



GREEN BITE



Sahar Alkhadouri
Fatima AL-Amri
Maryam AL-Aamri



REQUIREMENTS SPECIFICATION

GreenBite is a vegetarian dine-in restaurant that needs a system to manage data related to its staff, customers, tables, orders, bills, and menu items. Each staff member has a unique ID, name, position, contact number, email, hire date, salary, gender, and status, and they are responsible for serving tables and taking customer orders. Customers also have a unique customer ID, name, email, phone number, gender, and membership level. Inside the restaurant, every table has a table ID, number of guests, serving time, and the staff member assigned to serve it. Customers seated at a table can place orders, and each order includes an order ID, order timestamp, payment method, payment status, order status, and any special notes. An order is placed by one customer and taken by one staff member, although each customer may place multiple orders and each staff member can serve many customers. Orders consist of multiple vegetarian menu items, and each menu item—identified by item ID, name, category, calories, description, and price—can appear in many different orders. These details are stored in the order-item list, where the system tracks the quantity of each item ordered. After finishing their meal, the system generates a bill for the customer, recording the bill ID, order ID, bill date, and total amount. Each order has exactly one bill, and the bill reflects all vegetarian items included in the order.



Staff Table .1			
Constraints	Field Size	Data Type	Field Name
PRIMARY KEY, AUTO_INCREMENT	-	INT	Staff_ID
NOT NULL	50	VARCHAR	Name
NOT NULL ('Male', 'Female')	-	ENUM	Gender
CHECK (Age > 15)	-	INT	Age
NOT NULL ('Chef', 'Waiter', 'Manager')	-	ENUM	Position
CHECK (Salary >= 10)	10,2	DECIMAL	Salary
-	20	VARCHAR	Contact_Number
UNIQUE	30	VARCHAR	Email
NOT NULL	-	DATE	Hire_Date
DEFAULT 'Active' ('Active','Inactive')	-	ENUM	Status

Customer Table .2			
Constraints	Field Size	Data Type	Field Name
PRIMARY KEY, AUTO_INCREMENT	-	INT	Customer_ID
NOT NULL	50	VARCHAR	Name
UNIQUE	20	VARCHAR	Phone
UNIQUE	30	VARCHAR	Email
NOT NULL ('Male','Female')	-	ENUM	Gender
DEFAULT 'Regular' ('Regular','Silver','Gold')	-	ENUM	Membership_Level





Menu_Item Table .3

Constraints	Field Size	Data Type	Field Name
PRIMARY KEY, AUTO_INCREMENT	-	INT	Item_ID
NOT NULL	20	VARCHAR	Item_Name
NOT NULL ('Salad','Main Course','Smoothie','Dessert')	-	ENUM	Category
CHECK (Calories >= 0)	-	INT	Calories
CHECK (Price > 0)	8,2	DECIMAL	Price
	-	TEXT	Description

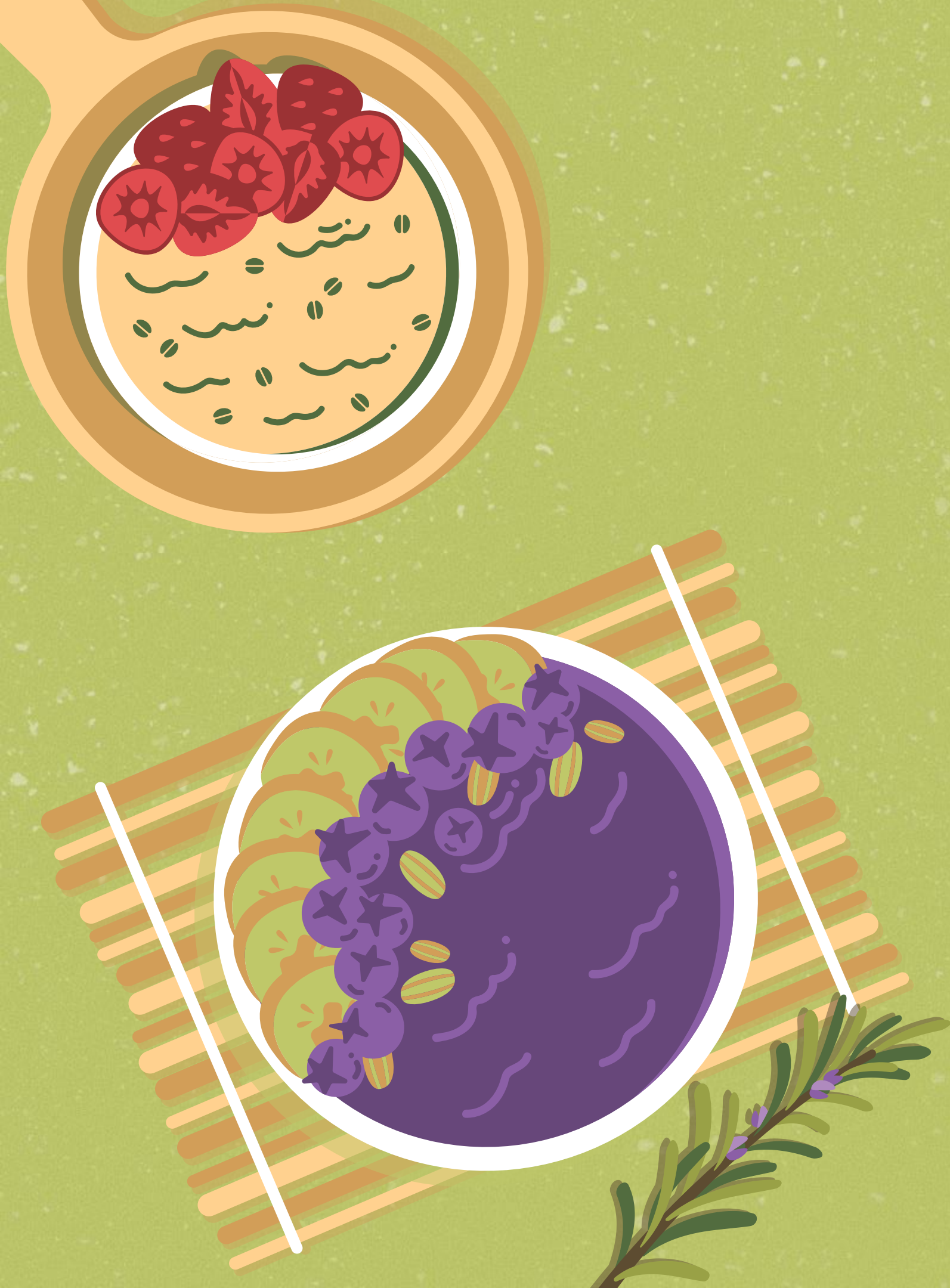
Order_r Table .4

Constraints	Field Size	Data Type	Field Name
PRIMARY KEY, AUTO_INCREMENT	-	INT	Order_ID
FOREIGN KEY REFERENCES Customer(Customer_ID) ON DELETE SET NULL ON UPDATE CASCADE	-	INT	Customer_ID
FOREIGN KEY REFERENCES Staff(Staff_ID) ON DELETE SET NULL ON UPDATE CASCADE	-	INT	Staff_ID
DEFAULT CURRENT_TIMESTAMP	-	DATETIME	Order_Timestamp
NOT NULL ('Cash','Card')	-	ENUM	Payment_Method
DEFAULT 'Unpaid' ('Paid','Unpaid')	-	ENUM	Payment_Status
DEFAULT 'Pending' ('Pending','Preparing','Ready','Delivered','Cancelled')	-	ENUM	Order_Status

Order_Item Table .5			
Constraints	Field Size	Data Type	Field Name
	-	INT	OrderItem_ID
FOREIGN KEY REFERENCES Order_r(Order_ID) ON DELETE CASCADE ON UPDATE CASCADE	-	INT	Order_ID
FOREIGN KEY REFERENCES Menu_Item(Item_ID) ON DELETE RESTRICT ON UPDATE CASCADE	-	INT	Item_ID
NOT NULL, CHECK (Quantity > 0)	-	INT	Quantity
(Order_ID, Item_ID)	-	Composite	Primary Key

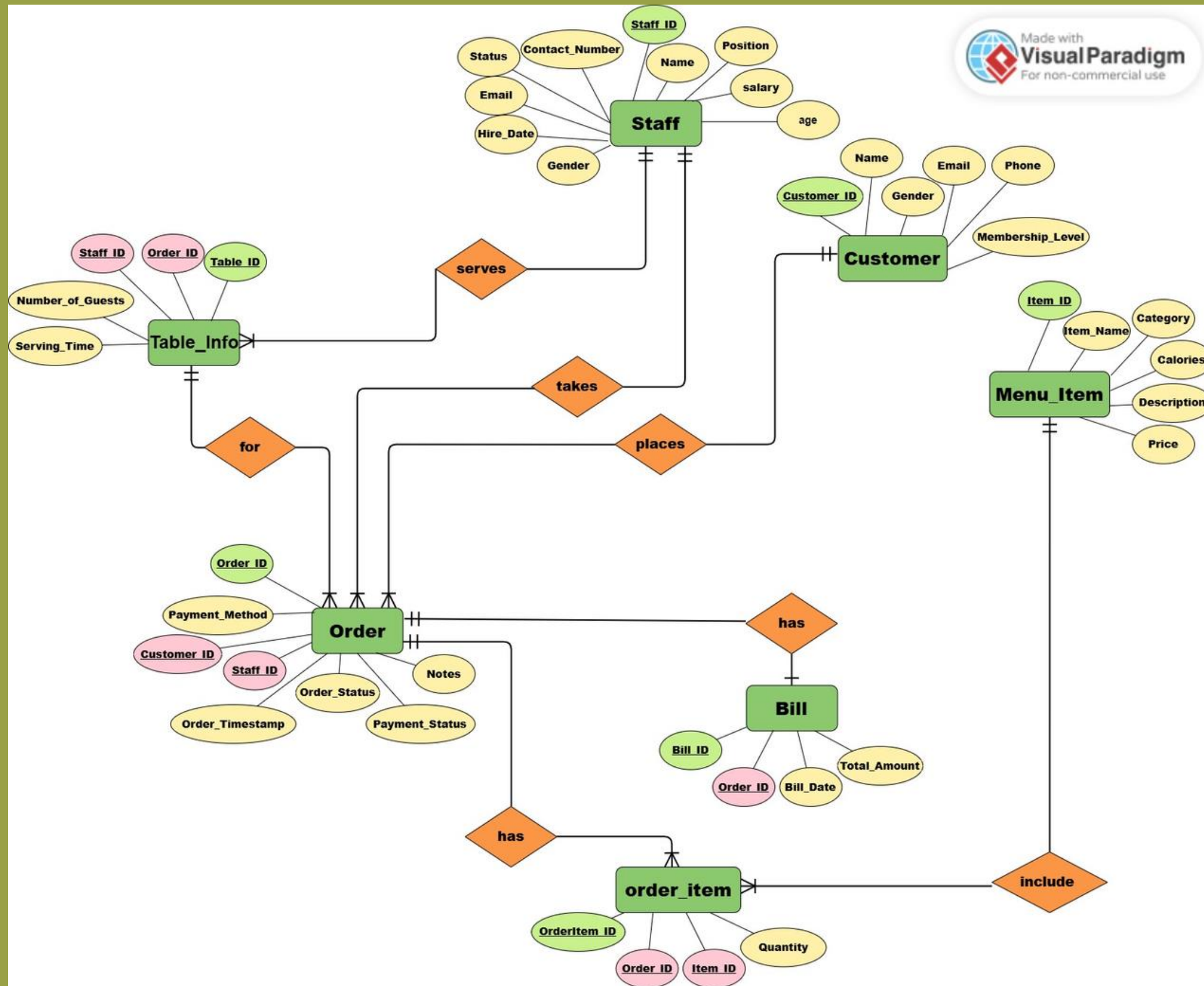
Bill Table .6			
Constraints	Field Size	Data Type	Field Name
PRIMARY KEY, AUTO_INCREMENT	-	INT	Bill_ID
UNIQUE, FOREIGN KEY REFERENCES Order_r(Order_ID) ON DELETE CASCADE ON UPDATE CASCADE	-	INT	Order_ID
DEFAULT CURRENT_TIMESTAMP	-	DATETIME	Bill_date
CHECK (Total_Amount >= 0)	10,2	DECIMAL	Total_Amount



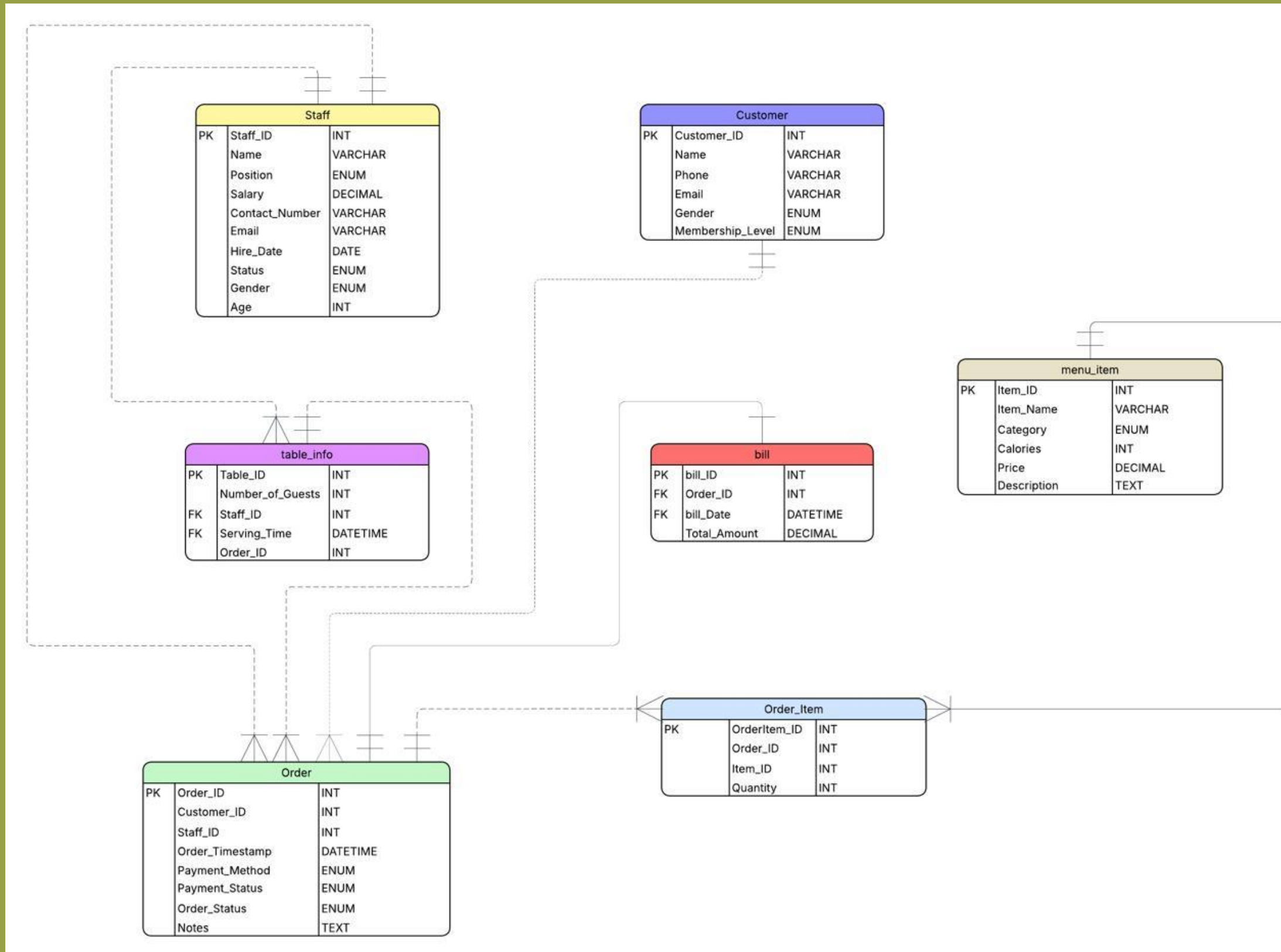


Table_info .7				
Constraints	Field Size	Data Type	Field Name	
PRIMARY KEY, AUTO_INCREMENT	-	INT	Table_ID	
CHECK (Number_of_Guests > 0)	-	INT	Number_of_Guests	
FOREIGN KEY REFERENCES Staff(Staff_ID) ON DELETE SET NULL ON UPDATE CASCADE	-	INT	Staff_ID	
	-	DATETIME	Serving_Time	
FOREIGN KEY REFERENCES Order_r(Order_ID) ON DELETE CASCADE ON UPDATE CASCADE	-	INT	Order_ID	

CONCEPTUAL ER DIAGRAM(ERD)



LOGICAL ER DIAGRAM(ERD)



Get all active staff who are Chefs

```
282 • SELECT *
283 FROM Staff
284 WHERE Status = 'Active' AND Position = 'Chef';
285
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

	Staff_ID	Name	Gender	Age	Position	Salary	Contact_Number	Email	Hire_Date	Status
▶	1	Ahmed Al-Balushi	Male	32	Chef	850.00	91234567	ahmed.b@resto.com	2020-05-12	Active
	4	Huda Al-Kindi	Female	30	Chef	820.00	93654321	huda.k@resto.com	2022-02-20	Active
	7	Talib Al-Busaidi	Male	38	Chef	880.00	91998877	talib.b@resto.com	2019-06-07	Active
	9	Khalid Al-Rashdi	Male	33	Chef	870.00	93778899	khalid.r@resto.com	2020-09-14	Active
	12	Layla Al-Habsi	Female	31	Chef	890.00	92775533	layla.h@resto.com	2020-11-05	Active
	16	Reem Al-Mahdi	Female	33	Chef	860.00	93445522	reem.m@resto.com	2021-02-02	Active
	19	Zahra Al-Saadi	Female	34	Chef	900.00	93774455	zahra.s@resto.com	2020-08-25	Active
	20	Ibrahim Al-Araimi	Male	39	Chef	910.00	93223344	ibrahim.a@resto.com	2018-05-15	Active
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Get all menu items that are Main Course and cost more than 4 OMR

```
288 • SELECT * FROM Menu_Item
289 WHERE Category = 'Main Course' AND Price > 4.00;
290
```

Result Grid | Filter Rows: | Edit: | Export/Import: |

	Item_ID	Item_Name	Category	Calories	Price	Description
▶	3	Veggie Burger	Main Course	450	4.80	Plant-based burger with fries
	6	Pasta Primavera	Main Course	480	4.50	Vegetables and olive oil pasta
	20	Veg Lasagna	Main Course	490	4.90	Layers of veggies and cheese
•	NULL	NULL	NULL	NULL	NULL	NULL

Get all customers who are Gold OR Silver members

293 • `SELECT *FROM Customer`
294 `WHERE Membership_Level = 'Gold'OR Membership_Level = 'Silver';`

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap

	Customer_ID	Name	Phone	Email	Gender	Membership_Level
▶	1	Hassan Said	91111111	hassan.s@example.com	Male	Gold
	3	Ahmed Nasser	93333333	ahmed.n@example.com	Male	Silver
	4	Maryam Saif	94444444	maryam.s@example.com	Female	Gold
	6	Fatma Rashid	96666666	fatma.r@example.com	Female	Silver
	7	Salim Khalid	97777777	salim.k@example.com	Male	Gold
	9	Ali Hamad	99999999	ali.h@example.com	Male	Silver
	10	Sara Fahad	91010101	sara.f@example.com	Female	Gold
	12	Khalid Salim	93030303	khalid.s@example.com	Male	Silver
	13	Fatma Zayed	94040404	fatma.z@example.com	Female	Gold
	15	Amina Juma	96060606	amina.j@example.com	Female	Gold
	16	Mohammed Said	97070707	mohammed.s@example....	Male	Silver
	18	Rashid Khalifa	99090909	rashid.k@example.com	Male	Silver
	19	Huda Salim	90101010	huda.s@example.com	Female	Gold
•	NULL	NULL	NULL	NULL	NULL	NULL

Get all bills where the total amount is between 10 and 30 OMR

298 • `SELECT *FROM Bill`
299 `WHERE Total_Amount BETWEEN 10 AND 30;`
300

Result Grid | Filter Rows: | Edit: | Wrap

	Bill_ID	Order_ID	Bill_date	Total_Amount
▶	6	6	2025-03-25 16:10:00	10.20
	13	13	2025-07-03 12:55:00	10.10
•	NULL	NULL	NULL	NULL

Get staff who have positions in this list ('Chef', 'Manager')

```
303 • SELECT * FROM Staff
304 WHERE Position IN ('Chef', 'Manager');
305
```

	Staff_ID	Name	Gender	Age	Position	Salary	Contact_Number	Email	Hire_Date	Status
▶	1	Ahmed Al-Balushi	Male	32	Chef	850.00	91234567	ahmed.b@resto.com	2020-05-12	Active
	3	Salim Al-Maskari	Male	41	Manager	1200.00	92112233	salim.m@resto.com	2018-07-01	Active
	4	Huda Al-Kindi	Female	30	Chef	820.00	93654321	huda.k@resto.com	2022-02-20	Active
	7	Talib Al-Busaidi	Male	38	Chef	880.00	91998877	talib.b@resto.com	2019-06-07	Active
	9	Khalid Al-Rashdi	Male	33	Chef	870.00	93778899	khalid.r@resto.com	2020-09-14	Active
	10	Sara Al-Siyabi	Female	35	Manager	1150.00	93556677	sara.s@resto.com	2019-03-03	Active
	12	Layla Al-Habsi	Female	31	Chef	890.00	92775533	layla.h@resto.com	2020-11-05	Active
	13	Hassan Al-Shanfari	Male	36	Manager	1220.00	93332244	hassan.s@resto.com	2017-10-10	Active
	16	Reem Al-Mahdi	Female	33	Chef	860.00	93445522	reem.m@resto.com	2021-02-02	Active
	17	Salma Al-Khalili	Female	40	Manager	1180.00	93669988	salma.k@resto.com	2019-09-12	Active
	19	Zahra Al-Saadi	Female	34	Chef	900.00	93774455	zahra.s@resto.com	2020-08-25	Active
	20	Ibrahim Al-Araimi	Male	39	Chef	910.00	93223344	ibrahim.a@resto.com	2018-05-15	Active
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Calculate the number and percentage of staff in each job position

```
392 • SELECT
393     Position,
394     COUNT(*) AS Total_Employees,
395     CONCAT(
396         ROUND( (COUNT(*) / (SELECT COUNT(*) FROM Staff)) * 100, 1 ),
397         '%'
398     ) AS Percentage
399 FROM Staff
400 GROUP BY Position
401 ORDER BY Total_Employees DESC;
402
```

	Position	Total_Employees	Percentage
▶	Chef	8	40.0%
	Waiter	8	40.0%
	Manager	4	20.0%

Update menu item 'Veggie Burger' (new price and description)

```
313 • UPDATE Menu_Item
314     SET Price = 4.100,
315         Description = 'Fresh vegetarian addition to our November special menu'
316     WHERE Item_Name = 'Veggie Burger';
317
318 • select * from Menu_Item;
```

Item_ID	Item_Name	Category	Calories	Price	Description
1	Greek Salad	Salad	250	2.80	Fresh lettuce, feta, olives
2	Fruit Smoothie	Smoothie	180	3.20	Mixed tropical fruits
3	Veggie Burger	Main Course	450	4.10	Fresh vegetarian addition to our November spe...
4	Mango Lassi	Smoothie	200	2.50	Mango yogurt drink
5	Falafel Wrap	Main Course	420	3.60	Falafel with tahini sauce
6	Pasta Primavera	Main Course	480	4.50	Vegetables and olive oil pasta

Show orders with customer and staff names (INNER JOIN)

```
321 • SELECT
322     O.Order_ID,
323     C.Name AS Customer_Name,
324     S.Name AS Staff_Name,
325     O.Payment_Method,
326     O.Payment_Status,
327     O.Order_Status
328 FROM Order_r O
329 INNER JOIN Customer C ON O.Customer_ID = C.Customer_ID
330 INNER JOIN Staff S ON O.Staff_ID = S.Staff_ID
331 ORDER BY O.Order_ID;
```

Order_ID	Customer_Name	Staff_Name	Payment_Method	Payment_Status	Order_Status
1	Hassan Said	Fatma Al-Harthy	Card	Paid	Delivered
2	Aisha Al-Habsi	Rashid Al-Hinai	Cash	Paid	Delivered
3	Ahmed Nasser	Maryam Al-Shibli	Card	Paid	Delivered
4	Maryam Saif	Aisha Al-Lawati	Card	Paid	Ready
5	Yousef Ali	Ali Al-Mahrouqi	Cash	Paid	Delivered
6	Fatma Rashid	Layla Al-Habsi	Card	Paid	Pending
7	Salim Khalid	Yousef Al-Bahrani	Card	Paid	Delivered
8	Mona Talib	Nasser Al-Amri	Cash	Paid	Delivered
9	Ali Hamad	Maha Al-Mamari	Card	Paid	Delivered
10	Sara Fahad	Khalid Al-Rashdi	Cash	Paid	Preparing
11	Layla Khalil	Reem Al-Mahdi	Card	Paid	Delivered

Average salary by position

```
366 • SELECT
367     Position,
368     ROUND(AVG(Salary), 2) AS Avg_Salary
369 FROM Staff
370 GROUP BY Position;
371
```

Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

	Position	Avg_Salary
►	Chef	872.50
	Waiter	425.00
	Manager	1187.50

MAX and MIN total amount in bills(info)

```
438 • SELECT
439     b.Bill_ID,
440     b.Total_Amount,
441     b.Bill_Date,
442     c.Name AS Customer_Name,
443     s.Name AS Staff_Name,
444     o.Payment_Method
445 FROM Bill b
446 JOIN Order_r o ON b.Order_ID = o.Order_ID
447 LEFT JOIN Customer c ON o.Customer_ID = c.Customer_ID
448 LEFT JOIN Staff s ON o.Staff_ID = s.Staff_ID
449 WHERE b.Total_Amount = (SELECT MAX(Total_Amount) FROM Bill)
450     OR b.Total_Amount = (SELECT MIN(Total_Amount) FROM Bill);
451
```

Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

	Bill_ID	Total_Amount	Bill_Date	Customer_Name	Staff_Name	Payment_Method
►	5	5.80	2025-03-07 11:20:00	Yousef Ali	Rashid Al-Hinai	Cash
	6	10.20	2025-03-25 16:10:00	Fatma Rashid	Fatma Al-Harthy	Card

Which waiter has the most orders (only Position = 'Waiter')

```
393 -- Update orders 2, 4, 6 to be handled by waiter with Staff_ID = 2
394 • UPDATE Order_r
395 SET Staff_ID = 2
396 WHERE Order_ID IN (2, 4, 6);
397
398 -- Update orders 5, 7 to be handled by waiter with Staff_ID = 5
399 • UPDATE Order_r
400 SET Staff_ID = 5
401 WHERE Order_ID IN (5, 7);
402
403 • SELECT
404     s.Staff_ID,
405     s.Name AS Waiter_Name,
406     COUNT(o.Order_ID) AS Total_Orders
407 FROM Staff s
408 LEFT JOIN Order_r o ON s.Staff_ID = o.Staff_ID
409 WHERE s.Position = 'Waiter'
410 GROUP BY s.Staff_ID, s.Name
411 ORDER BY Total_Orders DESC;
412
```

Result Grid

Filter Rows:

Export

Wrap Cell Content

	Staff_ID	Waiter_Name	Total_Orders
▶	2	Fatma Al-Harthy	4
	5	Rashid Al-Hinai	2
	6	Maryam Al-Shibli	1
	14	Maha Al-Mamari	1
	18	Nasser Al-Amri	1
	8	Aisha Al-Lawati	0
	11	Ali Al-Mahrouqi	0
	15	Yousef Al-Bahrani	0

Count each menu item and sum the total quantity ordered

```
415 • SELECT m.Item_ID,m.Item_Name, SUM(oi.Quantity) AS Total_Quantity
416 FROM Order_Item oi
417 JOIN Menu_Item m ON oi.Item_ID = m.Item_ID
418 GROUP BY oi.Item_ID, m.Item_Name
419 ORDER BY Total_Quantity DESC;
420
421
```

Result Grid

Filter Rows:

Export

Wrap Cell Content

	Item_ID	Item_Name	Total_Quantity
▶	5	Falafel Wrap	4
	11	Fruit Salad	4
	3	Veggie Burger	3
	4	Mango Lassi	3
	10	Chocolate Cake	3
	20	Veg Lasagna	3
	1	Greek Salad	2
	2	Fruit Smoothie	2
	6	Pasta Primavera	2
	7	Caesar Salad	2
	8	Avocado Toast	2
	9	Strawberry Milk...	2
	12	Lemon Mint Juice	2
	14	Pancakes	2
	15	Grilled Veg San...	2
	16	Banana Smoothie	2
	18	Quinoa Salad	2
	19	Fruit Tart	2
	13	Mushroom Soup	1
	17	Oat Bowl	1

Orders count per staff, only staff with more than 3 orders
(using HAVING)

```
454 • SELECT
455     S.Staff_ID,
456     S.Name AS Staff_Name,
457     COUNT(O.Order_ID) AS Orders_Count
458 FROM Staff S
459 JOIN Order_r O ON S.Staff_ID = O.Staff_ID
460 GROUP BY S.Staff_ID, S.Name
461 HAVING COUNT(O.Order_ID) > 3
462 ORDER BY Orders_Count DESC;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	Staff_ID	Staff_Name	Orders_Count
▶	2	Fatma Al-Harthy	4

Menu items sorted by price
(ascending)

```
466 • SELECT
467     Item_ID,
468     Item_Name,
469     Category,
470     Price
471 FROM Menu_Item
472 ORDER BY Price ASC;
```

Result Grid

Filter Rows:

Edit:

Export/Import:

	Item_ID	Item_Name	Category	Price
▶	12	Lemon Mint Juice	Smoothie	2.30
	4	Mango Lassi	Smoothie	2.50
	16	Banana Smoothie	Smoothie	2.60
	11	Fruit Salad	Dessert	2.70
	1	Greek Salad	Salad	2.80
	8	Avocado Toast	Main Course	2.90
	17	Oat Bowl	Dessert	2.90
	7	Caesar Salad	Salad	3.00
	2	Fruit Smoothie	Smoothie	3.20
	18	Quinoa Salad	Salad	3.30
	9	Strawberry Milks...	Smoothie	3.40
	15	Grilled Veg Sand...	Main Course	3.50
	5	Falafel Wrap	Main Course	3.60
	13	Mushroom Soup	Main Course	3.80
	14	Pancakes	Dessert	3.90
	19	Fruit Tart	Dessert	4.00
	3	Veggie Burger	Main Course	4.10
	10	Chocolate Cake	Dessert	4.20
	6	Pasta Primavera	Main Course	4.50
	20	Veg Lasagna	Main Course	4.90
★	NULL	NULL	NULL	NULL

Create a view that shows the total sales for each month

```
483 • CREATE VIEW MonthlySales AS
484 SELECT
485     DATE_FORMAT(Bill_date, '%Y-%m') AS Month,
486     SUM(Total_Amount) AS Total
487 FROM Bill
488 GROUP BY DATE_FORMAT(Bill_date, '%Y-%m');
489
490 /* Display the monthly sales summary */
491 • SELECT * FROM MonthlySales;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Month	Total
▶	2025-01	15.10
	2025-02	16.30
	2025-03	16.00
	2025-04	17.10
	2025-05	14.50
	2025-06	18.00
	2025-07	17.60
	2025-08	15.10
	2025-09	17.10
	2025-10	18.20

Compare membership levels by showing the count and percentage of customers in each level

```
494 • SELECT
495     Membership_Level,
496     COUNT(*) AS Total_Customers,
497     CONCAT(ROUND((COUNT(*) / (SELECT COUNT(*) FROM Customer)) * 100, 0), '%') AS Percentage
498 FROM Customer
499 GROUP BY Membership_Level
500 ORDER BY Total_Customers DESC;
501
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	Membership_Level	Total_Customers	Percentage
▶	Gold	7	35%
	Regular	7	35%
	Silver	6	30%

Calculate how many orders were paid using each payment method, and show the percentage

```
532 • SELECT
533     Payment_Method,
534     COUNT(*) AS Total,
535     CONCAT(ROUND(COUNT(*) * 100 / (SELECT COUNT(*) FROM Order_r), 1), '%') AS Percentage
536 FROM Order_r
537 GROUP BY Payment_Method;
538
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Payment_Method	Total	Percentage
▶	Card	12	60.0%
	Cash	8	40.0%

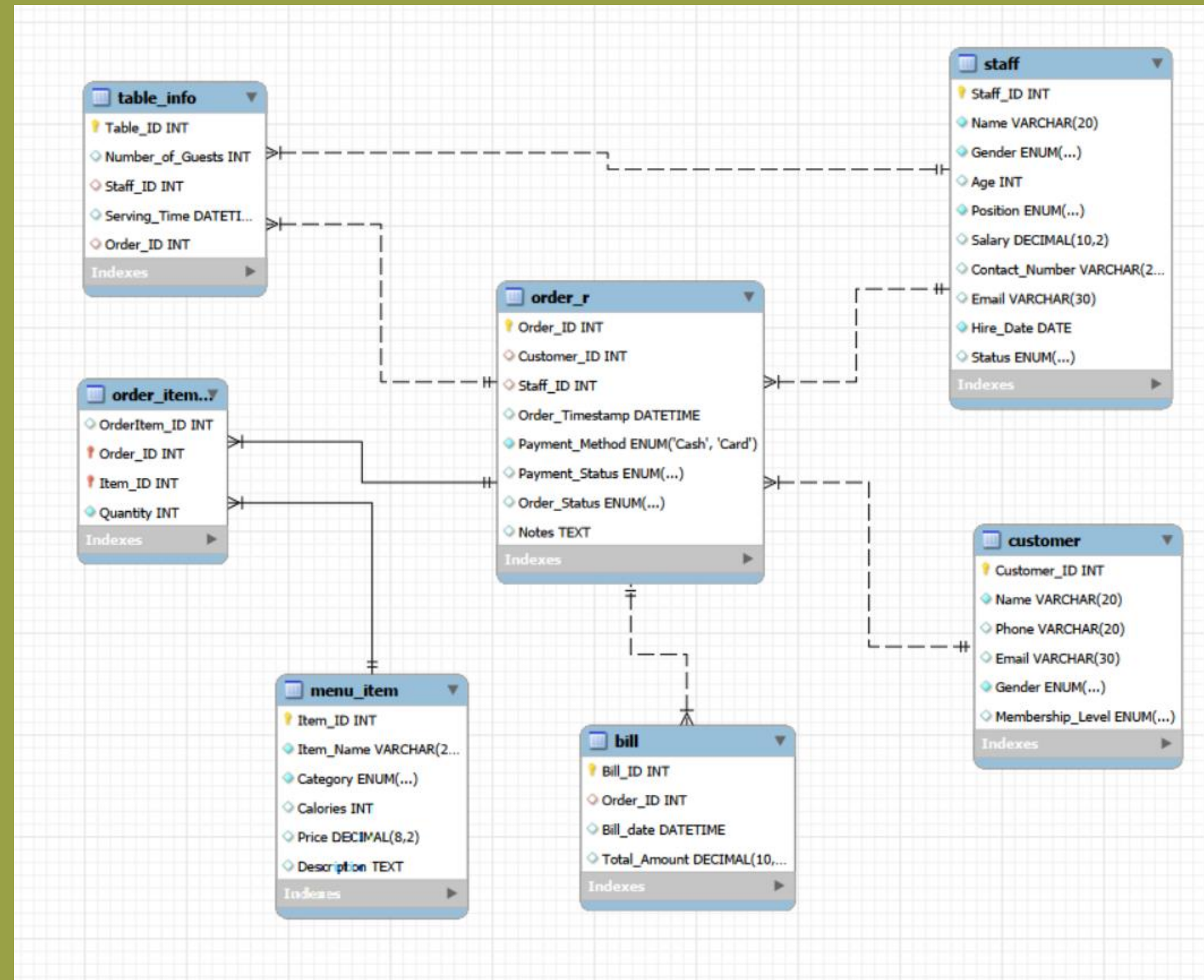
Find the relationship between the number of guests at a table and the average number of items ordered.

```
568 • SELECT
569     Number_of_Guests,
570     ROUND(AVG(Total_Items)) AS Avg_Items_Ordered
571 FROM (
572     SELECT
573         T.Table_ID,
574         T.Number_of_Guests,
575         SUM(OI.Quantity) AS Total_Items
576     FROM Table_info T
577     JOIN Order_Item OI
578         ON T.Order_ID = OI.Order_ID
579     GROUP BY T.Table_ID, T.Number_of_Guests
580 ) AS Sub
581 GROUP BY Number_of_Guests
582 ORDER BY Number_of_Guests;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	Number_of_Guests	Avg_Items_Ordered
▶	2	2
	3	2
	4	2
	5	2

PHYSICAL ERD





CONCLUSION

- A professional SQL-based database was designed and implemented for a vegetarian restaurant.
- Analysis showed no direct relationship between the number of guests and the number of items ordered.
- Card payment is the most commonly used payment method among customers.
- Balanced staffing between chefs and waiters supports better service.
- The project highlights the importance of databases in improving restaurant management and supporting decision-making.