Name: Rahul Vijayvargiya

Student ID: 245784

Prof. Marek Kopel

Subject: Database Design

Topic: Food Truck Business Database

Database Design Project Documentation

Introduction and Objective

1. The Food Truck industry has a unique requirement when managing the data it generates from its business.

2. There is no recognized system in place to support the food truck business.

3. The purpose of the database is to maintain data of FoodTrucks across the city and leverage the orders data to improve profits.

4. The problems we have addressed in this project are as follows:

5. Maintain all relevant information on Food Trucks such as Order, Customer, Staff, Supplies, Income , Expenditure and Revenue.

6. Analyse all location and cuisine related data from the orders of each Foodtruck.

7. Analyze Income, Expenditure and Revenue of all Food Trucks, such that Food Truck owners get clear understanding of their business output.

Database Purpose

1. The purpose of the database is to maintain data of Food Trucks across the city and leverage the orders data to improve profits, thus benefiting business growth.

2. The purpose of a food truck database project is to create a system for storing and organising information about food trucks, their menus, locations, and schedules. The database can be used to track inventory, manage orders, and analyse sales data.

Problem Domain

The problem domain for a food truck database project would likely include issues related to the management and tracking of food truck operations.
These could include:

1. Difficulty in keeping track of inventory and menu items
2. Inefficient processes for managing orders and payments
3. Difficulty in analysing sales data to make informed business decisions
4. Difficulty in keeping customers informed about food truck locations and menus
5. Difficulty in scheduling and coordinating food truck operations

Additionally, food truck operators may face challenges related to compliance with health and safety regulations, managing food truck staff and vendors, and dealing with competition and market trends. A food truck database project aims to address these challenges by providing a centralised system for managing and tracking food truck operations, thus helping to streamline the process and increase efficiency.

Business Problems Addressed

1. A source for information on Food Trucks across the city and a method to improve revenue for every FoodTruck.

2. Selection of locations for the Food Truck owners that are in line with the type of cuisine and income generated based on the number of orders.

3. Analysis of expenditure of each Foodtruck and systematically reducing expenditure thus improving business revenue.

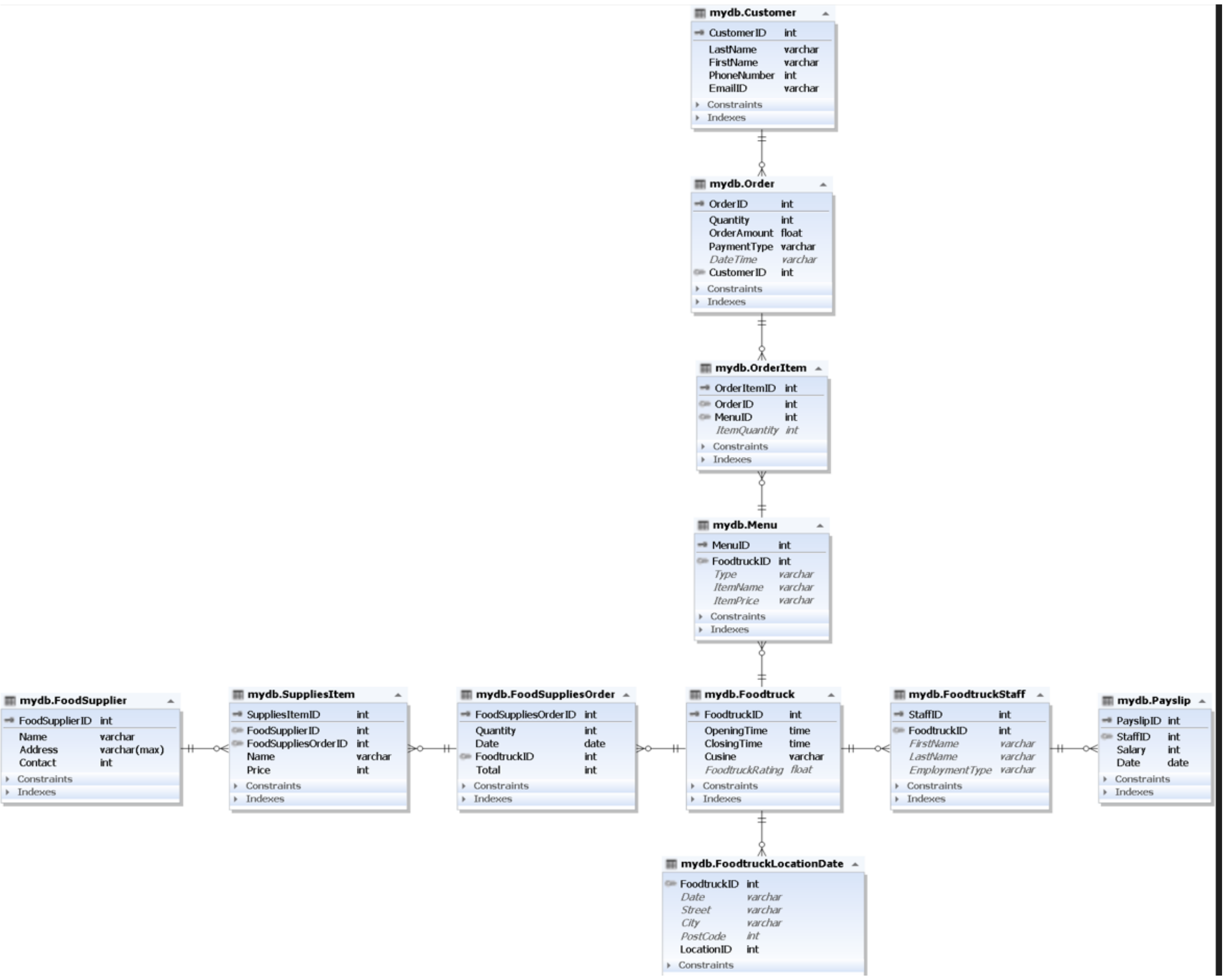
4. Analysis of Customer Orders and deciding on what is a popular food choice in a Neighbourhood.

Business Rules

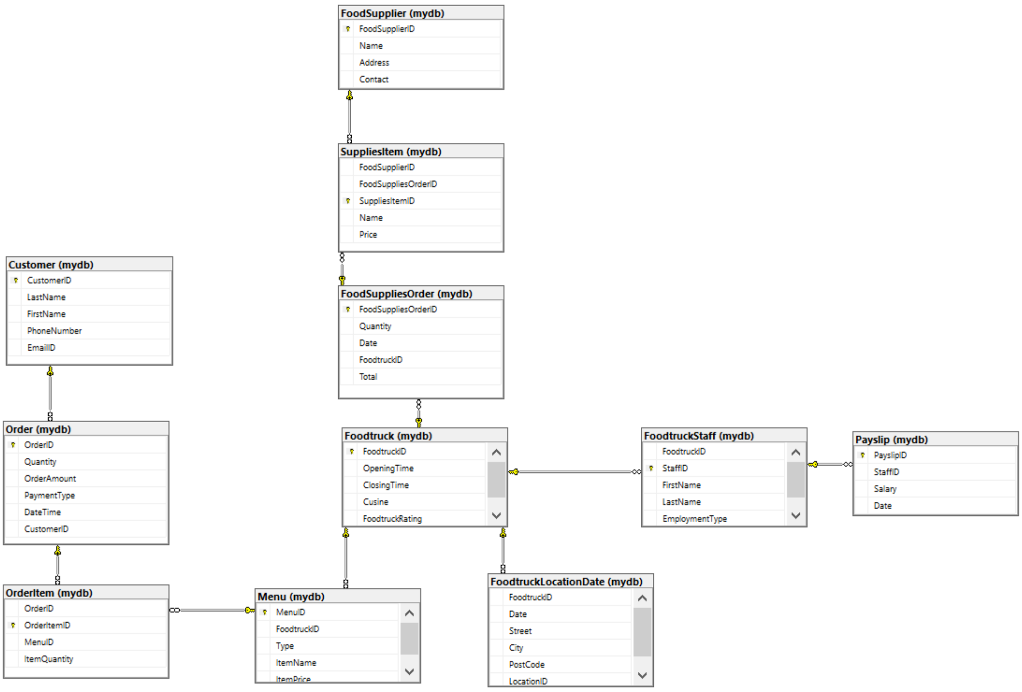
1. A customer can place multiple orders
2. One order can have multiple items
3. Menu consists of multiple food items

4. Food truck may offer multiple menu
5. Food truck offers food at a different location on a different date
6. Food truck can request one food supply order on a specific date
7. Each supplier can supply multiple food supply orders to food truck
8. Food supply orders consist of multiple quantities of food supply items.
9. Food trucks can have multiple staff at a time.
10. One payslip can be given to only one staff

ERD Diagram



Physical Diagram



Database DML

```
Insert into mydb.FoodtruckStaff values(1, 100, 'Weng', 'Simon', NULL);
Insert into mydb.FoodtruckStaff values(2, 101, 'Arsene', 'Wenger', NULL);
Insert into mydb.FoodtruckStaff values(3, 102, 'Nicolas', 'Lodeiro',NULL );
Insert into mydb.FoodtruckStaff values(4, 103, 'Phoebe', 'Buffet',NULL );
Insert into mydb.FoodtruckStaff values(5, 104, 'Joey', 'Tribbiani',NULL);
Insert into mydb.FoodtruckStaff values(6, 105, 'Harvey', 'Spectre',NULL);
Insert into mydb.FoodtruckStaff values(7, 106, 'Mike', 'Ross',NULL);
Insert into mydb.FoodtruckStaff values(8, 107, 'Charlie', 'Sheen',NULL);
Insert into mydb.FoodtruckStaff values(9, 108, 'Jesse', 'Pinkman', NULL);

Select * from mydb.FoodtruckStaff;
```

```
INSERT INTO mydb.foodtruck VALUES (1,'08:30:00','12:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (2,'07:30:00','11:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (3,'13:30:00','19:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (4,'15:00:00','20:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (5,'17:00:00','23:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (6,'09:00:00','15:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (7,'09:00:00','20:00:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (8,'00:30:00','10:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (9,'00:30:00','07:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (10,'07:30:00','10:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (11,'07:30:00','10:30:00','POL',0);
SELECT * FROM mydb.foodtruck;
```

```
Insert into mydb.FoodSupplier values(530, 'Greg Farms' , 'Wroclaw', 20);
Insert into mydb.FoodSupplier values(531, 'Jonny Sims' , 'Wroclaw', 39);
Insert into mydb.FoodSupplier values(532, 'Pollos Hermanos' , 'Wroclaw', 64);
Insert into mydb.FoodSupplier values(533, 'QFC' , 'Wroclaw', 70);
Insert into mydb.FoodSupplier values(534, 'RMDX' , 'Wroclaw', 33);
Insert into mydb.FoodSupplier values(535, 'TRPL' , 'Wroclaw', 24);
Insert into mydb.FoodSupplier values(536, 'FMCG' , 'Wroclaw', 50);
Insert into mydb.FoodSupplier values(537, 'Jose Depot' , 'Wroclaw', 10);
Insert into mydb.FoodSupplier values(538, 'Green Haven' , 'Wroclaw', 15);
Insert into mydb.FoodSupplier values(539, 'Gregory Farms' , 'Wroclaw', 40);
select * from mydb.FoodSupplier;
```

```
INSERT INTO mydb.customer
VALUES(11,'Venkataraman','Vijayakumar',486579740,'venkataraman.v@husky.neu.edu');
INSERT INTO mydb.customer VALUES(2,'Narra','Rohith
Reddy',987792324,'narra.r@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(3,'Venkat','Mithun',912354321,'mithun.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(4,'Hamilton','Lewis',948679740,'vic.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(5,'Vettel','Seb',948657098,'vettel.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(6,'Sainz','Carlos',948657097,'aman.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(7,'Raghuram','Arun',945437970,'raghu.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(8,'Kairnair','Ashvin',989989909,'kairnw.v@husky.neu.edu');
```

```
INSERT INTO mydb.customer
VALUES(9,'Itekela','Satya',900998761,'kata.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(10,'Verstappen','Max',948657970,'raman.v@husky.neu.edu');
SELECT * FROM mydb.customer;
GO
```

```
Insert into mydb.SuppliesItem values(530, 1,1, 'Onion', 20);
Insert into mydb.SuppliesItem values(531, 2,2, 'Cheese', 39);
Insert into mydb.SuppliesItem values(532, 3,3, 'Chciken', 64);
Insert into mydb.SuppliesItem values(533, 4,4, 'Rice', 70);
Insert into mydb.SuppliesItem values(534, 5,5,'Bell Pepper', 33);
Insert into mydb.SuppliesItem values(535, 6,6, 'Tomatoes', 24);
Insert into mydb.SuppliesItem values(536, 7,7,'Fish', 50);
Insert into mydb.SuppliesItem values(537, 8,8,'Taco Shells', 10);
Insert into mydb.SuppliesItem values(538, 4,9, 'Fetucchini', 15);
Insert into mydb.SuppliesItem values(539, 5,10, 'Pizza Dough', 40);
select * from mydb.SuppliesItem;
```

```
insert into mydb.FoodSuppliesOrder values(1,100, '2022-12-30', 10, 1000)
insert into mydb.FoodSuppliesOrder values(2, 100, '2022-12-30', 2, 1000)
insert into mydb.FoodSuppliesOrder values(3,  100, '2022-12-30', 3, 1000)
insert into mydb.FoodSuppliesOrder values(6, 100, '2022-12-30', 4, 1000)
insert into mydb.FoodSuppliesOrder values(4, 100, '2022-12-30', 5, 1000)
insert into mydb.FoodSuppliesOrder values(5, 100, '2022-12-30', 6, 1000)
insert into mydb.FoodSuppliesOrder values(7, 100, '2022-12-30', 7, 1000)
insert into mydb.FoodSuppliesOrder values(8,  100, '2022-12-30', 8, 1000)
insert into mydb.FoodSuppliesOrder values(9,  100, '2022-12-30', 9, 1000)
select * from mydb.FoodSuppliesOrder
```

```
INSERT INTO mydb.Menu VALUES(1,1,'Burger', 'AmericanBurger', 1);
INSERT INTO mydb.Menu VALUES(2,2,'Cluck It Up','Pasta',2);
INSERT INTO mydb.Menu VALUES(3,3,'Pimp My Rice','IND Rice',3);
INSERT INTO mydb.Menu VALUES(4,4,'Curry Up Now','Pasta',4);
INSERT INTO mydb.Menu VALUES(5,5,'Easy Slider','Pasta',5);
INSERT INTO mydb.Menu VALUES(6,6,'Grillenium Falcon','Pasta',6);
INSERT INTO mydb.Menu VALUES(7,7,'Hamborghini','Pasta',7);
INSERT INTO mydb.Menu VALUES(8,8,'Guac N Roll','Pasta',8);
INSERT INTO mydb.Menu VALUES(9,9,'I Dream of Weenie','Pasta',9);
INSERT INTO mydb.Menu VALUES(10,10,'Ms. Cheezious','Pasta',10);
SELECT * FROM mydb.Menu;
```

```
Insert into mydb.[Order]  values(2, 2, 10.00, 'card', '2019-08-24 20:36:23', 1);
Insert into mydb.[Order]  values( 3, 3, 40.00, 'venmo', '2019-11-28 12:38:27', 2);
Insert into mydb.[Order]  values( 4, 4, 100.0, 'card',  '2019-10-13 11:02:34', 3);
Insert into mydb.[Order]  values(5, 2, 10.0, 'venmo', '2019-11-07 14:26:07', 4);
Insert into mydb.[Order]  values( 6, 4, 400.4, 'Google Pay', '2019-03-27 15:36:10', 5);
Insert into mydb.[Order]  values( 7, 5, 1.00,'cash', '2019-02-05 19:28:10', 6);

SELECT * FROM mydb.[order];


Insert into mydb.FoodtruckLocationDate values(1, 98100, 'Northeastern', 'Seattle', 50205, 1)
Insert into mydb.FoodtruckLocationDate values(2, 98110, 'West Rochester', 'Seattle', 50206, 2)
Insert into mydb.FoodtruckLocationDate values(3, 98342, 'Lake City', 'Seattle', 50207, 3)
Insert into mydb.FoodtruckLocationDate values(4, 98934, 'Lloyd', 'Seattle', 50208, 4)
Insert into mydb.FoodtruckLocationDate values(5, 98245, 'Clive', 'Seattle', 50209, 5)
Insert into mydb.FoodtruckLocationDate values(6, 98234, 'Westminster', 'Seattle', 50210, 6)
Insert into mydb.FoodtruckLocationDate values(7, 98232, 'Buckingham', 'Seattle',50211, 7)
Insert into mydb.FoodtruckLocationDate values(8, 98167, 'Issaquah', 'Seattle', 50212, 8)
Insert into mydb.FoodtruckLocationDate values(9, 98104, 'Sammasish', 'Seattle', 50213, 9)
Insert into mydb.FoodtruckLocationDate values(10, 98196, 'Boren', 'Seattle', 50214, 10)
select * from mydb.FoodtruckLocationDate


insert into mydb.payslip values(101,101,2000,'10-01-2019')
insert into mydb.payslip values(102,102,2000,'10-01-2019')
insert into mydb.payslip values(103,103,2000,'10-01-2019')
insert into mydb.payslip values(104,104,2000,'10-01-2019')
insert into mydb.payslip values(105,105,2000,'10-01-2019')
insert into mydb.payslip values(106,106,2000,'10-01-2019')
insert into mydb.payslip values(107,107,2000,'10-01-2019')
insert into mydb.payslip values(108,108,2000,'10-01-2019')


select * from mydb.payslip;


Insert into mydb.FoodtruckStaff values(1, 100, 'Weng', 'Simon', NULL);
Insert into mydb.FoodtruckStaff values(2, 101, 'Arsene', 'Wenger', NULL);
Insert into mydb.FoodtruckStaff values(3, 102, 'Nicolas', 'Lodeiro',NULL );
Insert into mydb.FoodtruckStaff values(4, 103, 'Phoebe', 'Buffet',NULL );
Insert into mydb.FoodtruckStaff values(5, 104, 'Joey', 'Tribbiani',NULL);
Insert into mydb.FoodtruckStaff values(6, 105, 'Harvey', 'Spectre',NULL);
Insert into mydb.FoodtruckStaff values(7, 106, 'Mike', 'Ross',NULL);
Insert into mydb.FoodtruckStaff values(8, 107, 'Charlie', 'Sheen',NULL);
Insert into mydb.FoodtruckStaff values(9, 108, 'Jesse', 'Pinkman', NULL);

Select * from mydb.FoodtruckStaff;
```

```
Insert into mydb.FoodSupplier values(530, 'Greg Farms' , 'Wroclaw', 20);
Insert into mydb.FoodSupplier values(531, 'Jonny Sims' , 'Wroclaw', 39);
Insert into mydb.FoodSupplier values(532, 'Pollos Hermanos' , 'Wroclaw', 64);
Insert into mydb.FoodSupplier values(533, 'QFC' , 'Wroclaw', 70);
Insert into mydb.FoodSupplier values(534, 'RMDX' , 'Wroclaw', 33);
Insert into mydb.FoodSupplier values(535, 'TRPL' , 'Wroclaw', 24);
Insert into mydb.FoodSupplier values(536, 'FMCG' , 'Wroclaw', 50);
Insert into mydb.FoodSupplier values(537, 'Jose Depot' , 'Wroclaw', 10);
Insert into mydb.FoodSupplier values(538, 'Green Haven' , 'Wroclaw', 15);
Insert into mydb.FoodSupplier values(539, 'Gregory Farms' , 'Wroclaw', 40);
select * from mydb.FoodSupplier;
```

```
INSERT INTO mydb.foodtruck VALUES (1,'08:30:00','12:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (2,'07:30:00','11:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (3,'13:30:00','19:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (4,'15:00:00','20:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (5,'17:00:00','23:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (6,'09:00:00','15:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (7,'09:00:00','20:00:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (8,'00:30:00','10:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (9,'00:30:00','07:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (10,'07:30:00','10:30:00','IND',0);
INSERT INTO mydb.foodtruck VALUES (11,'07:30:00','10:30:00','POL',0);
SELECT * FROM mydb.foodtruck;

Insert into mydb.SuppliesItem values(530, 1 , 'Onion', 20, 3);
Insert into mydb.SuppliesItem values(531, 2 , 'Cheese', 39,5);
Insert into mydb.SuppliesItem values(532, 3 , 'Chciken', 64, 7);
Insert into mydb.SuppliesItem values(533, 4 , 'Rice', 70, 7);
Insert into mydb.SuppliesItem values(534, 5 , 'Bell Pepper', 33, 7);
Insert into mydb.SuppliesItem values(535, 6 , 'Tomatoes', 24, 8);
Insert into mydb.SuppliesItem values(536, 7 , 'Fish', 50, 9);
Insert into mydb.SuppliesItem values(537, 8, 'Taco Shells', 10, 20);
Insert into mydb.SuppliesItem values(538, 9 , 'Fetucchini', 15, 10);
Insert into mydb.SuppliesItem values(539, 10 , 'Pizza Dough', 40, 10);
select * from mydb.SuppliesItem;
```

```
insert into mydb.[OrderItem] values(3, 3, 3,5)
insert into mydb.[OrderItem] values(4, 4, 4,7)
insert into mydb.[OrderItem] values(5, 5, 5,7)
insert into mydb.[OrderItem] values(6, 6, 6,41)
insert into mydb.[OrderItem] values(7, 7, 7,7)
insert into mydb.[OrderItem] values(8, 8, 8,9)
select * from mydb.[OrderItem]
```

```
insert into mydb.payslip values(101,101,2000,'10-01-2019')
insert into mydb.payslip values(102,102,2000,'10-01-2019')
insert into mydb.payslip values(103,103,2000,'10-01-2019')
insert into mydb.payslip values(104,104,2000,'10-01-2019')
insert into mydb.payslip values(105,105,2000,'10-01-2019')
insert into mydb.payslip values(106,106,2000,'10-01-2019')
insert into mydb.payslip values(107,107,2000,'10-01-2019')
insert into mydb.payslip values(108,108,2000,'10-01-2019')


select * from mydb.payslip;
```

```
INSERT INTO mydb.customer
VALUES(11,'Venkataraman','Vijayakumar',486579740,'venkataraman.v@husky.neu.edu');
INSERT INTO mydb.customer VALUES(2,'Narra','Rohith
Reddy',987792324,'narra.r@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(3,'Venkat','Mithun',912354321,'mithun.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(4,'Hamilton','Lewis',948679740,'vic.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(5,'Vettel','Seb',948657098,'vettel.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(6,'Sainz','Carlos',948657097,'aman.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(7,'Raghuram','Arun',945437970,'raghu.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(8,'Kairnair','Ashvin',989989909,'kairnw.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(9,'Itekela','Satya',900998761,'kata.v@husky.neu.edu');
INSERT INTO mydb.customer
VALUES(10,'Verstappen','Max',948657970,'raman.v@husky.neu.edu');
SELECT * FROM mydb.customer;
GO
```

VIEWS

Salary expense

```
CREATE VIEW sum_salary_expense AS
```

```
select SUM(Salary) as Total_Salary_Expense from mydb.Payslip

select * from sum_salary_expense
```

Supply Expense

```
create view supplies_expense as
select sum(Price) supplies_expense from mydb.SuppliesItem

select * from supplies_expense
```

Employee Salary

```
CREATE VIEW Employee_Salary as
select FirstName, LastName, Salary from mydb.FoodtruckStaff
left join mydb.Payslip
ON mydb.Payslip.StaffID = mydb.FoodtruckStaff.StaffID
WHERE Salary is NOT NULL

GO
select * from Employee_Salary
```

Top Customer

```
CREATE VIEW top_customer as
select LastName, FirstName, Quantity from mydb.Customer
left join mydb.[Order]
ON mydb.[Order].[CustomerID] = mydb.Customer.CustomerID
WHERE Quantity IS NOT NULL
group by LastName, FirstName, Quantity
order by Quantity DESC
OFFSET 3 ROWS
FETCH NEXT 3 ROWS ONLY;

go
select * from top_customer
```

View IND Cuisine Food Truck

```
CREATE VIEW Cusine_IND AS
Select * FROM mydb.Foodtruck
WHERE Cusine = 'IND'
Go
Select * from Cusine_IND
```

Database DDL

```
CREATE DATABASE [FoodTruckDB_New]

GO

USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[Customer]    Script Date: 1/31/2023 3:23:33 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[Customer](
	[CustomerID] [int] NOT NULL,
	[LastName] [varchar](45) NOT NULL,
	[FirstName] [varchar](45) NOT NULL,
	[PhoneNumber] [int] NOT NULL,
	[EmailID] [varchar](50) NOT NULL,
 CONSTRAINT [PK__Customer__A4AE64B8DBB6D42A] PRIMARY KEY CLUSTERED
(
	[CustomerID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[FoodSupplier]    Script Date: 1/31/2023 3:23:53 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO
```

```
CREATE TABLE [mydb].[FoodSupplier](
	[FoodSupplierID] [int] NOT NULL,
	[Name] [varchar](50) NOT NULL,
	[Address] [varchar](max) NOT NULL,
	[Contact] [int] NOT NULL,
 CONSTRAINT [PK_FoodSupplier] PRIMARY KEY CLUSTERED
(
	[FoodSupplierID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO



USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[FoodSuppliesOrder]    Script Date: 1/31/2023 3:24:10 AM
******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[FoodSuppliesOrder](
	[FoodSuppliesOrderID] [int] NOT NULL,
	[Quantity] [int] NOT NULL,
	[Date] [date] NOT NULL,
	[FoodtruckID] [int] NOT NULL,
	[Total] [int] NOT NULL,
 CONSTRAINT [PK_FoodSuppliesOrder] PRIMARY KEY CLUSTERED
(
	[FoodSuppliesOrderID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[FoodSuppliesOrder]  WITH CHECK ADD  CONSTRAINT
[FK_FoodSuppliesOrder_Foodtruck] FOREIGN KEY([FoodtruckID])
REFERENCES [mydb].[Foodtruck] ([FoodtruckID])
GO
```

```
ALTER TABLE [mydb].[FoodSuppliesOrder] CHECK CONSTRAINT
[FK_FoodSuppliesOrder_Foodtruck]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[Foodtruck]    Script Date: 1/31/2023 3:24:23 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[Foodtruck](
          [FoodtruckID] [int] NOT NULL,
          [OpeningTime] [time](0) NOT NULL,
          [ClosingTime] [time](0) NOT NULL,
          [Cusine] [varchar](50) NOT NULL,
          [FoodtruckRating] [float] NULL,
 CONSTRAINT [PK__Foodtruc__D4204EC1178644FB] PRIMARY KEY CLUSTERED
(
          [FoodtruckID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[Foodtruck] ADD  CONSTRAINT
[DF__Foodtruck__Foodt__625A9A57]  DEFAULT ((0.0000)) FOR [FoodtruckRating]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[FoodtruckLocationDate]    Script Date: 1/31/2023 3:24:38 AM
******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[FoodtruckLocationDate](
          [FoodtruckID] [int] NOT NULL,
	[Date] [varchar](50) NULL,
	[Street] [varchar](50) NULL,
	[City] [varchar](50) NULL,
	[PostCode] [int] NULL,
	[LocationID] [int] NOT NULL
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[FoodtruckLocationDate]  WITH CHECK ADD  CONSTRAINT
[fk_Foodtruck_has_Location_date_Foodtruck1] FOREIGN KEY([FoodtruckID])
REFERENCES [mydb].[Foodtruck] ([FoodtruckID])
GO

ALTER TABLE [mydb].[FoodtruckLocationDate] CHECK CONSTRAINT
[fk_Foodtruck_has_Location_date_Foodtruck1]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[FoodtruckStaff]    Script Date: 1/31/2023 3:24:49 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[FoodtruckStaff](
	[FoodtruckID] [int] NOT NULL,
	[StaffID] [int] NOT NULL,
	[FirstName] [varchar](50) NULL,
	[LastName] [varchar](50) NULL,
	[EmploymentType] [varchar](50) NULL,
 CONSTRAINT [PK_FoodtruckStaff] PRIMARY KEY CLUSTERED
(
	[StaffID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[FoodtruckStaff]  WITH CHECK ADD  CONSTRAINT
[FK_FoodtruckStaff_Foodtruck1] FOREIGN KEY([FoodtruckID])
REFERENCES [mydb].[Foodtruck] ([FoodtruckID])
GO
```

```
ALTER TABLE [mydb].[FoodtruckStaff] CHECK CONSTRAINT
[FK_FoodtruckStaff_Foodtruck1]
GO

ALTER TABLE [mydb].[FoodtruckStaff]  WITH CHECK ADD  CONSTRAINT
[empTypeCheck] CHECK  (([EmploymentType]='Owner' OR [EmploymentType]='Chef' OR
[EmploymentType]='Driver'))
GO

ALTER TABLE [mydb].[FoodtruckStaff] CHECK CONSTRAINT [empTypeCheck]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[Menu]    Script Date: 1/31/2023 3:24:59 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[Menu](
	[MenuID] [int] NOT NULL,
	[FoodtruckID] [int] NOT NULL,
	[Type] [varchar](50) NULL,
	[ItemName] [varchar](50) NULL,
	[ItemPrice] [varchar](50) NULL,
 CONSTRAINT [PK_Menu] PRIMARY KEY CLUSTERED
(
	[MenuID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[Menu]  WITH CHECK ADD  CONSTRAINT [FK_Menu_Foodtruck]
FOREIGN KEY([FoodtruckID])
REFERENCES [mydb].[Foodtruck] ([FoodtruckID])
GO

ALTER TABLE [mydb].[Menu] CHECK CONSTRAINT [FK_Menu_Foodtruck]
GO


USE [FoodTruckDB_New]
```

```
GO

/****** Object:  Table [mydb].[Order]    Script Date: 1/31/2023 3:25:12 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[Order](
	[OrderID] [int] NOT NULL,
	[Quantity] [int] NOT NULL,
	[OrderAmount] [float] NOT NULL,
	[PaymentType] [varchar](45) NOT NULL,
	[DateTime] [varchar](50) NULL,
	[CustomerID] [int] NOT NULL,
 CONSTRAINT [PK__Order__C3905BAF3A8A2AEC] PRIMARY KEY CLUSTERED
(
	[OrderID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[Order] ADD  CONSTRAINT [DF__Order__OrderAmou__69FBBC1F]
DEFAULT ((0)) FOR [OrderAmount]
GO

ALTER TABLE [mydb].[Order]  WITH CHECK ADD  CONSTRAINT [fk_Order_Customer1]
FOREIGN KEY([CustomerID])
REFERENCES [mydb].[Customer] ([CustomerID])
GO

ALTER TABLE [mydb].[Order] CHECK CONSTRAINT [fk_Order_Customer1]
GO

ALTER TABLE [mydb].[Order]  WITH CHECK ADD  CONSTRAINT [paymentTypeCheck]
CHECK  ((([PaymentType]='Google Pay' OR [PaymentType]='card' OR [PaymentType]='cash'
OR [PaymentType]='venmo'))
GO

ALTER TABLE [mydb].[Order] CHECK CONSTRAINT [paymentTypeCheck]
GO


USE [FoodTruckDB_New]
GO
```

```
/****** Object:  Table [mydb].[OrderItem]    Script Date: 1/31/2023 3:25:25 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[OrderItem](
	[OrderID] [int] NOT NULL,
	[OrderItemID] [int] NOT NULL,
	[MenuID] [int] NOT NULL,
	[ItemQuantity] [int] NULL,
 CONSTRAINT [PK__Menu__C99ED250AF9CF6AA] PRIMARY KEY CLUSTERED
(
	[OrderItemID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[OrderItem]  WITH CHECK ADD  CONSTRAINT
[FK_OrderItem_Menu] FOREIGN KEY([MenuID])
REFERENCES [mydb].[Menu] ([MenuID])
GO

ALTER TABLE [mydb].[OrderItem] CHECK CONSTRAINT [FK_OrderItem_Menu]
GO

ALTER TABLE [mydb].[OrderItem]  WITH CHECK ADD  CONSTRAINT
[FK_OrderItem_Order] FOREIGN KEY([OrderID])
REFERENCES [mydb].[Order] ([OrderID])
GO

ALTER TABLE [mydb].[OrderItem] CHECK CONSTRAINT [FK_OrderItem_Order]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[Payslip]    Script Date: 1/31/2023 3:26:46 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO
```

```
CREATE TABLE [mydb].[Payslip](
	[PayslipID] [int] NOT NULL,
	[StaffID] [int] NOT NULL,
	[Salary] [int] NOT NULL,
	[Date] [date] NOT NULL,
 CONSTRAINT [PK_Payslip] PRIMARY KEY CLUSTERED
(
	[PayslipID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO

ALTER TABLE [mydb].[Payslip]  WITH CHECK ADD  CONSTRAINT
[FK_Payslip_FoodtruckStaff1] FOREIGN KEY([StaffID])
REFERENCES [mydb].[FoodtruckStaff] ([StaffID])
GO

ALTER TABLE [mydb].[Payslip] CHECK CONSTRAINT [FK_Payslip_FoodtruckStaff1]
GO


USE [FoodTruckDB_New]
GO

/****** Object:  Table [mydb].[SuppliesItem]    Script Date: 1/31/2023 3:26:57 AM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [mydb].[SuppliesItem](
	[FoodSupplierID] [int] NOT NULL,
	[FoodSuppliesOrderID] [int] NOT NULL,
	[SuppliesItemID] [int] NOT NULL,
	[Name] [varchar](50) NOT NULL,
	[Price] [int] NOT NULL,
 CONSTRAINT [PK_SuppliesItem] PRIMARY KEY CLUSTERED
(
	[SuppliesItemID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY =
OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON,
OPTIMIZE_FOR_SEQUENTIAL_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
```

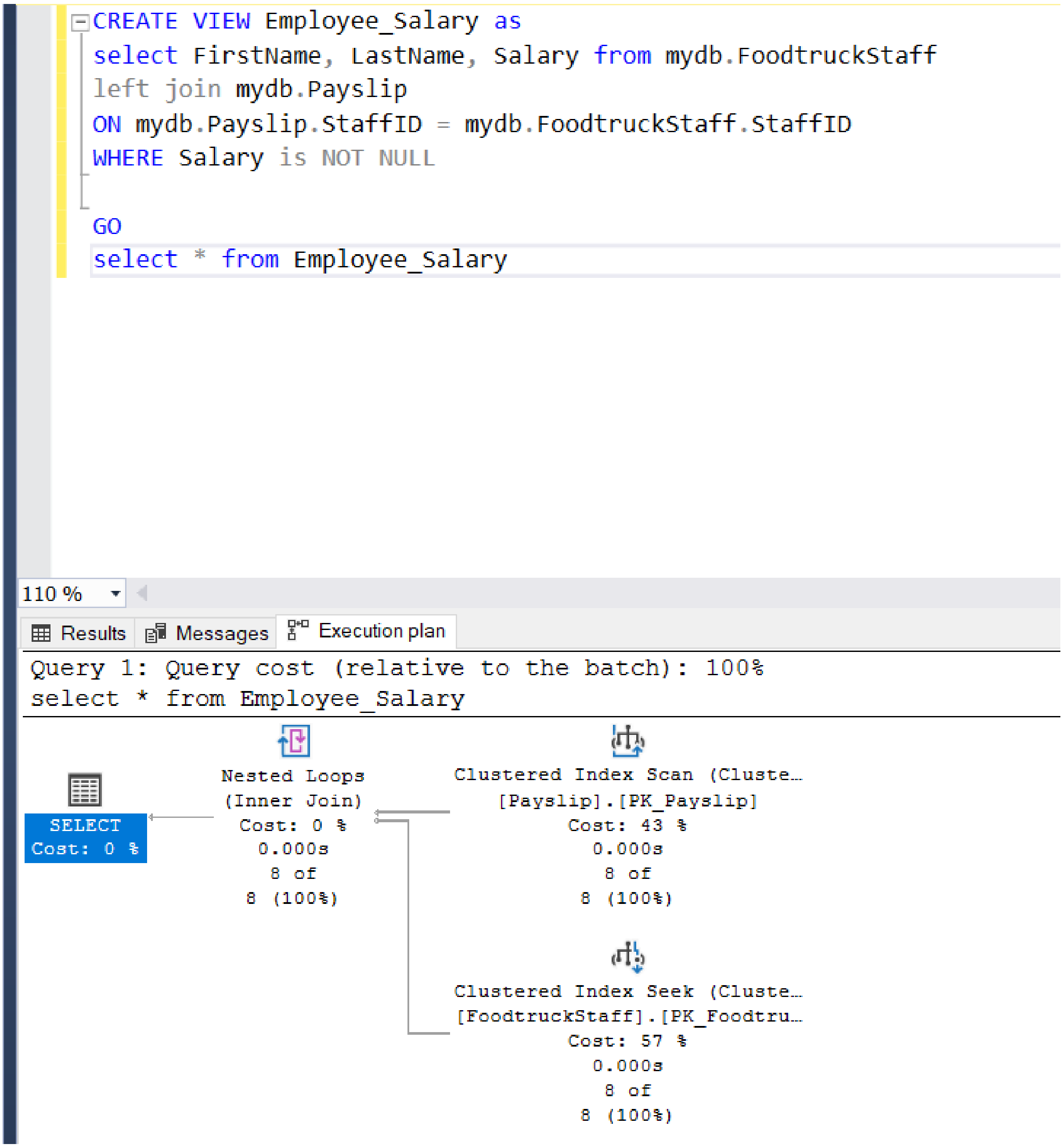
```
ALTER TABLE [mydb].[SuppliesItem]  WITH CHECK ADD  CONSTRAINT
[FK_SuppliesItem_FoodSupplier1] FOREIGN KEY([FoodSupplierID])
REFERENCES [mydb].[FoodSupplier] ([FoodSupplierID])
GO

ALTER TABLE [mydb].[SuppliesItem] CHECK CONSTRAINT
[FK_SuppliesItem_FoodSupplier1]
GO

ALTER TABLE [mydb].[SuppliesItem]  WITH CHECK ADD  CONSTRAINT
[FK_SuppliesItem_FoodSuppliesOrder] FOREIGN KEY([FoodSuppliesOrderID])
REFERENCES [mydb].[FoodSuppliesOrder] ([FoodSuppliesOrderID])
GO

ALTER TABLE [mydb].[SuppliesItem] CHECK CONSTRAINT
[FK_SuppliesItem_FoodSuppliesOrder]
GO
```

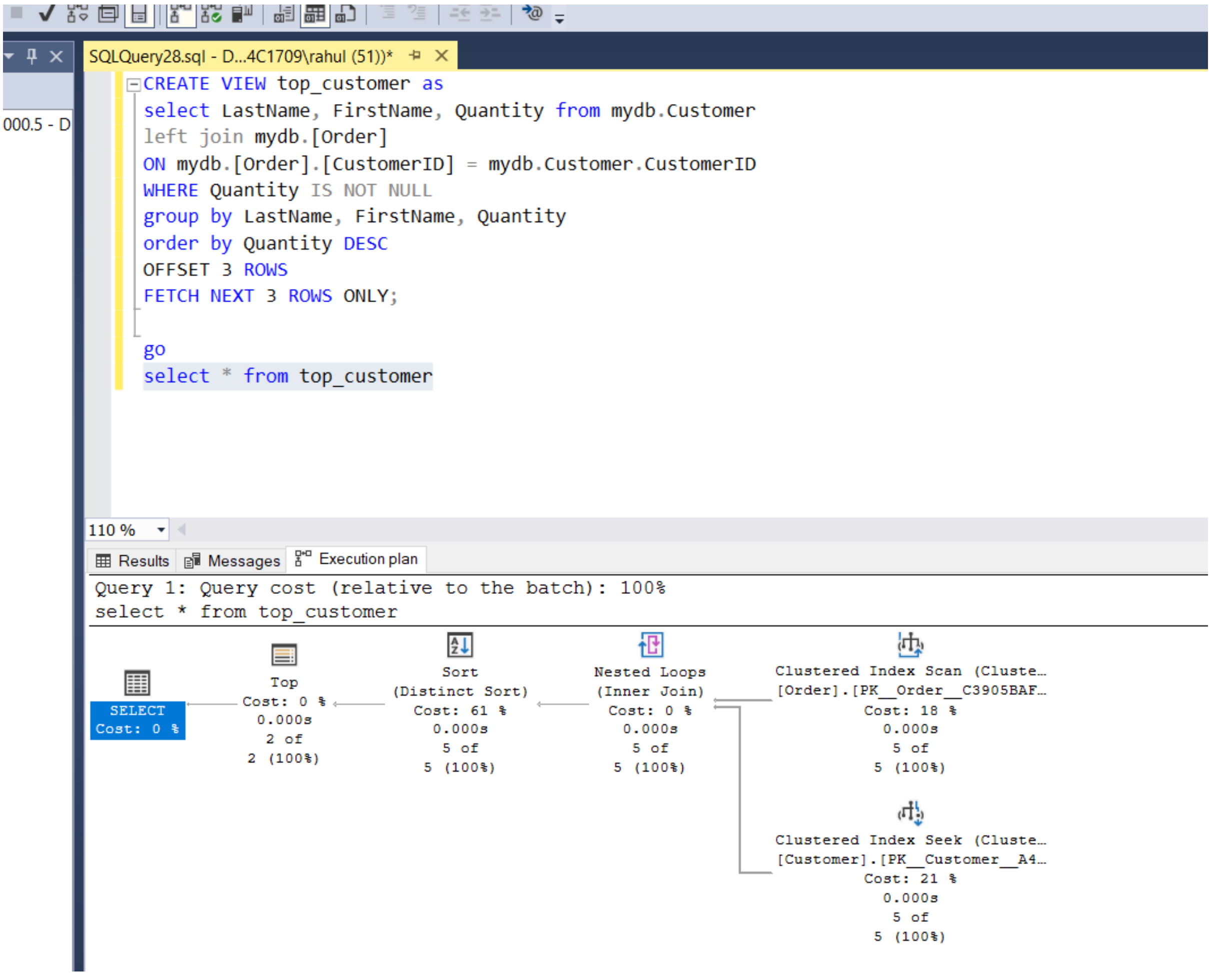
Query Execution Plan



```
CREATE VIEW Employee_Salary as
select FirstName, LastName, Salary from mydb.FoodtruckStaff
left join mydb.Payslip
ON mydb.Payslip.StaffID = mydb.FoodtruckStaff.StaffID
WHERE Salary is NOT NULL

GO
select * from Employee_Salary
```

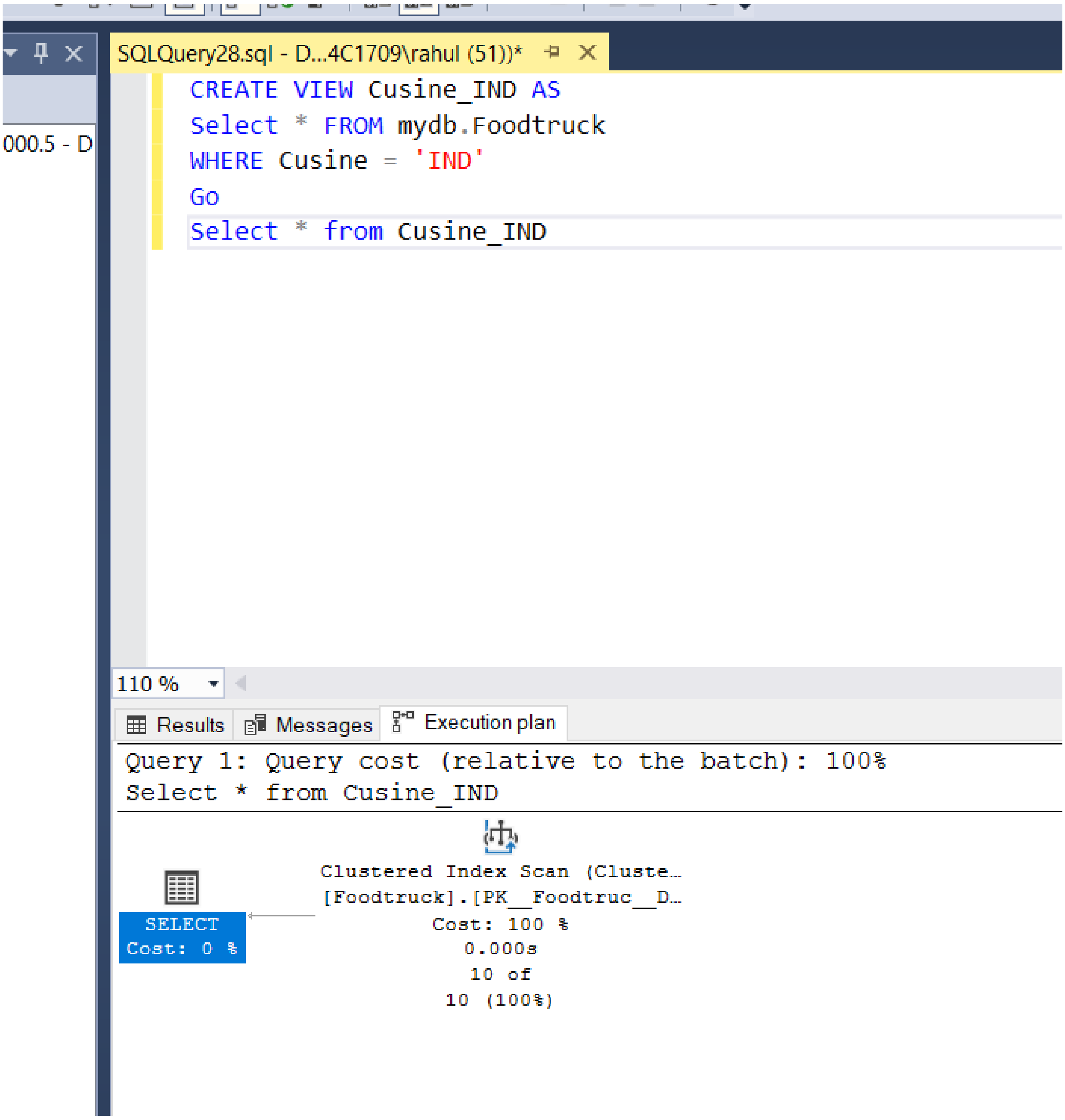
```
Query 1: Query cost (relative to the batch): 100%
select * from Employee_Salary
```



```
CREATE VIEW top_customer as
select LastName, FirstName, Quantity from mydb.Customer
left join mydb.[Order]
ON mydb.[Order].[CustomerID] = mydb.Customer.CustomerID
WHERE Quantity IS NOT NULL
group by LastName, FirstName, Quantity
order by Quantity DESC
OFFSET 3 ROWS
FETCH NEXT 3 ROWS ONLY;

go
select * from top_customer
```

```
Query 1: Query cost (relative to the batch): 100%
select * from top_customer
```



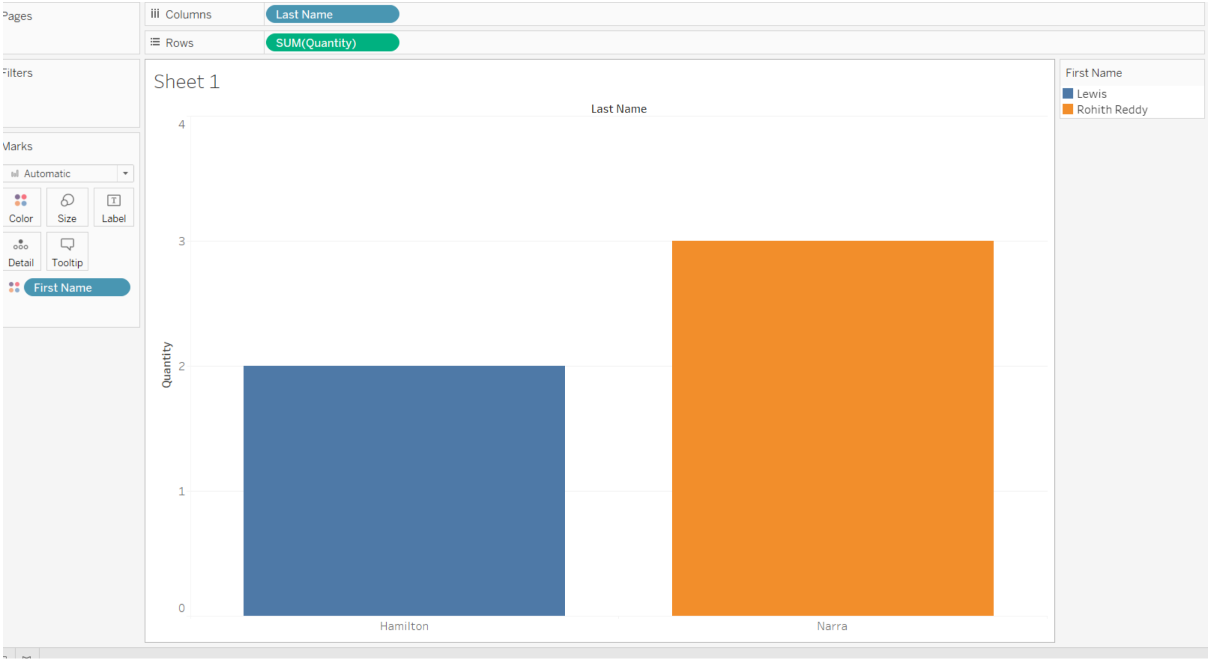
SQLQuery28.sql - D...4C1709\rahul (51))*
000.5 - D
110 %
Results
Messages
Execution plan

```
CREATE VIEW Cusine_IND AS
Select * FROM mydb.Foodtruck
WHERE Cusine = 'IND'
Go
Select * from Cusine_IND
```

```
Query 1: Query cost (relative to the batch): 100%
Select * from Cusine_IND
```

SELECT
Cost: 0 %
Clustered Index Scan (Cluste…
[Foodtruck].[PK__Foodtruc__D…
Cost: 100 %
0.000s
10 of
10 (100%)

Tableau Analysis



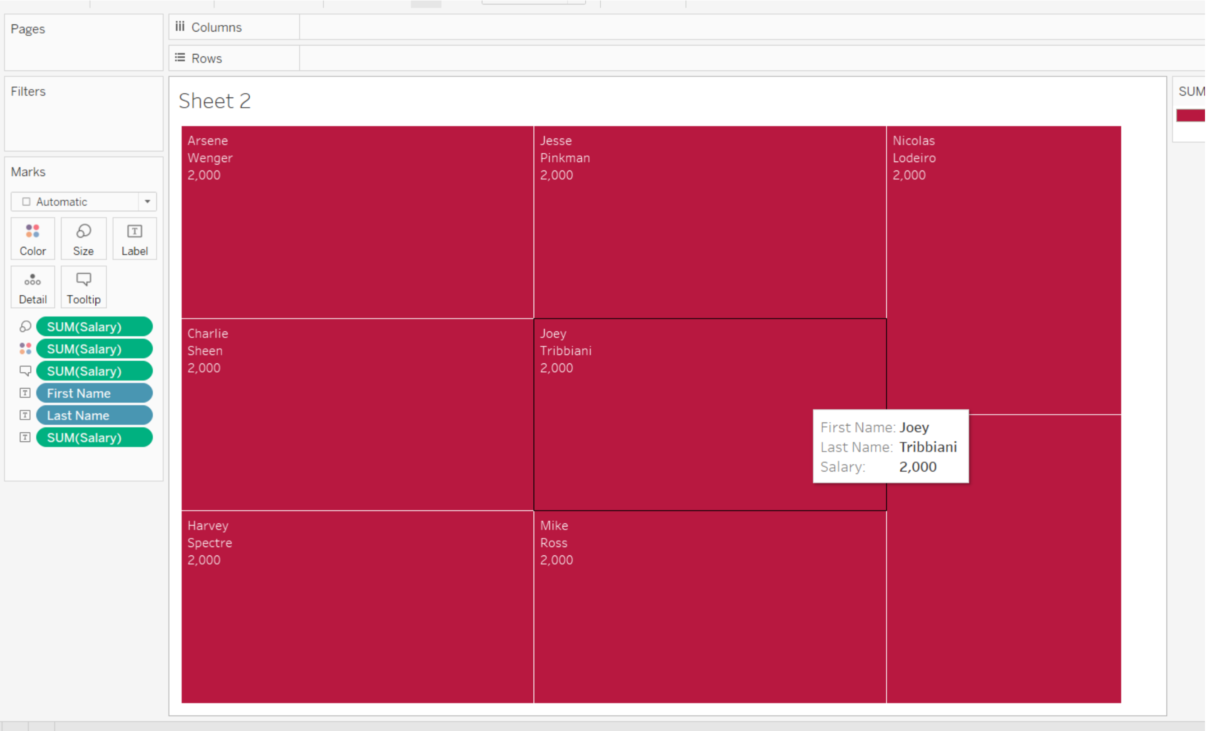
Columns: Last Name

Rows: SUM(Quantity)

Marks: Automatic

First Name

Sheet 1



Marks: Automatic

SUM(Salary)

SUM(Salary)

SUM(Salary)

First Name

Last Name

SUM(Salary)

Sheet 2

First Name: Joey
Last Name: Tribbiani
Salary: 2,000

Conclusion

This project helped me to gain the knowledge and skills required for designing a database with modelling methods and tools, which are helpful in understanding and defining the customer requirements clearly and helps the developer to identify the main use cases of the system.

A food truck database design project can provide valuable insights into the food truck industry. The database should include important information such as food truck names, location, menu items, operating hours, and owner information. This information can be used to track sales, analyse customer preferences, and optimise food truck routes. By effectively organising and analysing data, the food truck database can support informed business decisions and drive success for food truck owners.

Overall it was a good experience of learning and identifying the problem domain and helping organisation in making data driven decision