Phone_no1 Number(11) NOT NULL,
    Supplier_Phone_no2 Number(11) NOT NULL
    );
```

```
11. create table Ingredient_Info
    (
    Ingredient_ID Number(6) primary key,
     Ingredient_name Varchar2(30) not null,
     Description Varchar2(30) not null
    );
12. CREATE TABLE Chef_info
    (
     chef_ID Number(6) primary key,
     chef_name Varchar2(30) not null,
     chef_NID Number(17) not null,
     chef_phone_no_1 Number(11) not null,
     chef_phone_no_2  Number(11),
     Address Varchar2(30) not null,
     Salary Number(6) not null
    );
13. create table Common
    (Supplier_ID Number(6),
    Ingredient_ID Number(6));
```

Add foreign key**(from pastry)**:

```
 1.  Alter table online_Order_Info add(C_ID number(6));
     Alter table online_Order_Info
     add constraint fk_online_Order_Info_c_id foreign key(C_id)
     references customer_info (c_id);
 2.  Alter table online_Order_Info add(Pastry_ID number(6));
    Alter table online_Order_Info
    add constraint fk_online_Order_Pestry_INFO foreign key(Pastry_ID)
    references Pastry_INFO (Pastry_ID);
3.   Alter table Offline_Order_Info add(C_ID number(6), Pastry_ID number(6));
    Alter table Offline_Order_Info
    add constraint fk_Offline_Order_Info_c_id foreign key(C_id)
    references customer_info (c_id);
    Alter table Offline_Order_Info
    add constraint fk_Offline_Order_Pestry_INFO foreign key(Pastry_ID)
    references Pastry_INFO (Pastry_ID);
4.  Alter table Pastry_info add (EMP_ID number(6), EMP_Sal_ID number(6));
    Alter table Pastry_Info
    add constraint fk_Pastry_Info_EMP_ID foreign key(EMP_ID)
    references Employee_info (EMP_ID);
    Alter table Pastry_Info
    add constraint fk_Pastry_EMP_Sal foreign key(EMP_Sal_ID)
    references EMP_SAL_INFO (EMP_Sal_ID);
 5. Alter table Employee_Info add(MRG_ID number(6));
    Alter table Employee_Info
    add constraint fk_Employee_Info_EMP_sal_ID foreign key(EMP_Sal_ID)
    references EMP_Sal_Info (EMP_Sal_ID);
    Alter table Employee_Info
    add constraint fk_Employee_Info_MRG_ID foreign key(MRG_ID)
    references Manager_Info (MRG_ID);

 6. Alter table Manager_Info add(Company_ID number(6));
    Alter table Manager_Info
    add constraint fk_Manager_Info_Company_ID foreign key(Company_ID)
    references Company_Info (Company_ID);

 7. Alter table Owners_Info add(Company_ID number(6));
    Alter table Owners_Info
    add constraint fk_Owners_Info_Company_ID foreign key(Company_ID)
    references Company_Info (Company_ID);

 8. Alter table Chef_Info add(MRG_ID number(6));
    Alter table Chef_Info
    add constraint fk_Chef_Info_MRG_ID foreign key(MRG_ID)
    references Manager_Info (MRG_ID);

 9. Alter table Common
    add constraint fk_Common_Supplier_ID foreign key(Supplier_ID)
    references Supplier_Info (Supplier_ID);
    Alter table Common
    add constraint fk_Common_Ingredient_ID foreign key(Ingredient_ID)
    references Ingredient_Info (Ingredient_ID);

10.  Alter table Supplier_Info add(Chef_ID number(6));
    Alter table Supplier_Info
    add constraint fk_Supplier_Info_Chef_ID foreign key(Chef_ID)
    references Chef_Info (Chef_ID);
```

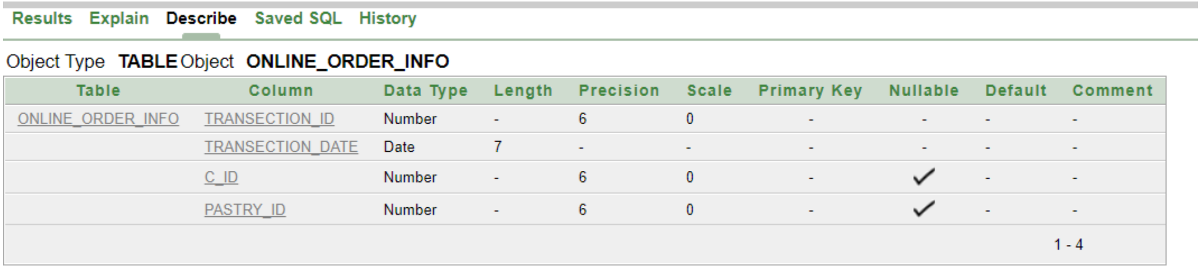
Screen Shot of Created Table**(from pastry)**:



Object Type **TABLE** Object **CUSTOMER_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CUSTOMER_INFO	C_ID	Number	-	6	0	1	-	-	-
	C_NAME	Varchar2	30	-	-	-	-	-	-
	PHONE_NUMBER_1	Number	-	11	0	-	-	-	-
	PHONE_NUMBER_2	Number	-	11	0	-	✓	-	-
	CARD_NO	Number	-	16	0	-	-	-	-

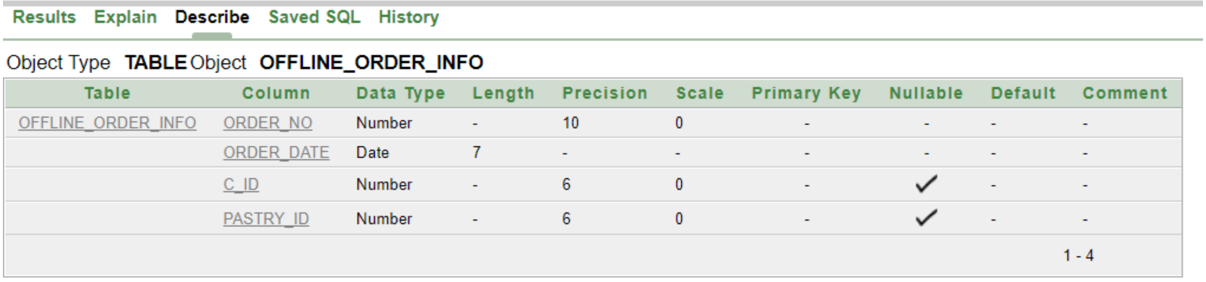
1 - 5



Object Type **TABLE** Object **ONLINE_ORDER_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
ONLINE_ORDER_INFO	TRANSECTION_ID	Number	-	6	0	-	-	-	-
	TRANSECTION_DATE	Date	7	-	-	-	-	-	-
	C_ID	Number	-	6	0	-	✓	-	-
	PASTRY_ID	Number	-	6	0	-	✓	-	-

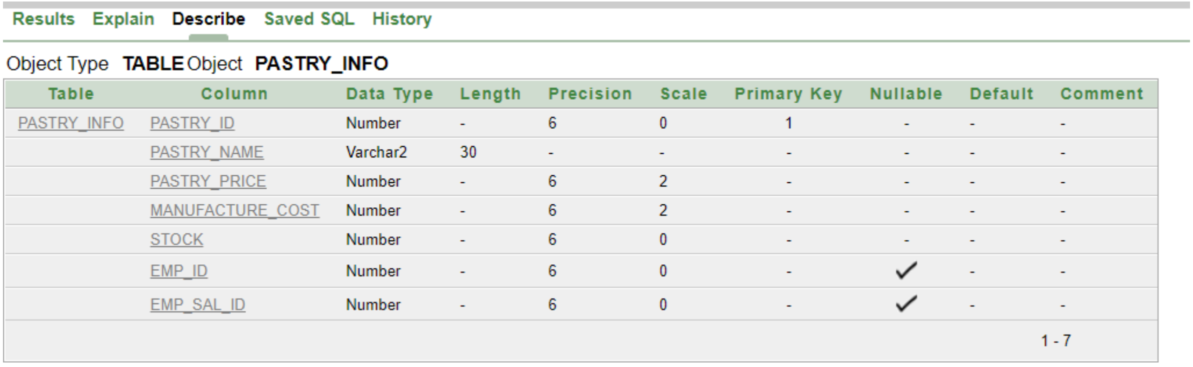
1 - 4



Object Type **TABLE** Object **OFFLINE_ORDER_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OFFLINE_ORDER_INFO	ORDER_NO	Number	-	10	0	-	-	-	-
	ORDER_DATE	Date	7	-	-	-	-	-	-
	C_ID	Number	-	6	0	-	✓	-	-
	PASTRY_ID	Number	-	6	0	-	✓	-	-

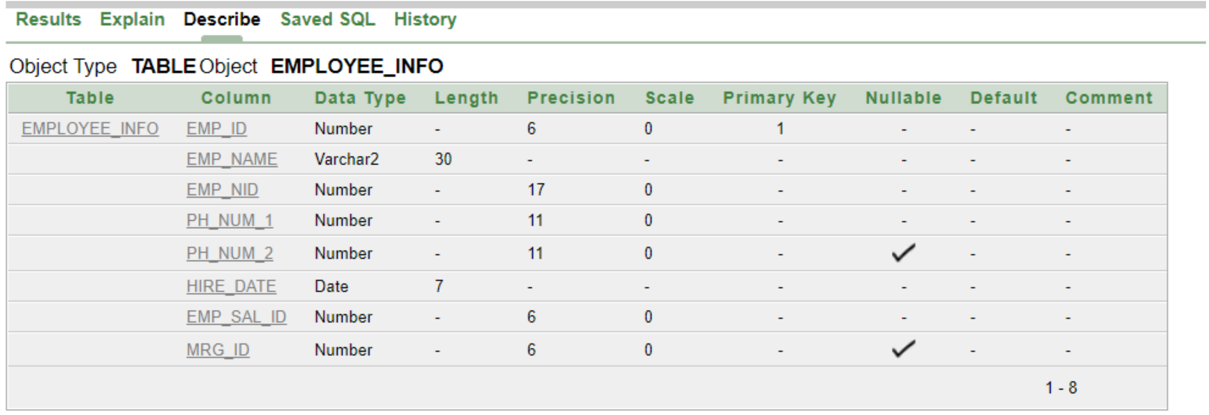
1 - 4



Object Type **TABLE** Object **PASTRY_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
PASTRY_INFO	PASTRY_ID	Number	-	6	0	1	-	-	-
	PASTRY_NAME	Varchar2	30	-	-	-	-	-	-
	PASTRY_PRICE	Number	-	6	2	-	-	-	-
	MANUFACTURE_COST	Number	-	6	2	-	-	-	-
	STOCK	Number	-	6	0	-	-	-	-
	EMP_ID	Number	-	6	0	-	✓	-	-
	EMP_SAL_ID	Number	-	6	0	-	✓	-	-

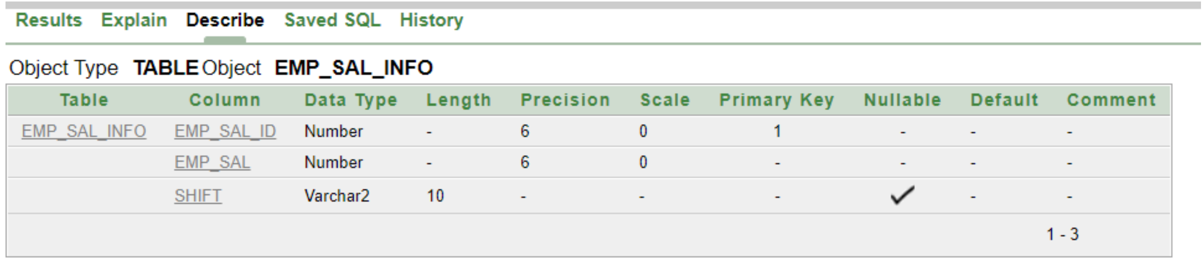
1 - 7



Object Type **TABLE** Object **EMPLOYEE_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEE_INFO	EMP_ID	Number	-	6	0	1	-	-	-
	EMP_NAME	Varchar2	30	-	-	-	-	-	-
	EMP_NID	Number	-	17	0	-	-	-	-
	PH_NUM_1	Number	-	11	0	-	-	-	-
	PH_NUM_2	Number	-	11	0	-	✓	-	-
	HIRE_DATE	Date	7	-	-	-	-	-	-
	EMP_SAL_ID	Number	-	6	0	-	-	-	-
	MRG_ID	Number	-	6	0	-	✓	-	-

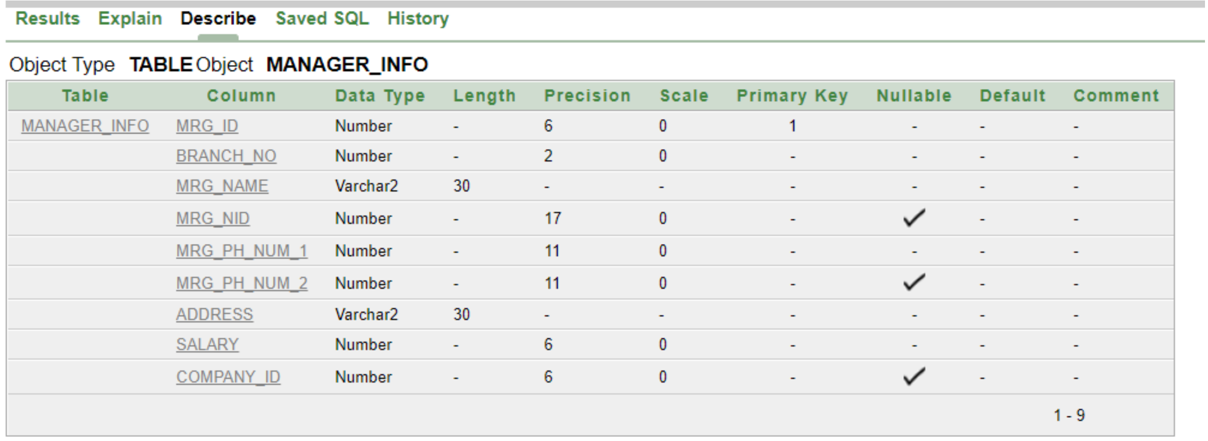
1 - 8



Object Type **TABLE** Object **EMP_SAL_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMP_SAL_INFO	EMP_SAL_ID	Number	-	6	0	1	-	-	-
	EMP_SAL	Number	-	6	0	-	-	-	-
	SHIFT	Varchar2	10	-	-	-	✓	-	-

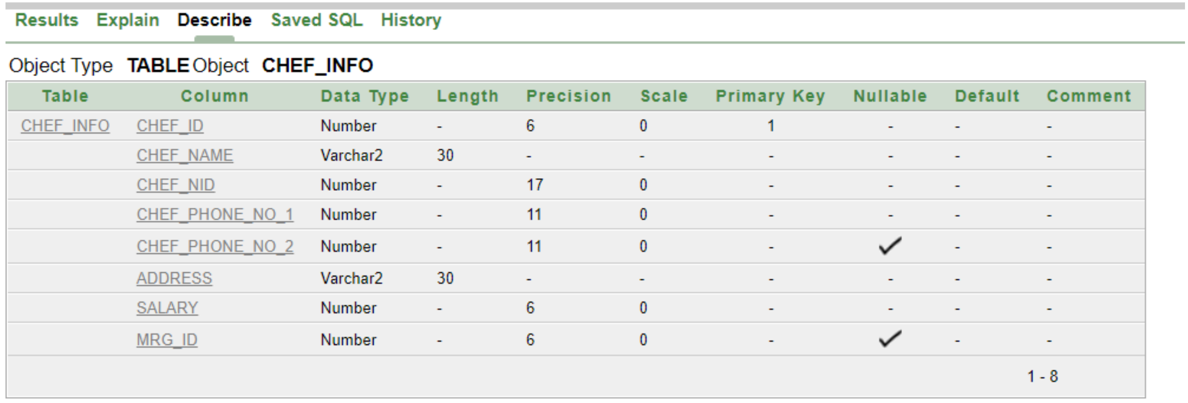
1 - 3



Object Type **TABLE** Object **MANAGER_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
MANAGER_INFO	MRG_ID	Number	-	6	0	1	-	-	-
	BRANCH_NO	Number	-	2	0	-	-	-	-
	MRG_NAME	Varchar2	30	-	-	-	-	-	-
	MRG_NID	Number	-	17	0	-	✓	-	-
	MRG_PH_NUM_1	Number	-	11	0	-	-	-	-
	MRG_PH_NUM_2	Number	-	11	0	-	✓	-	-
	ADDRESS	Varchar2	30	-	-	-	-	-	-
	SALARY	Number	-	6	0	-	-	-	-
	COMPANY_ID	Number	-	6	0	-	✓	-	-

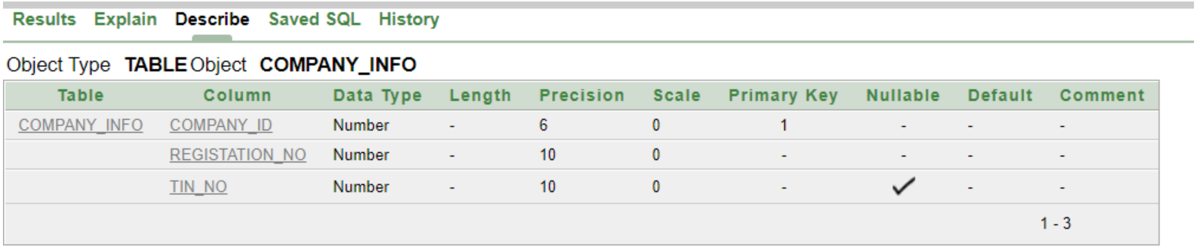
1 - 9



Object Type **TABLE** Object **CHEF_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
CHEF_INFO	CHEF_ID	Number	-	6	0	1	-	-	-
	CHEF_NAME	Varchar2	30	-	-	-	-	-	-
	CHEF_NID	Number	-	17	0	-	-	-	-
	CHEF_PHONE_NO_1	Number	-	11	0	-	-	-	-
	CHEF_PHONE_NO_2	Number	-	11	0	-	✓	-	-
	ADDRESS	Varchar2	30	-	-	-	-	-	-
	SALARY	Number	-	6	0	-	-	-	-
	MRG_ID	Number	-	6	0	-	✓	-	-

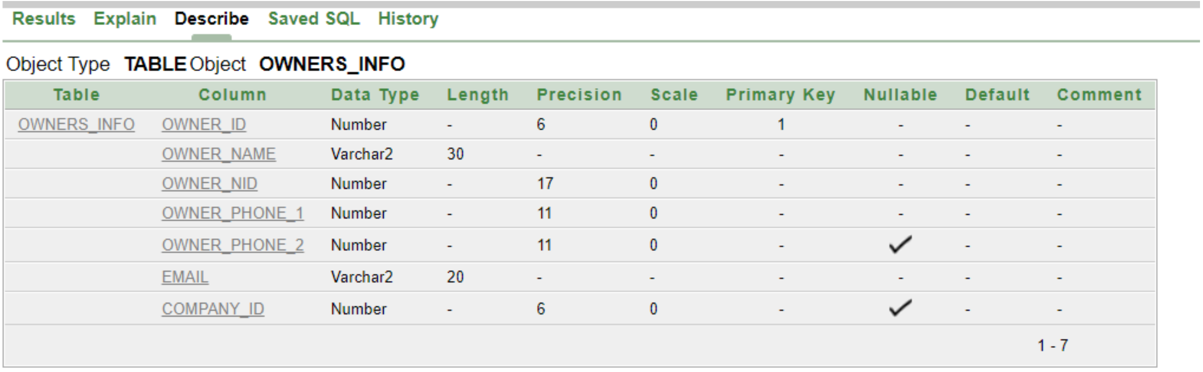
1 - 8



Object Type **TABLE** Object **COMPANY_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
COMPANY_INFO	COMPANY_ID	Number	-	6	0	1	-	-	-
	REGISTATION_NO	Number	-	10	0	-	-	-	-
	TIN_NO	Number	-	10	0	-	✓	-	-

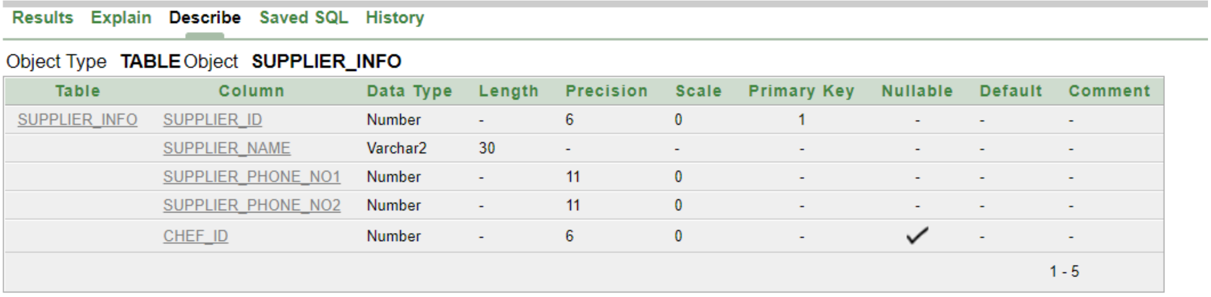
1 - 3



Object Type **TABLE** Object **OWNERS_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
OWNERS_INFO	OWNER_ID	Number	-	6	0	1	-	-	-
	OWNER_NAME	Varchar2	30	-	-	-	-	-	-
	OWNER_NID	Number	-	17	0	-	-	-	-
	OWNER_PHONE_1	Number	-	11	0	-	-	-	-
	OWNER_PHONE_2	Number	-	11	0	-	✓	-	-
	EMAIL	Varchar2	20	-	-	-	-	-	-
	COMPANY_ID	Number	-	6	0	-	✓	-	-

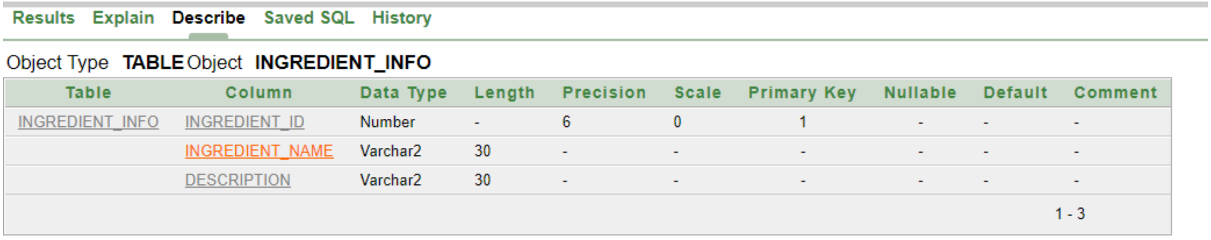
1 - 7



Object Type **TABLE** Object **SUPPLIER_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
SUPPLIER_INFO	SUPPLIER_ID	Number	-	6	0	1	-	-	-
	SUPPLIER_NAME	Varchar2	30	-	-	-	-	-	-
	SUPPLIER_PHONE_NO1	Number	-	11	0	-	-	-	-
	SUPPLIER_PHONE_NO2	Number	-	11	0	-	-	-	-
	CHEF_ID	Number	-	6	0	-	✓	-	-

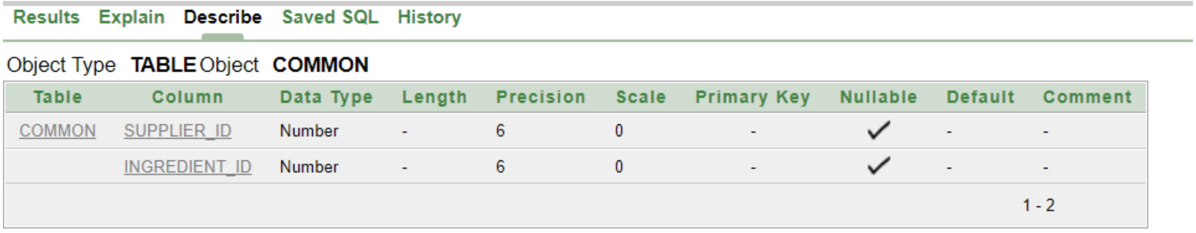
1 - 5



Object Type **TABLE** Object **INGREDIENT_INFO**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
INGREDIENT_INFO	INGREDIENT_ID	Number	-	6	0	1	-	-	-
	INGREDIENT_NAME	Varchar2	30	-	-	-	-	-	-
	DESCRIPTION	Varchar2	30	-	-	-	-	-	-

1 - 3



Object Type **TABLE** Object **COMMON**

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
COMMON	SUPPLIER_ID	Number	-	6	0	-	✓	-	-
	INGREDIENT_ID	Number	-	6	0	-	✓	-	-

1 - 2

Add Sequences **(from pastry)**:

1.
   ```
   CREATE SEQUENCE customer_info_c_id
   INCREMENT BY 1
   START WITH 1000
   MAXVALUE 1999
   NOCACHE
   NOCYCLE;
   ```

2.
   ```
   CREATE SEQUENCE pastry_info_pastry_id
   INCREMENT BY 1
   START WITH 2000
   MAXVALUE 2999
   NOCACHE
   NOCYCLE;
   ```

3.
   ```
   CREATE SEQUENCE Employee_info_emp_id
   INCREMENT BY 1
   START WITH 3000
   MAXVALUE 3999
   NOCACHE
   NOCYCLE;
   ```

4.
   ```
   CREATE SEQUENCE Manager_info_MRG_id
   INCREMENT BY 1
   START WITH 4000
   MAXVALUE 4999
   NOCACHE
   NOCYCLE;
   ```

5.
   ```
   CREATE SEQUENCE emp_sal_info_emp_sal_id
   INCREMENT BY 1
   START WITH 5000
   MAXVALUE 5999
   NOCACHE
   NOCYCLE;
   ```

6.
   ```
   CREATE SEQUENCE owners_info_owners_id
   INCREMENT BY 1
   START WITH 6000
   MAXVALUE 6999
   NOCACHE
   NOCYCLE;
   ```

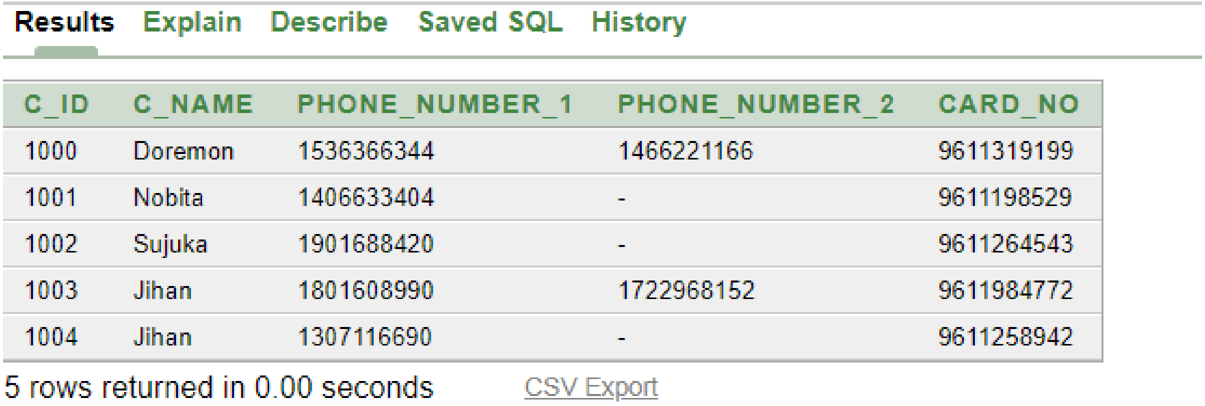
7.
   ```
   CREATE SEQUENCE chef_info_chef_id
   INCREMENT BY 1
   START WITH 7000
   MAXVALUE 7999
   NOCACHE
   NOCYCLE;
   ```

8. CREATE SEQUENCE supplier_info_supplier_id
   INCREMENT BY 1
   START WITH 8000
   MAXVALUE 8999
   NOCACHE
   NOCYCLE;
9. CREATE SEQUENCE Ingredient_info_Ingredient_id
   INCREMENT BY 1
   START WITH 9000
   MAXVALUE 9999
   NOCACHE
   NOCYCLE;

7. Data insertion(from pastry):

Customer info Table:

1. insert into customer_info values(customer_info_c_id.nextval, 'Doremon', 01536366344, 01466221166, 9611319199);
2. insert into customer_info values(customer_info_c_id.nextval, 'Nobita', 01406633404," , 9611198529);
3. insert into customer_info values(customer_info_c_id.nextval, 'Sujuka', 01901688420," , 9611264543);
4. insert into customer_info values(customer_info_c_id.nextval, 'Jihan', 01801608990,'01722968152' , 9611984772);
5. insert into customer_info values(customer_info_c_id.nextval, 'Jihan', 01307116690,' ' , 9611258942);

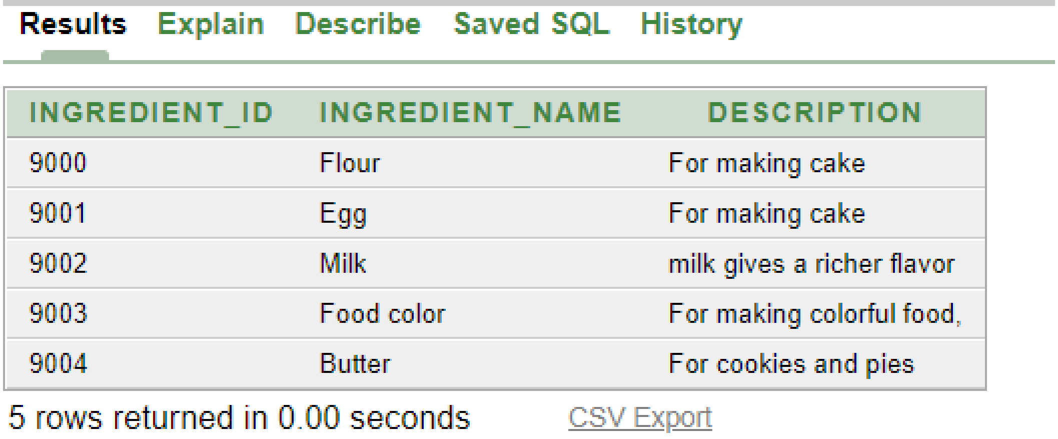


C_ID	C_NAME	PHONE_NUMBER_1	PHONE_NUMBER_2	CARD_NO
1000	Doremon	1536366344	1466221166	9611319199
1001	Nobita	1406633404	-	9611198529
1002	Sujuka	1901688420	-	9611264543
1003	Jihan	1801608990	1722968152	9611984772
1004	Jihan	1307116690	-	9611258942

5 rows returned in 0.00 seconds

Ingredient info Table:

1. insert into Ingredient_info values(Ingredient_info_Ingredient_id.nextval, 'Flour', 'For making cake');
2. insert into Ingredient_info values(Ingredient_info_Ingredient_id.nextval, 'Egg', 'For making cake');
3. insert into Ingredient_info values(Ingredient_info_Ingredient_id.nextval, 'Milk', 'milk gives a richer flavor');
4. insert into Ingredient_info values(Ingredient_info_Ingredient_id.nextval, 'Food color', 'For making colorful food,');
5. insert into Ingredient_info values(Ingredient_info_Ingredient_id.nextval, 'Butter', 'For cookies and pies');

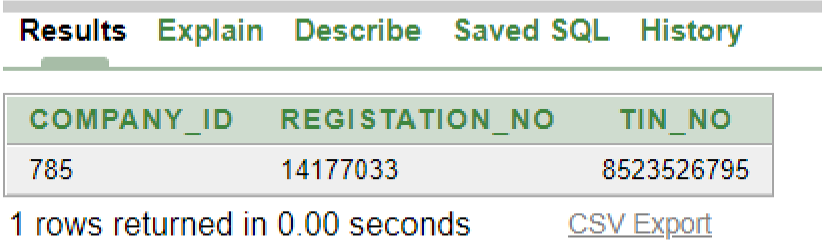


INGREDIENT_ID	INGREDIENT_NAME	DESCRIPTION
9000	Flour	For making cake
9001	Egg	For making cake
9002	Milk	milk gives a richer flavor
9003	Food color	For making colorful food,
9004	Butter	For cookies and pies

5 rows returned in 0.00 seconds CSV Export

Company info Table:

1. insert into Company_info values('785', '14177033', '8523526795');

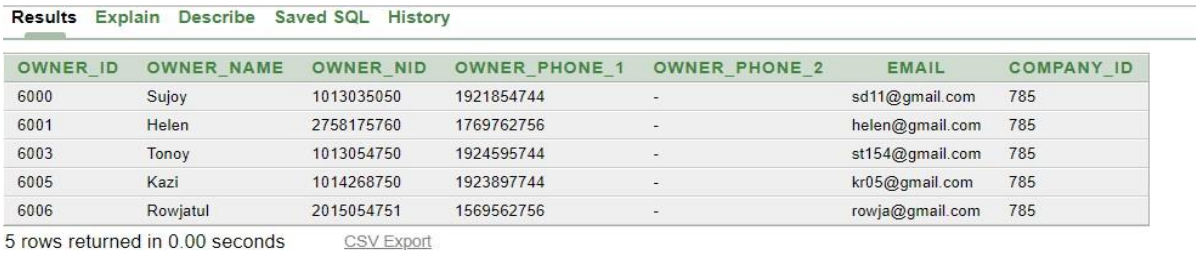


COMPANY_ID	REGISTATION_NO	TIN_NO
785	14177033	8523526795

1 rows returned in 0.00 seconds CSV Export

Owners info Table:

1. insert into Owners_info values(owners_info_owners_id.NEXTVAL, 'Sujoy', 1013035050 , 01921854744, '', 'sd11@gmail.com', 785);
2. insert into Owners_info values(owners_info_owners_id.NEXTVAL, 'Helen', 2758175760 , 01769762756, '', 'helen@gmail.com', 785);
3. insert into Owners_info values(owners_info_owners_id.NEXTVAL, 'Tonoy', 1013054750 , 01924595744, '', 'st154@gmail.com', 785);
4. insert into Owners_info values(owners_info_owners_id.NEXTVAL, 'Kazi', 1014268750 , 01923897744, '', 'kr05@gmail.com', 785);
5. insert into Owners_info values(owners_info_owners_id.NEXTVAL, 'Rowjatul', 2015054751, 01569562756, '', 'rowja@gmail.com', 785);

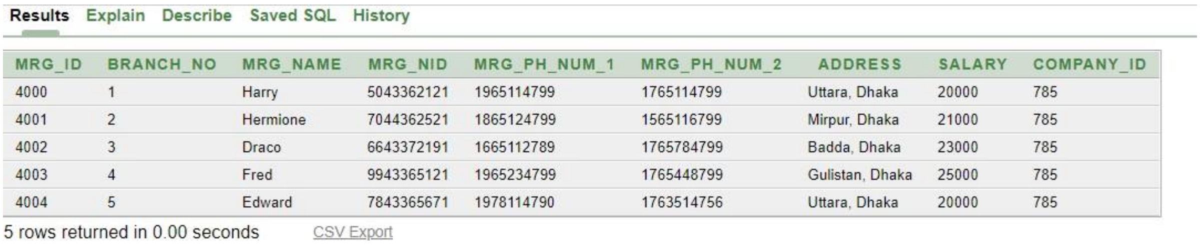


OWNER_ID	OWNER_NAME	OWNER_NID	OWNER_PHONE_1	OWNER_PHONE_2	EMAIL	COMPANY_ID
6000	Sujoy	1013035050	1921854744	-	sd11@gmail.com	785
6001	Helen	2758175760	1769762756	-	helen@gmail.com	785
6003	Tonoy	1013054750	1924595744	-	st154@gmail.com	785
6005	Kazi	1014268750	1923897744	-	kr05@gmail.com	785
6006	Rowjatul	2015054751	1569562756	-	rowja@gmail.com	785

5 rows returned in 0.00 seconds

Manager info Table:

1. insert into Manager_info values(Manager_info_MRG_id.NEXTVAL, 1, 'Harry', 5043362121, 01965114799, 01765114799,'Uttara, Dhaka', 20000, 785);
2. insert into Manager_info values(Manager_info_MRG_id.NEXTVAL, 2, 'Hermione', 7044362521, 01865124799, 01565116799,'Mirpur, Dhaka', 21000, 785);
3. insert into Manager_info values(Manager_info_MRG_id.NEXTVAL, 3, 'Draco', 6643372191, 01665112789, 01765784799,'Badda, Dhaka', 23000, 785);
4. insert into Manager_info values(Manager_info_MRG_id.NEXTVAL, 4, 'Fred', 9943365121, 01965234799, 01765448799,'Gulistan, Dhaka', 25000, 785);
5. insert into Manager_info values(Manager_info_MRG_id.NEXTVAL, 5, 'Edward', 7843365671, 01978114790, 01763514756,'Uttara, Dhaka', 20000, 785);

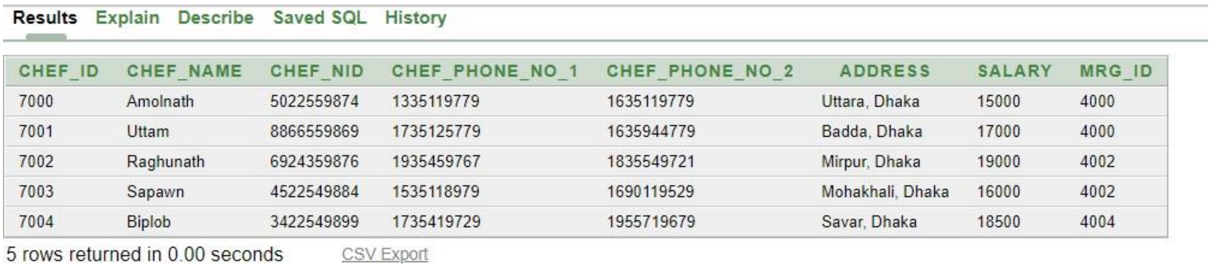


MRG_ID	BRANCH_NO	MRG_NAME	MRG_NID	MRG_PH_NUM_1	MRG_PH_NUM_2	ADDRESS	SALARY	COMPANY_ID
4000	1	Harry	5043362121	1965114799	1765114799	Uttara, Dhaka	20000	785
4001	2	Hermione	7044362521	1865124799	1565116799	Mirpur, Dhaka	21000	785
4002	3	Draco	6643372191	1665112789	1765784799	Badda, Dhaka	23000	785
4003	4	Fred	9943365121	1965234799	1765448799	Gulistan, Dhaka	25000	785
4004	5	Edward	7843365671	1978114790	1763514756	Uttara, Dhaka	20000	785

5 rows returned in 0.00 seconds

Chef info Table:

1. insert into chef_info values(chef_info_chef_id.NEXTVAL, 'Amolnath', 5022559874, 01335119779, 01635119779,'Uttara, Dhaka', 15000, 4000);
2. insert into chef_info values(chef_info_chef_id.NEXTVAL, 'Uttam', 8866559869, 01735125779, 01635944779,'Badda, Dhaka', 17000, 4000);
3. insert into chef_info values(chef_info_chef_id.NEXTVAL, 'Raghunath', 6924359876, 01935459767, 01835549721,'Mirpur, Dhaka', 19000, 4002);
4. insert into chef_info values(chef_info_chef_id.NEXTVAL, 'Sapawn', 4522549884, 01535118979, 01690119529,'Mohakhali, Dhaka', 16000, 4002);
5. insert into chef_info values(chef_info_chef_id.NEXTVAL, 'Biplob', 3422549899, 01735419729, 01955719679,'Savar, Dhaka', 18500, 4004);



CHEF_ID	CHEF_NAME	CHEF_NID	CHEF_PHONE_NO_1	CHEF_PHONE_NO_2	ADDRESS	SALARY	MRG_ID
7000	Amolnath	5022559874	1335119779	1635119779	Uttara, Dhaka	15000	4000
7001	Uttam	8866559869	1735125779	1635944779	Badda, Dhaka	17000	4000
7002	Raghunath	6924359876	1935459767	1835549721	Mirpur, Dhaka	19000	4002
7003	Sapawn	4522549884	1535118979	1690119529	Mohakhali, Dhaka	16000	4002
7004	Biplob	3422549899	1735419729	1955719679	Savar, Dhaka	18500	4004

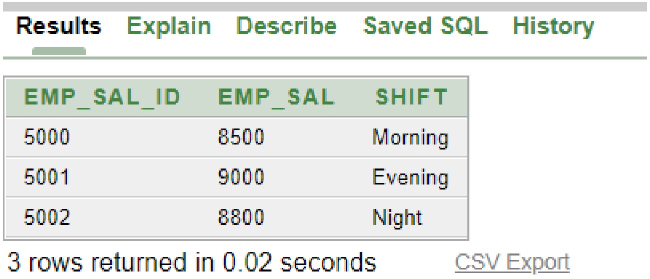
5 rows returned in 0.00 seconds

Employee sal info Table:

1. insert into emp_sal_info values(emp_sal_info_emp_sal_id.NEXTVAL, 8500, 'Morning');

2 insert into emp_sal_info values(emp_sal_info_emp_sal_id.NEXTVAL, 9000, 'Evening');

3. insert into emp_sal_info values(emp_sal_info_emp_sal_id.NEXTVAL, 8800, 'Night');

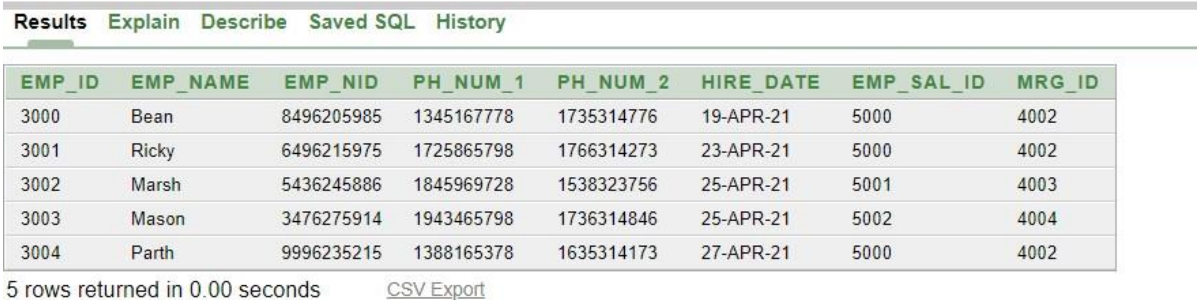


EMP_SAL_ID	EMP_SAL	SHIFT
5000	8500	Morning
5001	9000	Evening
5002	8800	Night

3 rows returned in 0.02 seconds

Employee info Table

1. insert into Employee_info values(Employee_info_emp_id.NEXTVAL, 'Bean', 8496205985, 01345167778, 01735314776,to_date('19-04-2021','dd-mm-yyyy'), 5000, 4002);
2. insert into Employee_info values(Employee_info_emp_id.NEXTVAL, 'Ricky', 6496215975, 01725865798, 01766314273,to_date('23-04-2021','dd-mm-yyyy'), 5000, 4002);
3. insert into Employee_info values(Employee_info_emp_id.NEXTVAL, 'Marsh', 5436245886, 01845969728, 01538323756,to_date('25-04-2021','dd-mm-yyyy'), 5001, 4003);
4. insert into Employee_info values(Employee_info_emp_id.NEXTVAL, 'Mason', 3476275914, 01943465798, 01736314846,to_date('25-04-2021','dd-mm-yyyy'), 5002, 4004);
5. insert into Employee_info values(Employee_info_emp_id.NEXTVAL, 'Parth', 9996235215, 01388165378, 01635314173,to_date('27-04-2021','dd-mm-yyyy'), 5000, 4002);

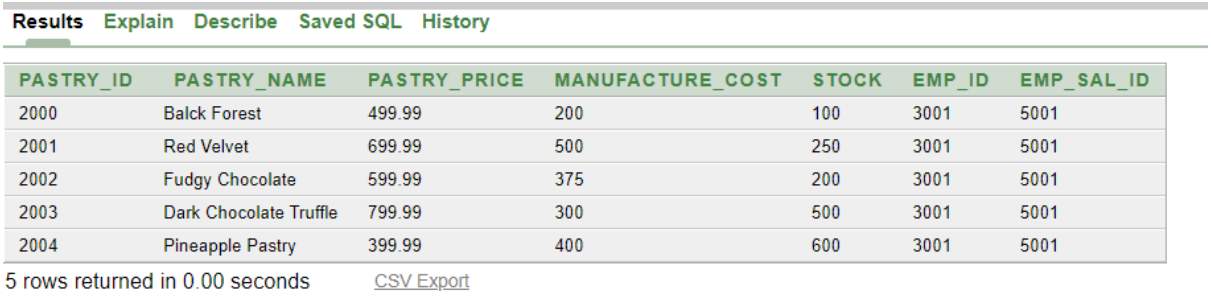


EMP_ID	EMP_NAME	EMP_NID	PH_NUM_1	PH_NUM_2	HIRE_DATE	EMP_SAL_ID	MRG_ID
3000	Bean	8496205985	1345167778	1735314776	19-APR-21	5000	4002
3001	Ricky	6496215975	1725865798	1766314273	23-APR-21	5000	4002
3002	Marsh	5436245886	1845969728	1538323756	25-APR-21	5001	4003
3003	Mason	3476275914	1943465798	1736314846	25-APR-21	5002	4004
3004	Parth	9996235215	1388165378	1635314173	27-APR-21	5000	4002

5 rows returned in 0.00 seconds

Pastry Info Table

1. insert into pastry_info values(pastry_info_pastry_id.nextval, 'Balck Forest', 499.99, 200, 100, 3001, 5001);
2. insert into pastry_info values(pastry_info_pastry_id.nextval, 'Red Velvet', 699.99, 500, 250, 3001, 5001);
4. insert into pastry_info values(pastry_info_pastry_id.nextval, ' Fudgy Chocolate', 599.99, 375, 200, 3001, 5001);
5. insert into pastry_info values(pastry_info_pastry_id.nextval, 'Dark Chocolate Truffle ', 799.99, 300,500, 3001, 5001);
6. insert into pastry_info values(pastry_info_pastry_id.nextval, 'Pineapple Pastry', 399.99, 400,600, 3001, 5001);

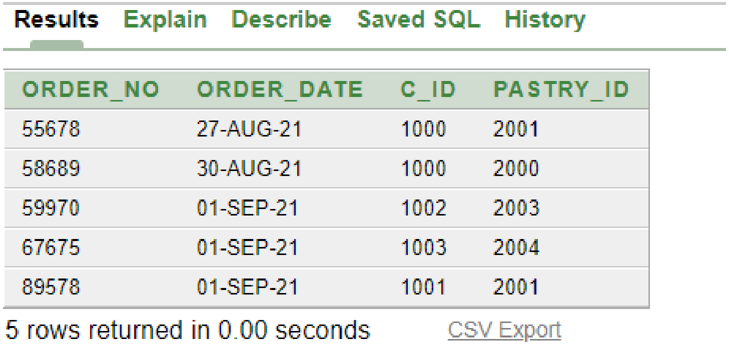


PASTRY_ID	PASTRY_NAME	PASTRY_PRICE	MANUFACTURE_COST	STOCK	EMP_ID	EMP_SAL_ID
2000	Balck Forest	499.99	200	100	3001	5001
2001	Red Velvet	699.99	500	250	3001	5001
2002	Fudgy Chocolate	599.99	375	200	3001	5001
2003	Dark Chocolate Truffle	799.99	300	500	3001	5001
2004	Pineapple Pastry	399.99	400	600	3001	5001

5 rows returned in 0.00 seconds CSV Export

Offline Order Info Table

1. INSERT INTO Offline_order_info values(55678,to_date('27-08-2021','dd-mm-yyyy'), 1000, 2001);
2 INSERT INTO Offline_order_info values(58689,to_date('30-08-2021','dd-mm-yyyy'), 1000, 2000);
3. INSERT INTO Offline_order_info values(59970,to_date('1-09-2021','dd-mm-yyyy'), 1002, 2003);
4. INSERT INTO Offline_order_info values(67675,to_date('1-09-2021','dd-mm-yyyy'), 1003, 2004);
5. INSERT INTO Offline_order_info values(89578,to_date('1-09-2021','dd-mm-yyyy'), 1001, 2001);

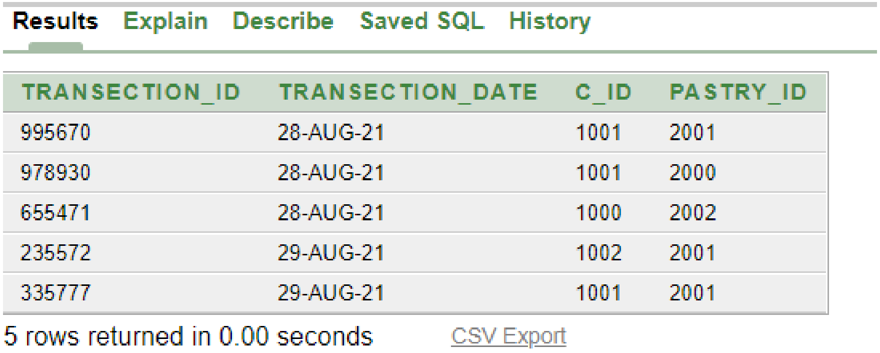


ORDER_NO	ORDER_DATE	C_ID	PASTRY_ID
55678	27-AUG-21	1000	2001
58689	30-AUG-21	1000	2000
59970	01-SEP-21	1002	2003
67675	01-SEP-21	1003	2004
89578	01-SEP-21	1001	2001

5 rows returned in 0.00 seconds CSV Export

Online Order Info Table

1. insert into Online_order_info values(995670,to_date('28-08-2021','dd-mm-yyyy'), 1001, 2001);
2. insert into Online_order_info values(978930,to_date('28-08-2021','dd-mm-yyyy'), 1001, 2000);
3. insert into Online_order_info values(655471,to_date('28-08-2021','dd-mm-yyyy'), 1000, 2002);
4. insert into Online_order_info values(235572,to_date('29-08-2021','dd-mm-yyyy'), 1002, 2001);
5. insert into Online_order_info values(335777,to_date('29-08-2021','dd-mm-yyyy'), 1001, 2001);

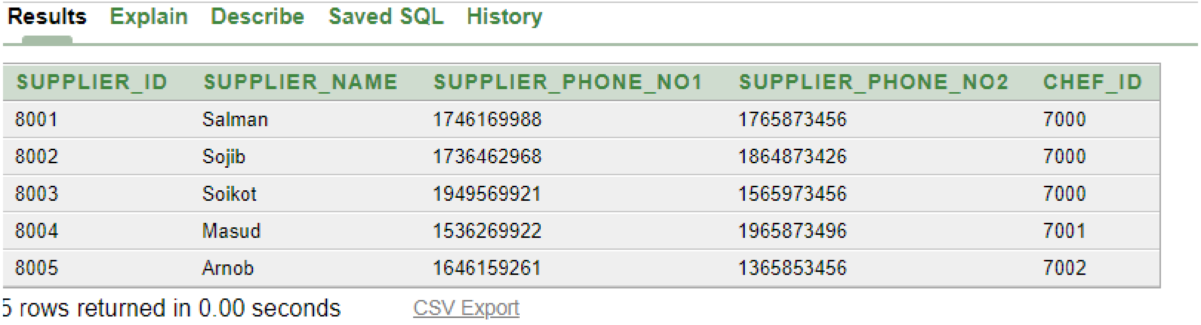


TRANSECTION_ID	TRANSECTION_DATE	C_ID	PASTRY_ID
995670	28-AUG-21	1001	2001
978930	28-AUG-21	1001	2000
655471	28-AUG-21	1000	2002
235572	29-AUG-21	1002	2001
335777	29-AUG-21	1001	2001

5 rows returned in 0.00 seconds

Supplier Info Table

1. Insert into Supplier_info values (supplier_info_supplier_id.NEXTVAL, 'Salman', 01746169988, 01765873456, 7000);
2. insert into Supplier_info values (supplier_info_supplier_id.NEXTVAL, 'Sojib', 01736462968, 01864873426, 7000);
3. insert into Supplier_info values (supplier_info_supplier_id.NEXTVAL, 'Soikot', 01949569921, 01565973456, 7000);
4. insert into Supplier_info values (supplier_info_supplier_id.NEXTVAL, 'Masud', 01536269922, 01965873496, 7001);
5. insert into Supplier_info values (supplier_info_supplier_id.NEXTVAL, 'Arnob', 01646159261, 01365853456, 7002);

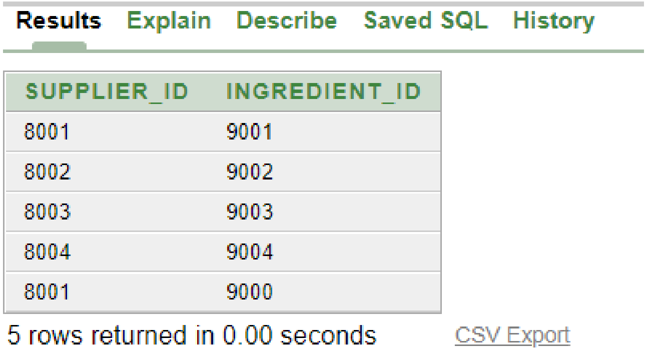


SUPPLIER_ID	SUPPLIER_NAME	SUPPLIER_PHONE_NO1	SUPPLIER_PHONE_NO2	CHEF_ID
8001	Salman	1746169988	1765873456	7000
8002	Sojib	1736462968	1864873426	7000
8003	Soikot	1949569921	1565973456	7000
8004	Masud	1536269922	1965873496	7001
8005	Arnob	1646159261	1365853456	7002

5 rows returned in 0.00 seconds

Common Table:

1. insert into common values(8001,9000);
2. insert into common values(8001,9001);
3. insert into common values(8002,9002);
4. insert into common values(8003,9003);
5. insert into common values(8004,9004);



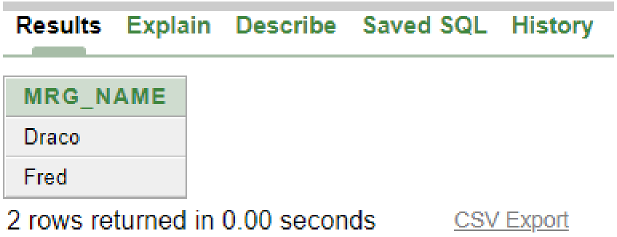
SUPPLIER_ID	INGREDIENT_ID
8001	9001
8002	9002
8003	9003
8004	9004
8001	9000

5 rows returned in 0.00 seconds

8. Query Writing

SUBQUERIES

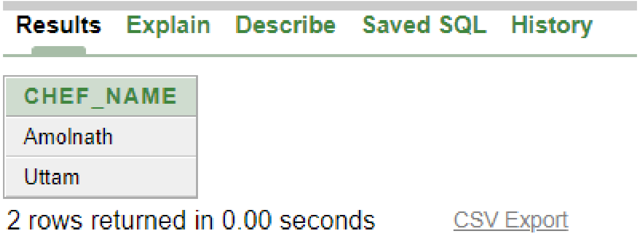
1. Display the manager names from the manager info table who earn more than manager Harmione.
   Solution: Select MRG_Name from Manager_info where (Salary>21000);



MRG_NAME
Draco
Fred

2 rows returned in 0.00 seconds

2. Display the chef names from the chef info table who have the same MRG_ID as chef Uttam.
   Solution: Select Chef_Name from Chef_info where MRG_ID = 4000;



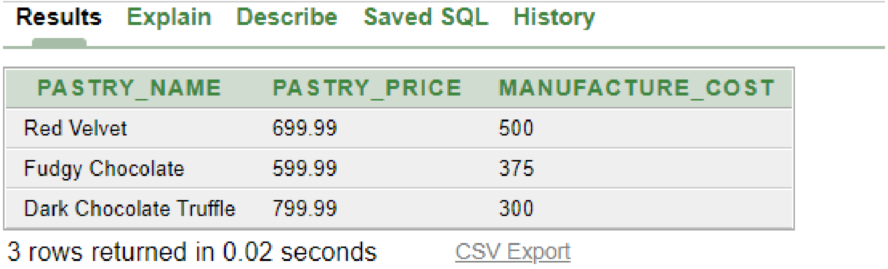
CHEF_NAME
Amolnath
Uttam

2 rows returned in 0.00 seconds

View

1. Create view called Pastry info view based on pastry name, Manfacture Cost and pastry price for those pastry prices is greater than 500.
   Solution:

```
CREATE VIEW pastry_info_view AS
SELECT        pastry_name, pastry_price, Manufacture_Cost
FROM           pastry_info
WHERE          pastry_price>500;
```

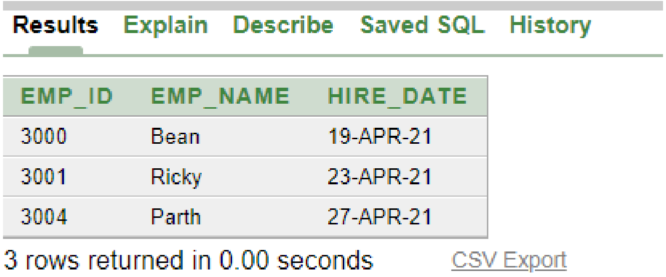


PASTRY_NAME	PASTRY_PRICE	MANUFACTURE_COST
Red Velvet	699.99	500
Fudgy Chocolate	599.99	375
Dark Chocolate Truffle	799.99	300

3 rows returned in 0.02 seconds   CSV Export

2 Create view called employee_info_view based on employee ID, employee name and hire date for those who are manger of 4002.
Solution:

```
CREATE VIEW employee_info_view AS
SELECT        emp_id, emp_name, hire_date
FROM          employee_info
WHERE         mrg_id = 4002;
```



EMP_ID	EMP_NAME	HIRE_DATE
3000	Bean	19-APR-21
3001	Ricky	23-APR-21
3004	Parth	27-APR-21

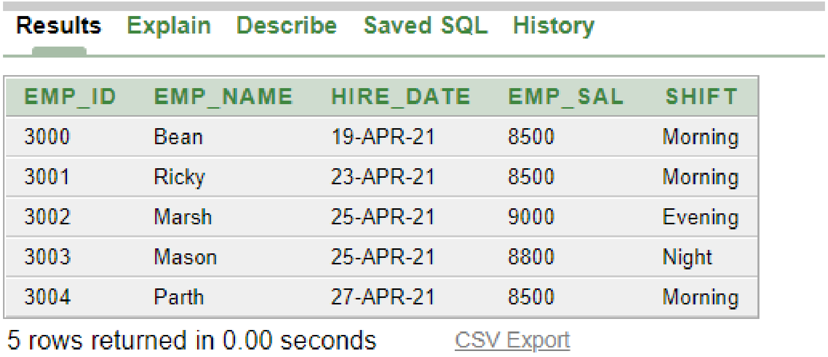
3 rows returned in 0.00 seconds   CSV Export

Joining

1. Display the employee's salary and shift.

Solution:

```
select employee_info.emp_id, employee_info.emp_name, employee_info.hire_date,
            emp_sal_info.emp_sal,emp_sal_info.shift
from        employee_info, emp_sal_info
where       employee_info.emp_sal_id=emp_sal_info.emp_sal_id;
```



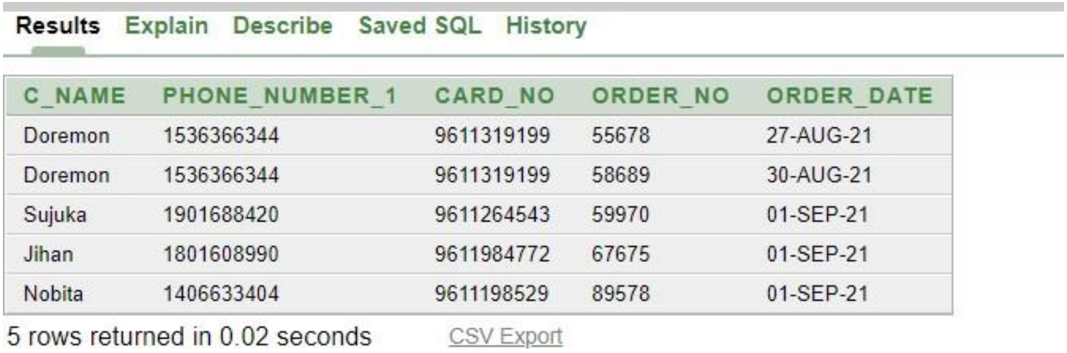
EMP_ID	EMP_NAME	HIRE_DATE	EMP_SAL	SHIFT
3000	Bean	19-APR-21	8500	Morning
3001	Ricky	23-APR-21	8500	Morning
3002	Marsh	25-APR-21	9000	Evening
3003	Mason	25-APR-21	8800	Night
3004	Parth	27-APR-21	8500	Morning

5 rows returned in 0.00 seconds

2. Display the offline order list with customer name and phone number.

Solution:

```
select customer_info.c_name, customer_info.phone_number_1, customer_info.card_no,
            offline_order_info.order_no, offline_order_info.order_date
from     customer_info, offline_order_info
where  customer_info.c_id=offline_order_info.c_id;
```



C_NAME	PHONE_NUMBER_1	CARD_NO	ORDER_NO	ORDER_DATE
Doremon	1536366344	9611319199	55678	27-AUG-21
Doremon	1536366344	9611319199	58689	30-AUG-21
Sujuka	1901688420	9611264543	59970	01-SEP-21
Jihan	1801608990	9611984772	67675	01-SEP-21
Nobita	1406633404	9611198529	89578	01-SEP-21

5 rows returned in 0.02 seconds

9. Relational Algebra

1. Find the name of the employee where emp id is 3002.

$$\Pi_{emp\_name} (\sigma_{emp\_id\ =3002} (employee\_info))$$

2. Find the pastry price where pastry name is Red Velvet.

$$\Pi_{pastry\_price} (\sigma_{pastry\_name\ = \text{“Red valvet”}} (pastry\_info))$$

3. Find the manager’s name who’s salary is less than 23000.

$$\Pi_{mrg\_name} (\sigma_{salary\ <23000}(manager\_info))$$

4. Find the name of owner where owner id is 6001.

$$\Pi_{owner\_name} (\sigma_{owner\_id\ =\ 6001}(owner\_info))$$

5. Find the phone number of owner where owner id is 6001.

$$\Pi_{owner\_phone\_1,\ owner\_phone\_2} (\sigma_{owner\_id\ =\ 6001}(owner\_info))$$

10. Conclusion

We have completed the project titled Pastry Shop Management System successfully.

The purpose of this project was to develop a relational database management system to manage all aspects of a Pastry Shop company from employee management to inventory management to sales management. We have tried to implement all topics taughtto us in the course. We have taken care to develop the system free of errors and make it user friendly.

Although we are happy with how much we have done in our project, there is scope for also development. Various features can be implemented for the customers such as the customer being able to see history of all their previous purchases. Some features that can be implemented which would help the business are daily stock update, record of sale of a specific Pastry over a month to determine which Pastry is more in demand.