

Ice Cream Shop

Management System

a sql project

FALL 2022

Introduction To Database [SECTION- H]

DATE OF SUBMISSION: 12 Dec, 2022

**ProJect Title :** AN ice cream shop management system

|  |  |
| --- | --- |
| Name | Id |
| tahsin, md tasmim al | 22-46299-1 |
| hossain, mir md mofakkar | 22-46245-1 |
| Tanzil, Md rayhan | 22-46300-1 |

**Submitted by**

**GROUP: 2**

**instructed by**

SIFAT RAHMAN AHONA

###### Assistant Professor, Computer Science

AMERICAN INTERNATIONAL UNIVERSITY – BANGLADESH

TABLE OF CONTENTS

|  |  |
| --- | --- |
| * CASE STUDY | 3 |
| * ER DIAGRAM | 4 |
| * NORMALIZATION | 5 |
| * TABLE CREATION | 12 |
| * DATA INSERTION | 21 |
| * CONSTRAINTS ADDING | 33 |
| * QUERY WRITING | 35 |

CASE STUDY

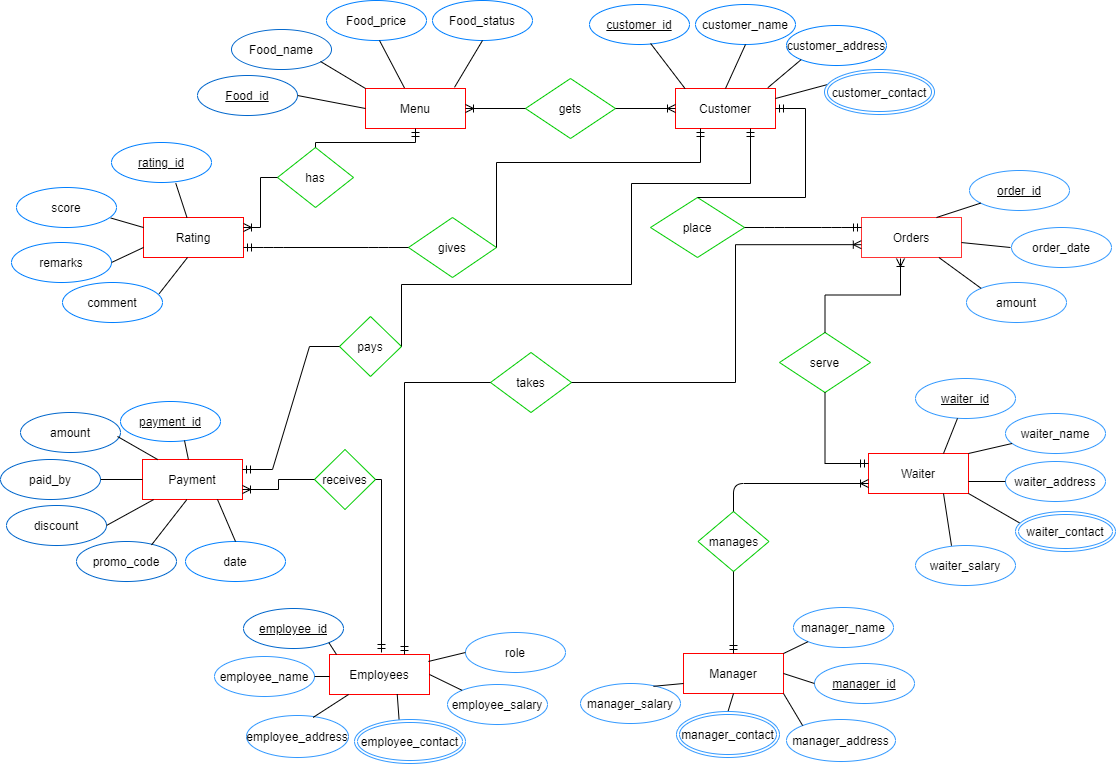
# Introduction:

A method for systematically managing an ice-cream shops daily commodities is called an ice-cream shop management system. This system's primary benefit is that it makes managing the business a lot easier. Reduced manual labor and increased workplace productivity can execute through this management system. The everyday flow of services and orders is aided by this method.

# Scenario Description:

Ice-cream shop management system offers an efficient working system for the shop. The shop has Menu in their system where food\_name, food\_id, Price, and the status of a food is stored. A customer can request menu details. One customer can submit a single order. A customer is identified as customer\_id, customer\_name, customer\_address, and contact\_number. However, customers can give ratings on the menu. Rating has unique rating\_id, score, remarks and comment. Orders details are associated with order\_id, order\_date, amount, total\_amount and order\_details. When a customer completes an order, then they process their payments like payment\_id, amount, paid\_by, discount, and date. An ice\_cream shop management system has many waiters to serve food. The attributes for the waiters are waiter\_id, waiter\_name, waiter\_address, phone\_number, salary, and role. Each order should produce one serve, and each serve is conducted by one waiter. Also, there are some employees to run the shops other commodities. Their attributes are employee\_id, employee\_name, employee\_address, phone\_number,salary and their role. The waiters and employees work under a manager and the managers name, address, phone\_number and salary is also stored in the system.

ER DIAGRAM



NORMALIZATION

Gets (Food\_id, Food\_name, Food\_status, Price, Customer\_id, Customer\_name, Customer\_address, Customer\_contact)

**1NF:** Customer\_contact Multivalued Attribute

**2NF:** Food\_id, Food\_name, Food\_status, Price

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

CF\_id, Customer\_id, Food\_id

**3NF:** There is no Transitive Dependency.

Food\_id, Food\_name, Food\_status, Price

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

CF\_id, Customer\_id, Food\_id

**TABLE:**

Food\_id, Food\_name, Food\_status, Price

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

CF\_id, Customer\_id, Food\_id

Place (Order\_id, Order\_date, Amount, Customer\_id, Customer\_name, Customer\_address, Customer\_contact)

**1NF:** Customer\_contact Multivalued Attribute

**2NF:** Order\_id, Order\_date, Amount

Customer\_id, Customer\_name, Customer\_address, Customer\_contact, Order\_id

**3NF:** There is no Transitive Dependency.

Order\_id, Order\_date, Amount, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact, Order\_id

**TABLE:**

Order\_id, Order\_date, Amount, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact, Order\_id

Serve (Order\_id, Order\_date, Amount, Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary)

**1NF:** Waiter\_contact Multivalued Attribute

**2NF:** Order\_id, Order\_date, Amount, Waiter\_id

Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary

**3NF:** There is no Transitive Dependency.

Order\_id, Order\_date, Amount, Waiter\_id

Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary

**TABLE:**

Order\_id, Order\_date, Amount, Waiter\_id

Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary

Manages (Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact, Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary)

**1NF:** Manager\_contact, Waiter\_contact Multivalued Attribute

**2NF:** Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary, Manager\_id

**3NF:** There is no Transitive Dependency.

Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact

Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary, Manager\_id

**TABLE:**

Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact

Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary, Manager\_id

Takes (Order\_id, Order\_date, Amount, Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role)

**1NF:** Employee\_contact Multivalued Attribute

**2NF:** Order\_id, Order\_date, Amount, Employee\_id

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

**3NF:** There is no Transitive Dependency.

Order\_id, Order\_date, Amount, Employee\_id

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

**TABLE:**

Order\_id, Order\_date, Amount, Employee\_id

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

Receives (Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role)

**1NF:** Employee\_contact Multivalued Attribute

**2NF:** Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Employee\_id

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

**3NF:** There is no Transitive Dependency.

Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Employee\_id

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

**TABLE:**

Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date

Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role

Pays (Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id, Customer\_name, Customer\_address, Customer\_contact)

**1NF:** Customer\_contact Multivalued Attribute

**2NF:** Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

**3NF**: There is no Transitive Dependency.

Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

**TABLE:**

Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

Gives (Rating\_id, Score, Remarks, Comment, Customer\_id, Customer\_name, Customer\_address, Customer\_contact)

**1NF**: Customer\_contact Multivalued Attribute

**2NF:** Rating\_id, Score, Remarks, Comment, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

**3NF:** There is no Transitive Dependency.

Rating\_id, Score, Remarks, Comment, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

**TABLE:**

Rating\_id, Score, Remarks, Comment, Customer\_id

Customer\_id, Customer\_name, Customer\_address, Customer\_contact

Has (Food\_id, Food\_name, Food\_status, Price, Rating\_id, Score, Remarks, Comment)

**1NF:** No Multivalued Attribute

**2NF:** Food\_id, Food\_name, Food\_status, Price

Rating\_id, Score, Remarks, Comment, Food\_id

**3NF:** There is no Transitive Dependency.

Food\_id, Food\_name, Food\_status, Price

Rating\_id, Score, Remarks, Comment, Food\_id

**TABLE:**

Food\_id, Food\_name, Food\_status, Price

Rating\_id, Score, Remarks, Comment, Food\_id

**TOTAL TABLE:**

1. Food\_id, Food\_name, Food\_status, Price
2. ~~Customer\_id, Customer\_name, Customer\_address, Customer\_contact~~
3. CF\_id, Customer\_id, Food\_id
4. ~~Order\_id, Order\_date, Amount~~
5. Customer\_id, Customer\_name, Customer\_address, Customer\_contact, Order\_id
6. Order\_id, Order\_date, Amount, Waiter\_id
7. ~~Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary~~
8. Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact
9. Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary, Manager\_id
10. Order\_id, Order\_date, Amount, Employee\_id
11. Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role
12. ~~Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date~~
13. ~~Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role~~
14. Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id
15. ~~Customer\_id, Customer\_name, Customer\_address, Customer\_contact~~
16. Rating\_id, Score, Remarks, Comment, Customer\_id
17. ~~Customer\_id, Customer\_name, Customer\_address, Customer\_contact~~
18. ~~Food\_id, Food\_name, Food\_status, Price~~
19. Rating\_id, Score, Remarks, Comment, Food\_id

**FINAL TABLE:**

1. Food\_id, Food\_name, Food\_status, Price
2. CF\_id, Customer\_id, Food\_id
3. Customer\_id, Customer\_name, Customer\_address, Customer\_contact, Order\_id
4. Order\_id, Order\_date, Amount, Waiter\_id
5. Manager\_id, Manager\_name, Manager\_address, Manager\_salary, Manager\_contact
6. Waiter\_id, Waiter\_name, Waiter\_address, Waiter\_contact, Waiter\_salary, Manager\_id
7. Order\_id, Order\_date, Amount, Employee\_id
8. Employee\_id, Employee\_name, Employee \_address, Employee\_contact, Employee\_salary, role
9. Payment\_id, Amount, Paid\_by, Discount, Promo\_code, Date, Customer\_id
10. Rating\_id, Score, Remarks, Comment, Customer\_id
11. Rating\_id, Score, Remarks, Comment, Food\_id

TABLE CREATION

MENU

create table Menu(Food\_id number(10) primary key, Food\_name varchar2(40), Food\_status varchar2(20), Price number(10))

create sequence menuseq

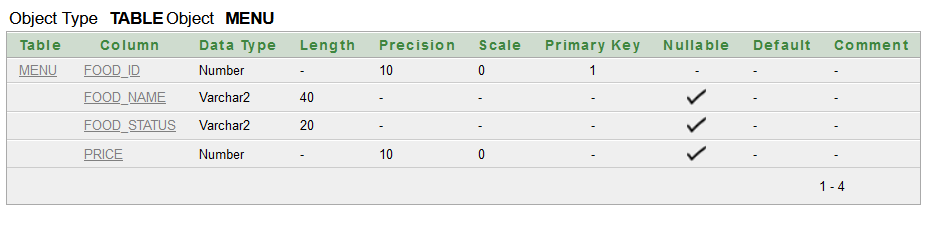
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



CUSTOMER

create table Customer(Customer\_id number(10) primary key, Customer\_name varchar2(40), Customer\_address varchar2(50), Customer\_contact number(20), Order\_id number(10))

create sequence customerseq

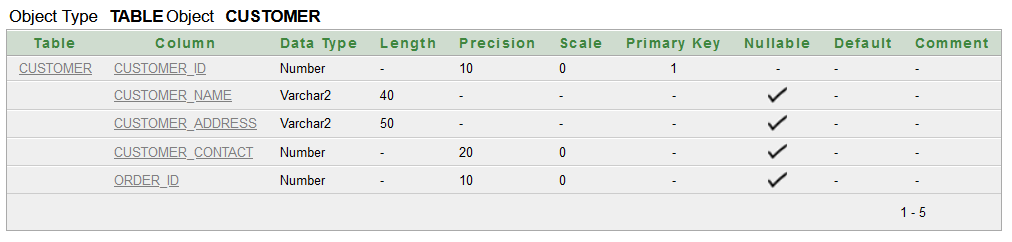
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



MANAGER

create table Manager(Manager\_id number(10) primary key, Manager\_name varchar2(30), Manager\_address varchar2(50), Manager\_salary number(10), Manager\_contact number(20))

create sequence managerseq

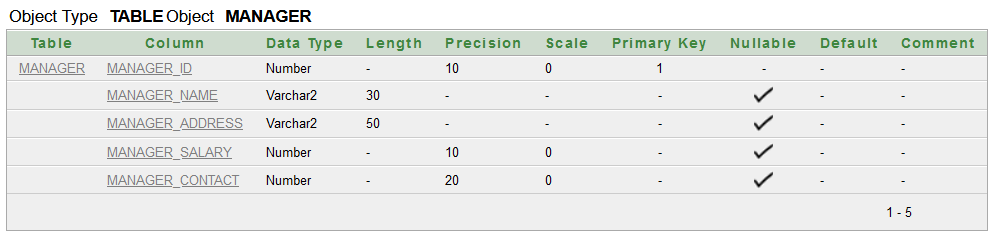
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



EMPLOYEE

create table Employee(Employee\_id number(10) primary key, Employee\_name varchar2(30), Employee\_address varchar2(50), Employee\_contact number(20), Employee\_salary number(10), role varchar2(20))

create sequence employeeseq

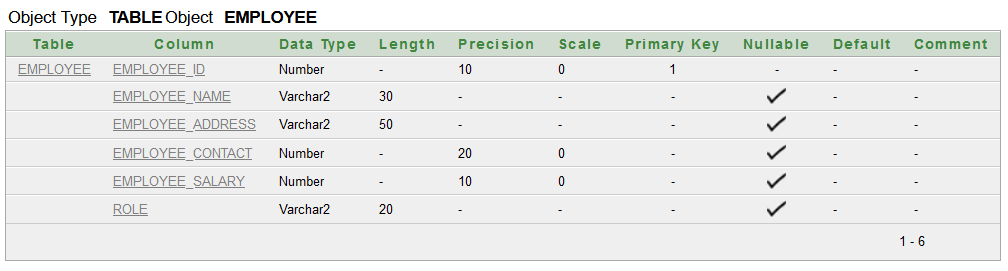
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



WAITER

create table Waiter (waiter\_id number(10) primary key, waiter\_name varchar2(30),waiter\_address varchar2(50), waiter\_contact number(20), waiter\_salary number(10), manager\_id number(10))

create sequence waiterseq

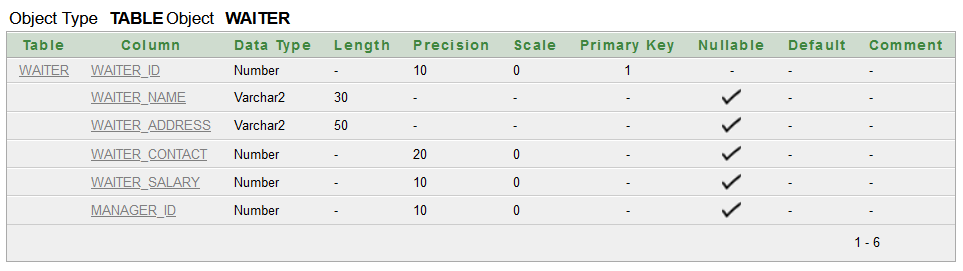
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



PAYMENT

create table Payment(payment\_id number(10) primary key, amount number(10), paid\_by varchar2(30), discounts number(10), promo\_code number(10), dates varchar2(10), customer\_id number(10))

create sequence paymentseq

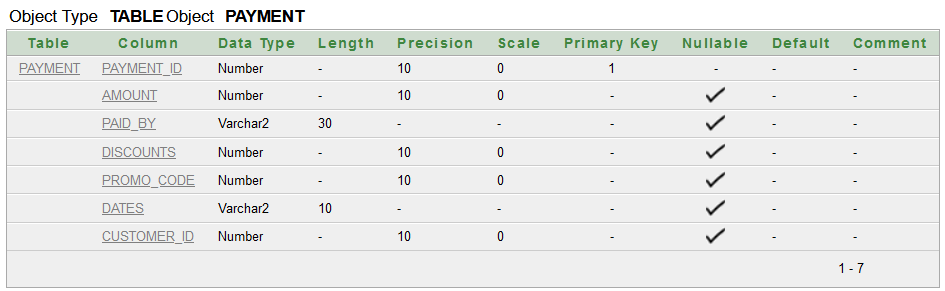
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



RATING

create table Rating(Rating\_id number(10) primary key, Scores number(10), Remarks varchar2(100), Comments varchar2(100), Customer\_id number(10), Food\_id number(10))

create sequence ratingseq

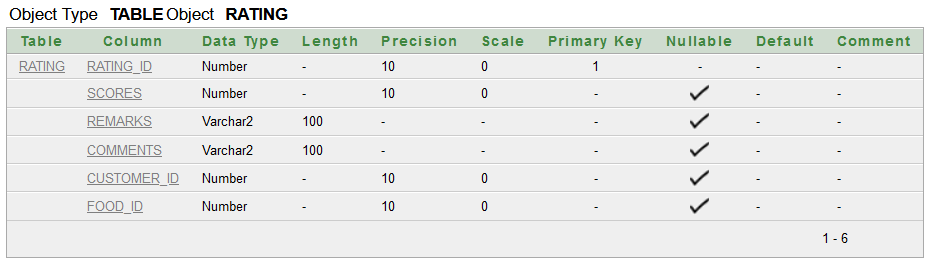
start with 1

increment by 1

maxvalue 500

NOCACHE

NOCYCLE



ORDERS

create table Orders(Order\_id number(10) primary key, Order\_date varchar2(20), Amount number(10), Waiter\_id number(10), Employee\_id number(10))

create sequence orderseq

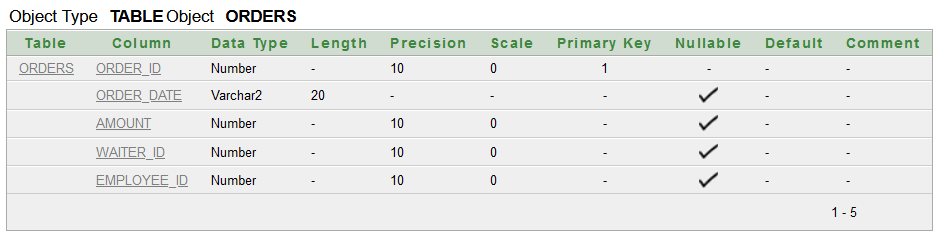
start with 1

increment by 1

maxvalue 500

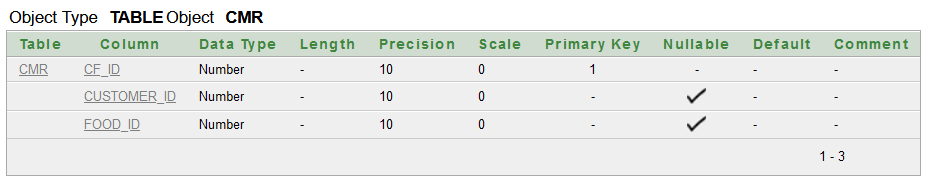
NOCACHE

NOCYCLE



CMR

create table CMR(CF\_id number(10) primary key, customer\_id number(10), food\_id number(10))



DATA INSERTION

MENU

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CHOCOLATE ICECREAM', 'AVAILABLE', 160);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'VANILLA ICECREAM', 'AVAILABLE', 120);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'PEANUT BUTTER ICECREAM', 'AVAILABLE', 199);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CINNAMON ICECREAM', 'AVAILABLE', 220);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'STRAWBERRY ICECREAM', 'AVAILABLE', 180);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CARAMEL ICECREAM', 'AVAILABLE', 220);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'BANANA ICECREAM', 'AVAILABLE', 120);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'MINT ICECREAM', 'AVAILABLE', 190)

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'BUTTER ICECREAM', 'AVAILABLE', 220);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CHERRY ICECREAM', 'AVAILABLE', 220);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'KIWI ICECREAM', 'AVAILABLE', 280);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'MELON MIX', 'AVAILABLE', 250);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'YOUGURT', 'AVAILABLE', 200);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CREAMY MILK', 'AVAILABLE', 200);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'MANGO ICECREAM', 'AVAILABLE', 190);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CHOCOLATE CONE', 'AVAILABLE', 199);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'STRAWBERRY CONE', 'AVAILABLE', 220);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'SUGER CONE', 'AVAILABLE', 180);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'WAFER CONE', 'UNAVAILABLE', 320);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'RASPBERRY ICECREAM', 'AVAILABLE', 380);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'HOT CHOCOLATE', 'AVAILABLE', 190);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'VANILLA MILKSHAKE', 'AVAILABLE', 120);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CHOCOLATE MILKSHAKE', 'AVAILABLE', 149);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'STRAWBERRY MILKSHAKE', 'AVAILABLE', 149);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'ORANGE SMOOTHIE', 'AVAILABLE', 99);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'KIWI SMOOTHIE', 'AVAILABLE', 120);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'CHOCOLATE SMOOTHIE', 'AVAILABLE', 99);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'STRAWBERRY SMOOTHIE', 'AVAILABLE', 99);

INSERT INTO Menu VALUES(menuseq.NEXTVAL, 'BIRTHDAY SURPRISE', 'AVAILABLE', 899);



CUSTOMER

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'RAYHAN', 'UTTARA, DHAKA', 8801884801110, 1);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'TAMIM', 'BANANI, DHAKA', 8801884871110, 2);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'NIHAB', 'BANANI, DHAKA', 8801884801189, 3);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'LIMA', 'KURIL, DHAKA', 8801884878110, 4);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'HIMEL', 'UTTARA, DHAKA', 8801884801120, 5);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'AKASH', 'UTTARA, DHAKA', 8801884801130, 6);

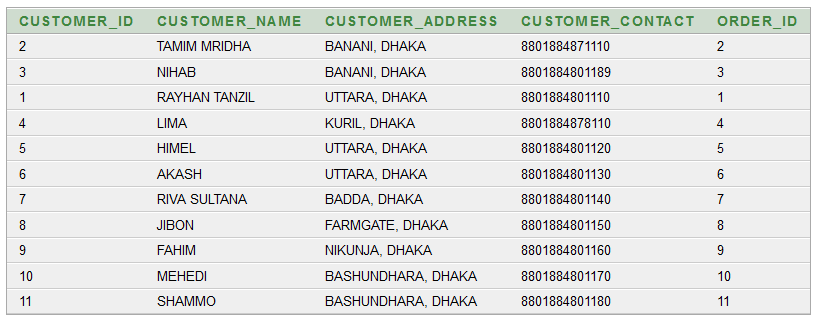
INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'RIVA', 'BADDA, DHAKA', 8801884801140, 7);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'JIBON', 'FARMGATE, DHAKA', 8801884801150, 8);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'FAHIM', 'NIKUNJA, DHAKA', 8801884801160, 9);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'MEHEDI', 'BASHUNDHARA, DHAKA', 8801884801170, 10);

INSERT INTO Customer VALUES(customerseq.NEXTVAL, 'SHAMMO', 'BASHUNDHARA, DHAKA', 8801884801180, 11);



MANAGER

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'MOFAKKAR', 'CHITTAGONG',25000, 8801478988574);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'RAYHAN', 'DHAKA',25000, 8801478988577);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'NIYAN', 'DHAKA',25000, 8801478988584);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'RABBI', 'BARISHAL',25000, 8801478988585);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'ROFIQ', 'DHAKA',25000, 8801478988587);

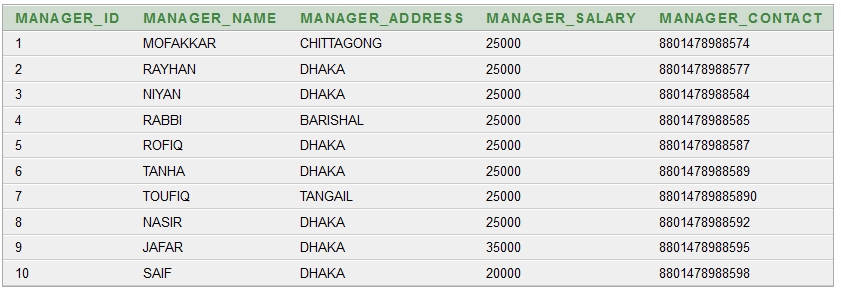
INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'TANHA', 'DHAKA',25000, 8801478988589);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'TOUFIQ', 'TANGAIL',25000, 88014789885890);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'NASIR', 'DHAKA',25000, 8801478988592);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'JAFAR', 'DHAKA',35000, 8801478988595);

INSERT INTO Manager VALUES(managerseq.NEXTVAL, 'SAIF', 'DHAKA',20000, 8801478988598);



EMPLOYEE

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'ALEX', 'DHAKA', 8801758998454, 10000, 'CASHIER');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'HALES', 'DHAKA', 8801758998455, 8000, 'CLEANER');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'TRUMP', 'CHITTAGONG', 8801758998456, 15000, 'GUARD');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'PUTIN', 'BARISHAL', 8801758998457, 15000, 'GUARD');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'MIKEL', 'DHAKA', 8801758998458, 8000, 'CLEANER');

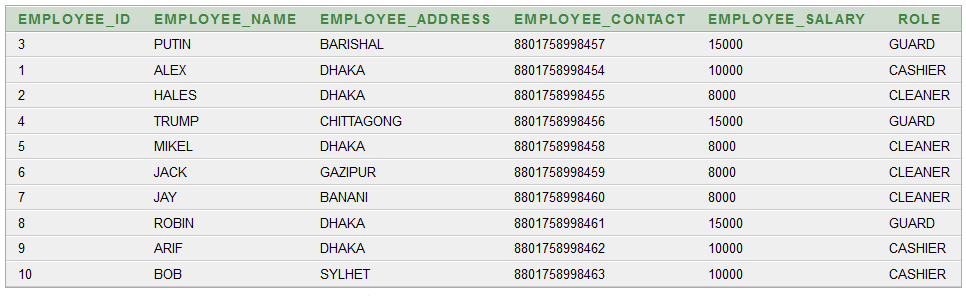
INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'JACK', 'GAZIPUR', 8801758998459, 8000, 'CLEANER');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'JAY', 'BANANI', 8801758998460, 8000, 'CLEANER');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'ROBIN', 'DHAKA', 8801758998461, 15000, 'GUARD');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'ARIF', 'DHAKA', 8801758998462, 10000, 'CASHIER');

INSERT INTO Employee VALUES(employeeseq.NEXTVAL, 'BOB', 'SYLHET', 8801758998463, 10000, 'CASHIER');



WAITER

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'NILOY', 'CHITTAGONG', 8801478987749,10000, 1);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'ALAMGIR', 'DINAJPUR', 8801478987750,10000, 2);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'SIAM', 'DHAKA', 8801478987789,10000, 3);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'JUNIOR', 'BARISHAL', 8801478987790,10000, 3);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'MD ALI', 'CHITTAGONG', 8801478987791,10000, 2);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'HIRA', 'RANGPUR', 8801478987747,10000, 2);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'YASHIR', 'COMILLA', 8801478987719,10000, 3);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'AFIF', 'CHITTAGONG', 8801478987730,10000, 4);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'SAIF', 'DINAJPUR', 8801478987750,10000, 5);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'SIFAT', 'DHAKA', 8801478987789,10000, 6);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'LAZIM', 'BARISHAL', 8801478987790,10000, 7);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'POROSH', 'CHITTAGONG', 8801478987791,10000, 8);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'FARUK', 'RANGPUR', 8801478987747,10000, 9);

INSERT INTO Waiter VALUES(waiterseq.NEXTVAL, 'HASIB', 'COMILLA', 8801478987719,10000, 10);



PAYMENT

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 520,'RAYHAN', '08DEC 2022', 1);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 120,'TAMIM', '08DEC 2022', 2);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 450,'NIHAB', '09DEC 2022', 3);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 550,'LIMA', '09DEC 2022', 4);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 600,'HIMEL', '10DEC 2022', 5);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 300,'AKASH', '10DEC 2022', 6);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 200,'RIVA', '10DEC 2022', 7);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 1000,'JIBON', '11DEC 2022', 8);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 900,'FAHIM', '11DEC 2022', 9);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 700,'MEHEDI', '11DEC 2022', 10);

INSERT INTO Payment (payment\_id ,amount,paid\_by,dates,customer\_id) VALUES(paymentseq.NEXTVAL, 300,'SHAMMO', '11DEC 2022', 11);



RATING

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Delicious food',1,8);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Delicious food',2,9);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Awesome experience',4,17);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Good',9,10);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Best in town',5,11);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Icy',3,20);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'what a surprise!',10,20);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Great food',6,7);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Best flavour',7,16);

INSERT INTO Rating (Rating\_id ,Scores,Remarks,Customer\_id,Food\_id) VALUES(ratingseq.NEXTVAL, 9,'Nice behaviour',8,12);



ORDERS

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '08-DEC-2022', 520, 6,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 120, 5,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 450, 2,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 220, 2,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 320, 6,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 260, 5,1);

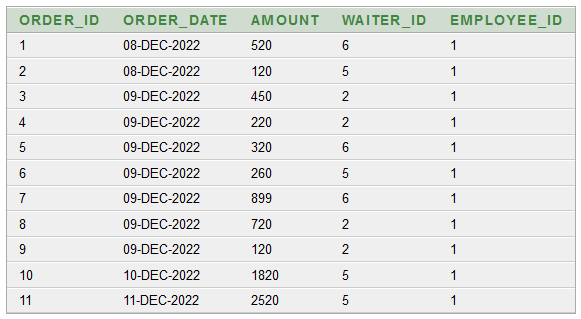
INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 899, 6,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 720, 2,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '09-DEC-2022', 120, 2,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '10-DEC-2022', 1820, 5,1);

INSERT INTO Orders VALUES(orderseq.NEXTVAL, '11-DEC-2022', 2520, 5,1);



CMR

INSERT INTO CMR VALUES(1,1,8);

INSERT INTO CMR VALUES(2,2,9);

INSERT INTO CMR VALUES(3,4,17);

INSERT INTO CMR VALUES(4,9,10);

INSERT INTO CMR VALUES(5,5,11);

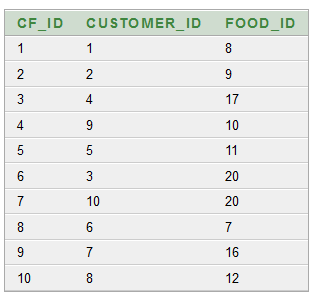
INSERT INTO CMR VALUES(6,3,20);

INSERT INTO CMR VALUES(7,10,20);

INSERT INTO CMR VALUES(8,6,7);

INSERT INTO CMR VALUES(9,7,16);

INSERT INTO CMR VALUES(10,8,12);



CONSTRAINTS ADDING

MENU

ALTER TABLE Menu MODIFY Price NOT NULL;

CUSTOMER

ALTER TABLE Customer MODIFY Order\_id NOT NULL;

alter table Customer add constraint ofk foreign key (Order\_id) references orders(Order\_id);

MANAGER

ALTER TABLE Manager MODIFY Manager\_salary NOT NULL;

WAITER

alter table Waiter add constraint mfk foreign key (Manager\_id) references Manager(Manager\_id);

PAYMENT

alter table Payment add constraint cfk foreign key (Customer\_id) references Customer(Customer\_id);

RATING

alter table Rating add constraint rfk foreign key (Customer\_id) references Customer(Customer\_id);

alter table Rating add constraint rmfk foreign key (Food\_id) references Menu(Food\_id);

ORDERS

alter table Orders add constraint wfk foreign key (Waiter\_id) references Waiter(Waiter\_id);

alter table Orders add constraint efk foreign key (Employee\_id) references Employee(Employee\_id);

CMR

alter table CMR add constraint cufk foreign key (Customer\_id) references Customer(Customer\_id);

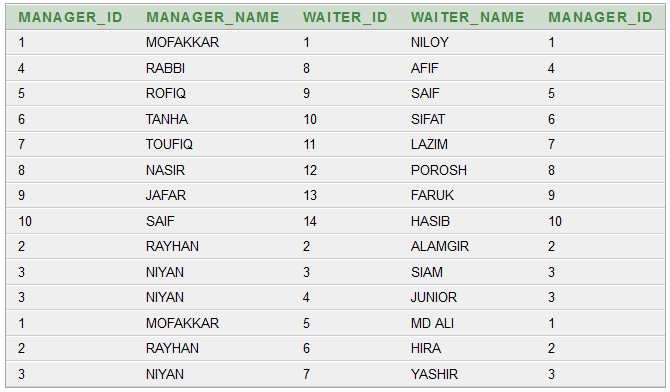
alter table CMR add constraint fofk foreign key (Food\_id) references Menu(Food\_id);

QUERY WRITING

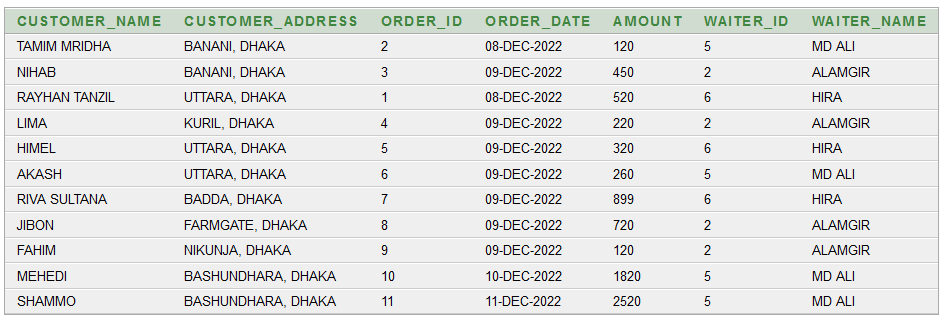
JOINING

EQUIJOIN

1. select m.manager\_id, m.manager\_name, w.waiter\_id, w.waiter\_name, w.manager\_id from manager m, waiter w where m.manager\_id = w.manager\_id

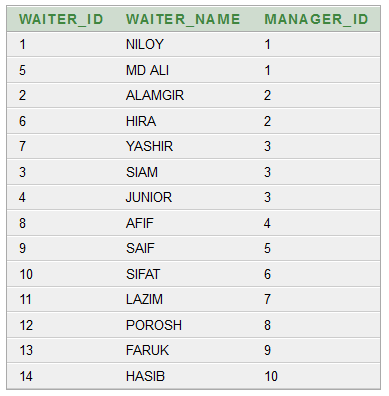


1. select c.customer\_name, c.customer\_address, c.order\_id, o.order\_date, o.amount, o.waiter\_id, w.waiter\_name from customer c, orders o, waiter w where c.order\_id = o.order\_id and o.waiter\_id = w.waiter\_id

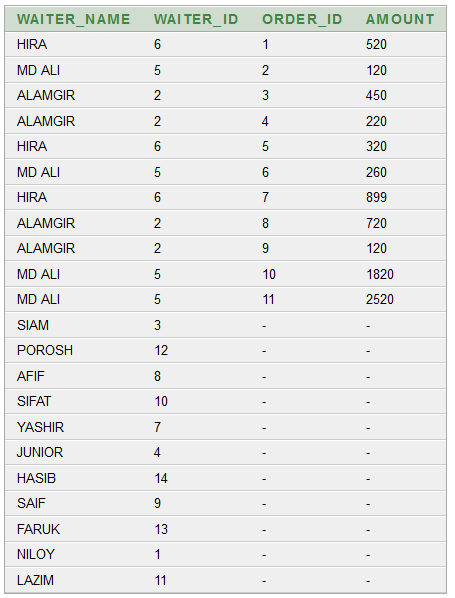


OUTERJOIN

1. select w.waiter\_id, w.waiter\_name, m.manager\_id from waiter w, manager m where w.manager\_id(+) = m.manager\_id order by w.manager\_id

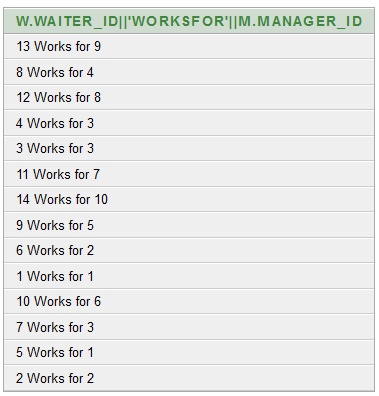


1. select w.waiter\_name, w.waiter\_id, o.order\_id, o.amount from orders o, waiter w where w.waiter\_id = o.waiter\_id (+)

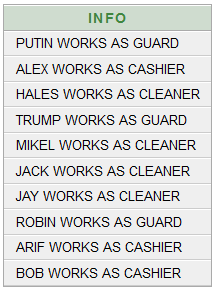


SELFJOIN

1. select distinct w.waiter\_id || ' Works for ' || m.manager\_id from waiter w, waiter m where w.manager\_id = m.manager\_id



1. select e.employee\_name ||' WORKS AS '|| r.role as info from employee e, employee r where e.employee\_id = r.employee\_id

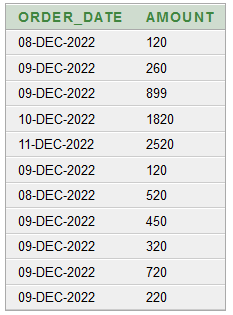


NON\_EQUIJOIN

1. select distinct e.employee\_name from waiter w, employee e where w.waiter\_salary>=10000



1. select distinct o.order\_date, o.amount from orders o, waiter w where w.waiter\_id between 5 and 7



SUBQUERY

1. select \* from menu where food\_id in (select food\_id from menu where price>200)
2. select \* from customer where customer\_id = any (select customer\_id from customer where customer\_address = 'UTTARA, DHAKA')
3. select \* from employee where employee\_salary > all (select employee\_salary from employee where role='CLEANER')
4. select \* from waiter where waiter\_address <> all (select waiter\_address from waiter where manager\_id = 2)
5. select \* from orders where order\_id = any (select order\_id from orders where amount>300)

SINGLE ROW FUNCTION

1. select \* from employee where LOWER(role) = 'guard'
2. select \* from rating where UPPER(remarks) = 'GOOD'
3. select \* from menu where INITCAP(food\_name) = 'Mint Icecream'
4. select employee\_name, CONCAT(employee\_name,role), LENGTH(role) from employee where employee\_address = 'DHAKA'
5. select TO\_CHAR(employee\_salary, '$99,999') salary from employee where employee\_name='ALEX'

GROUP FUNCTION

1. select MIN(employee\_salary) from employee where employee\_address = 'DHAKA'
2. select MAX(employee\_salary) from employee where employee\_address = 'DHAKA'
3. select COUNT(\*) from waiter where manager\_id = 2
4. select AVG(employee\_salary) from employee group by role
5. select role, SUM(employee\_salary) from employee where employee\_name<>'JACK' group by role having SUM(employee\_salary) >5000 order by SUM(employee\_salary) desc

VIEW

create or replace view menuview

as

select m.food\_name, m.food\_status, m.price, r.remarks, r.food\_id from menu m, rating r

where m.food\_id = r.food\_id

select \* from menuview



SEQUENCE

create sequence menuseq

start with 1

increment by

maxvalue 500

NOCACHE

NOCYCLE

GENERAL QUERY

1. select employee\_name from employee where employee\_name like '%L%' ;
2. select customer\_name, order\_id from customer where customer\_id in (5,9,10);
3. select food\_name, food\_id, price from menu where food\_status = 'UNAVAILABLE' ;
4. select employee\_id, employee\_name, employee\_salary from employee where employee\_id<>90 ;
5. select manager\_id, manager\_name ||' FROM '|| manager\_address as info from manager;

THE END