Name: Kanchan Thakare

**SQL Server Assignments**

**Design the above database with following table by applying Primary key and Foreign key.**

**In Customer table FirstName Attribute should not accept null value**

CREATE TABLE Customer(

Id int PRIMARY KEY,

FirstName nvarchar(40) NOT NULL,

LastName nvarchar(40),

City nvarchar(40),

Country nvarchar(40),

Phone nvarchar(20),

Fax\_number int

);

In Order table OrderDate should not accept null value

CREATE TABLE Order(

Id int IDENTITY(1,1) PRIMARY KEY,

OrderDate DATETIME NOT NULL,

OrderNumber nvarchar(10),

CustomerId int FOREIGN KEY references Customer(Id),

TotalAmount decimal(12,2)

);

CREATE TABLE OrderItem(

Id int PRIMARY KEY,

OrderId int FOREIGN KEY references Order(Id),

ProductId int FOREIGN KEY references Product(Id),

UnitPrice decimal(12,2),

Quantity int

);

CREATE TABLE Product(

Id int PRIMARY KEY,

ProductName nvarchar(50),

UnitPrice decimal(12,2),

Package nvarchar(30),

IsDiscontinued bit

);

**Insert Records in all tables**

INSERT INTO Customer (Id, FirstName, LastName, City, Country, Phone, Fax\_number)

VALUES (‘1’, ‘Steven', 'King’, ‘Washington DC’, ‘USA’, ‘515.123.4567’,101);

INSERT INTO Customer (Id, FirstName, LastName, City, Country, Phone, Fax\_number)

VALUES (‘2’, ‘Neena’, ‘Kochhar’, ‘London’, ‘United Kingdom’, ‘9678543218’,102);

INSERT INTO Customer (Id, FirstName, LastName, City, Country, Phone, Fax\_number)

VALUES (‘3’, ‘Bruce’, ‘Ernst’, ‘Berlin’, ‘Germany’, ‘030-0074321’);

INSERT INTO Order (OrderDate, OrderNumber, CustomerId, TotalAmount)

VALUES (‘2020-03-22 03:12:12’, ‘01’,1,695.0);

INSERT INTO Order (OrderDate, OrderNumber, CustomerId, TotalAmount)

VALUES ('2020-05-24 05:12:12','O2',2,345.0);

INSERT INTO Order (OrderDate, OrderNumber, CustomerId, TotalAmount)

VALUES ('2020-07-18 08:18:12','O3',3,896.0);

insert into Order(OrderDate,OrderNumber,CustomerId,TotalAmount)

values ('2018-08-12 07:18:11','O5',5,580.00);

INSERT INTO OrderItem (Id, OrderId, ProductId, UnitPrice, Quantity)

VALUES (1,1,1,22,3);

INSERT INTO OrderItem (Id, OrderId, ProductId, UnitPrice, Quantity)

VALUES (2,2,2,38,5);

INSERT INTO OrderItem (Id, OrderId, ProductId, UnitPrice, Quantity)

VALUES (3,3,3,42,6);

INSERT INTO OrderItem (Id, OrderId, ProductId, UnitPrice, Quantity)

VALUES (4,4,4,60,1);

INSERT INTO Product (Id, ProductName ,UnitPrice, Package, IsDiscontinued)

VALUES (1,'Chai',18.00,'Package1',1);

INSERT INTO Product (Id, ProductName, UnitPrice, Package, IsDiscontinued)

VALUES (2,'Cheese',19.00,'Package2',1);

INSERT INTO Product (Id, ProductName, UnitPrice,Package, IsDiscontinued)

VALUES (3,'Aniseed Syrup',10.00,'Package3',0);

INSERT INTO Product (Id, ProductName, UnitPrice, Package, IsDiscontinued)

VALUES (4,'Tofu',23.25,'Package4',0);

**Display all customer details**

SELECT \* FROM Customer;

**Write a query to display Country whose name starts with A or I**

SELECT \* FROM Customer

WHERE Country LIKE ‘A%’ OR Country LIKE ‘I%’;

**write a query to display whose name of customer whose third character is i**

SELECT \* FROM Customer

WHERE FirstName LIKE ‘\_\_i%’;

**Display the details from Customer table who is from country Germany**

SELECT \* FROM Customer

WHERE Country= ‘Germany’

**Display the customer details who has Fax number**

SELECT \* FROM Customer

WHERE Fax\_number is NOT NULL;

**display the customer details whose name holds second letter as U**

SELECT \* from Customer

WHERE FirstName like ‘\_U%’;

**select order Details where unit price is greater than 10 and less than 20**

SELECT \* FROM OrderItem

WHERE UnitPrice>10 AND UnitPrice<20;

**Display order details which contains shipping date and arrange the order by date**

SELECT \* FROM Order

ORDER BY OrderDate;

**Display the full name of the employee**

CREATE TABLE employee(

emp\_id int PRIMARY KEY,

FirstName nvarchar(40) NOT NULL,

LastName nvarchar(40),

Department\_Name nvarchar(40),

Manager\_name nvarchar(30)

);

INSERT INTO employee (emp\_id, Firstname, LastName, Department\_Name, Manager\_name)

VALUES (4, 'Steven', 'King','Administration', ‘John’);

INSERT INTO employee (emp\_id, Firstname, LastName, Department\_Name, Manager\_name)

VALUES (4, 'Neena', 'Kochhar', ‘Marketing’, ‘Alex’);

INSERT INTO employee (emp\_id, Firstname, LastName, Department\_Name, Manager\_name)

VALUES (4, 'Bruce', 'Ernst', ‘IT’, ‘Naina’);

SELECT FirstName, LastName FROM employee;

**Print the orders shipped by ship name 'La corned'abondance' between 2 dates(Choose dates of your choice)**

SELECT \* FROM Orders

WHERE Ship\_name='La corned'abondance' AND Shipping\_date BETWEEN ‘2000-12-12’ AND ‘2002-12-12’;

**Print the products supplied by 'Exotic Liquids'**

SELECT \* FROM Products where ShipName = ‘Exotic Liquids’

**Print all the Shipping company name and the ship names if they are operational**

SELECT Company\_name ,Ship\_name from Shipping\_details where operational=1;

**Print all Employees with Manager Name**

SELECT \* FROM Employee where Manager\_name is not null;

**Print the bill for a given order id ,bill should contain Productname, Categoryname,price after discount**

SELECT order\_id, Productname, Categoryname, price -price\*discount\_persent AS ‘price’ FROM order;

**Print the Total price of orders which have the products supplied by 'Exotic Liquids' if the price is > 50 and also print it by Shipping company's Name**

SELECT Company\_name, UnitPrice, Quantity, UnitPrice\*Quantity AS Total\_price from Shipping\_details

WHERE Supplier\_name= ‘Exotic Liquids’;

**write a query to display the orders placed by customer with phone number 030-0074321**

SELECT \* FROM Order

INNER JOIN Customer on Order.CustomerId = Customer.Id

WHERE Phone = '030-0074321';

**fetching all the products which are available under Category ‘Seafood’.**

SELECT \* FROM OrderDetails WHERE Category='Seafood';

**Display the orders placed by customers not in London**

SELECT \* FROM Order

INNER JOIN Customer on Order.CustomerId = Customer.Id

WHERE City != 'London';

**selects all the order which are placed for the product Chai.**

SELECT \* FROM Product

WHERE ProductName='Chai';

**Write a query to display the name , department name and rating  of any given employee**

ALTER TABLE employee

ADD Rating int DEFAULT 0;

ALTER TABLE employee

ADD constraint Check\_rating

CHECK (Rating>0 and Rating<5);

UPDATE employee

SET Rating=1 WHERE emp\_id = 2;

SELECT \* FROM employee;

**Print the Total price of orders which have the products supplied by 'Exotic Liquids' if the price is > 50 and also print it by Shipping company's Name**

SELECT Supplier\_name, Quantity, Price, (Price \* Quantity) AS Total FROM OrderDetails

WHERE Supplier\_name ='Exotic Liquids';

SELECT SUM(Price) AS Total, Categoryname FROM product\_supply

GROUP BY Supplier\_name

HAVING Supplier\_name = ‘Exotic Liquids’ and Price >50;

**Display the employee details whose joined at first**

ALTER TABLE employee

ADD Joining\_date date;

UPDATE employee

SET Joining\_date= ‘2022-06-6’ WHERE emp\_id = 3;

SELECT top 1 \* FROM employee

ORDER BY JoiningDate ASC;

**Display the employee details whose joined at recently**

SELECT top 1 \* FROM employee

ORDER BY JoiningDate DESC;

**Write a query to get most expense and least expensive Product list (name and unit price)**

SELECT \* FROM Product

WHERE UnitPrice IN (select max(UnitPrice) FROM Product) OR UnitPrice IN (select min(UnitPrice) FROM Product)

**Display the list of products that are out of stock**

SELECT \* FROM Product

WHERE Availability\_status= ‘Out of Stock’;

**Display the list of products whose unitinstock is less than unitonorder**

alter table Product add unitinstock int, unitonorder int;

update Product set unitinstock=25, unitonorder=50;

SELECT \* FROM Product

WHERE unitinstock<unitonorder;

**Display list of categories and suppliers who supply products within those categories**

ALTER TABLE Product ADD category nvarchar(40), supplier nvarchar(40);

UPDATE Product SET category='Beverages' , supplier ='Sup1' WHERE Id=101;

UPDATE Product SET category='Dairy' , supplier ='Sup2' WHERE Id=102;

UPDATE Product SET category='Syrups' , supplier ='Sup3' WHERE Id=103;

UPDATE Product SET category='Soy based' , supplier ='Sup4' WHERE Id=104;

SELECT category , supplier FROM Product ORDER BY category;

**Write query that determines the customer who has placed the maximum number of orders**

SELECT \* FROM Customer

WHERE Id IN (SELECT CustomerId FROM Order GROUP BY CustomerId

HAVING COUNT(Id)= (SELECT top 1 COUNT(Id) FROM Order GROUP BY CustomerId ORDER BY 1 DESC));

**Display the CustomerId whose name has substring ‘RA’**

UPDATE Customer SET FirstName= ‘ARADHYA’ WHERE Id=2;

SELECT Id FROM Customer WHERE FirstName LIKE ‘%RA%’;