Dine Navigator: Comprehensive Dining Experience Platform

Basi di dati modul 2

SAPIENZA
UNIVERSITÀ DI ROMA

Mateo Ramón
24/06/2022

Specification Of Requirements

This is the database of a platform that allows restaurants to expose the daily menus.

- The database must have information about the **restaurants** (name, telephone, address, city, number of forks). The forks of a restaurant have to be between 1-5

- The **cities** must have a name, obviously, but we also need to know the **area** in which it is located.

- Each restaurant, for each day, can publish:
    - A **menu**, consisting of **first** dish, **second** dish, **photo** of each, its price and **Time zone** (morning, noon or evening).
    - a **dish of the day**, with the same information: price, photo and service strip.

- Description of each **dish** and, whether by **menu** or **dish of the day**, it is necessary to say whether coffee and / or drinks are included in the price

- This information has to be published 2 weeks before the offer.

- Customers, anonymously, must be able to **rate** the menus or dishes of the day, in the form of a score (scale 1 to 5), and must also enter the date on which the score was made that has to be after the day of the offer.

- For customers, they can be able to consult the offer by **type** of cuisine. That is, they will order restaurants from a Korean cuisine, for example. Or, Asian cuisine, in which case the Korean, Japanese, Chinese, Thai, etc have to appear.

- Customers should be able to check the Staff of every restaurant, this Staff can be cook, waiter, chef manager and owner, and rating each one. Suppose that each person can only have one function or position in the company

Meeting Minimum Requirements

Point	Description	Project
A	The first version of the diagram presents 12 Entities	22 classes
B	The first version of the diagram contains both generalizations and IS-A	Yes
C	The Diagram presents more than one cycle if it is interpreted as a graph	Yes
D	All the cardinalities presented in class were used	Yes
E	In the diagram 2 optional attributes (in X and in the Y relationship) and 1 multi-valued attribute (in X), then transformed into an entity in the restructuring	Yes
F	There are more than 4 external constraints, discussed below	5 external contraints
G	There is a table of volumes accompanied by an explanation of the volumes themselves	Yes
H	About x queries were handled in the project, each one has an indication of their frequency, in the table of accesses to operations	Yes
H.1	Many of the queries contain more than one aggregate operator	Yes
H.II	There are many nested queries, even 3 levels	Yes
H.III	About half of the queries were processed through the use of views	Yes
H.IV	By removing the inserts, all operations involve at least two tables	Yes

Conceptual Design

1st version

Glossary of Terms

Term	Description
Area	Part of Mallorca's territory
City	Large town
Kitchen style	General style of cooking depending on the continent
Subtype	A more specific style of cooking depending on the country or region
Offer	Offer of a restaurant a specific day
Menu	Cart formed by the 3 dish of a full meal
Dish	A dish in gastronomy is a specific food preparation, ready to eat or to be served.
First	First dish of the menu, normally smaller and introductory
Second	Main dish of the meal
Desset	Last dish, normally small portion of something sweet
Dish of the Day	Unique dish that is offered apart of the menu
Time Zone	Differentiation of the differents moments for eating during the day
Evaluation	Score that a client has given to an offer
Dish photo	Picture of a dish
Restaurant	Establishment were you can buy and consume prepared food
Service	Score that each customer gives to every Staff
Waiter	Those attending to customers by supplying them with food and drink as requested
Chef	Is a trained professional cook and tradesman who is proficient in all aspects of food preparation
Manager	Managing waiters
Owner	Owner of the restaurant
Cook	Individuals who prepare food for consumption in restaurants.
Person	Individual who works in a restaurant

Clases

Restaurant
This class represents a restaurant in our system.
➢ NameRest → Name of the restaurant.
➢ Telephone → Restaurant telephone.
➢ Addresses → Restaurant addresses.
➢ Forks → Forks that the restaurant has (restaurant score).

KitchenStyle
KitchenStyle stores the restaurant's cooking specialty (Japanese, Chinese, Catalan, etc.). Being a numbered list.
➢ TypeCooking → Restaurant specialization (Chinese, Asian, Afghan, etc.)

Subtype
Stores the subtype of the kitchen style.
➢SubName → Name of the subtype.

Zone
Save information related to a geographic area.
➢ ZoneName → Name of the zone.
➢ Description → Brief description of the main characteristics of the area.

City
City of our database.
➢ CityName → Name of the city.

Offer
Offer of the restaurant on a specific day.
➢ Date → Date of the offer.
➢ Coffee → Boolean value indicating whether or not coffee is included.
➢ Drink → Boolean value indicating whether or not drinks are included.
➢ Price → Offer Price.

Evaluation
Score made by a customer on an offer.
➢ Date → Date of valuation.
➢ Score → Score of the related offer. The score can range from 1 to 5.

TimeZone
Time zone where an offer is made. This is a stand-alone class for better database maintenance (for example, if you want to add a new time zone or modify existing ones).
➢ Time_zone → Service Strip (can be Noon, Morning or Night)

Dish
Dish of the general menu.
➢ DishName → Plate Name.
➢ Description → Description of the dish offered.

First
First dish.
➢ FirstName → Plate Name.

Second
Second dish.
➢ SecondName → Plate Name.

Dessert
Dessert.
➢ DessertName → Plate Name.

Menu
Inherits the attributes of the offer class to obtain the remaining required information.

DishDay
A restaurant dish of the day is related to a dish of the general menu. Inherits the attributes of the offer class to obtain the remaining required information.

DishPhoto
Each restaurant may have a different photo for the same dish.
➢ Photo → Photo of the dish.

Person
Person designated to a restaurant.
➢IdPerson.
➢NamePerson
➢Surname

Manager
Managing everything except the kitchen.

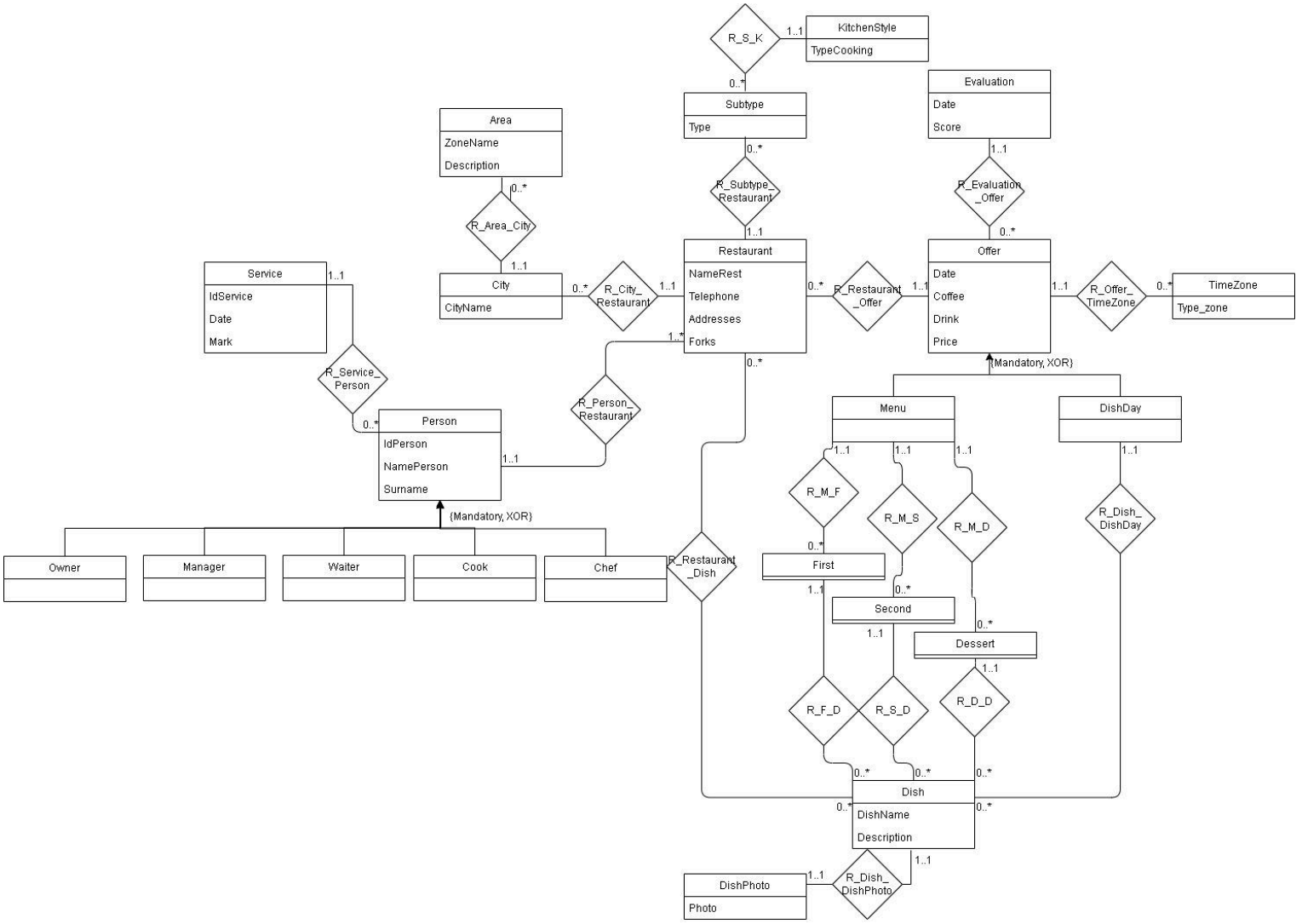
Waiter
Serving dishes.

Cook
Cooking dishes.

Cheff
Managing kitchen.

Owner
Owner of the restaurant.
Service
Ratings of every worker.
➢IdService
➢Mark
➢Date

First ER diagram



Dictionary of Data

Class

Class	Description	External links	Identificator
Area	Part of Mallorca's territory		NameArea
City	Large town		NameCity
KitchenStyle	General style of cooking depending on the continent		TypeCooking
Subtype	A more specific style of cooking depending on the country or region		IdSubttype
Offer	Offer of a restaurant a specific day	This information has to be published 2 weeks before the offer.	IdOffer
Menu	Cart formed by the 3 dish of a full meal		IdMenu
Dish	A dish in gastronomy is a specific food preparation, ready to eat or to be served.		NameDish
First	First dish of the menu, normally smaller and introductory		IdFirst
Second	Main dish of the meal		IdSecond
Dessert	Last dish, normally small portion of something sweet		IdDessert
DishDay	Unique dish that is offered apart of the menu		IdPdd
TimeZone	Differentiation of the moments for eating		Time_Zone

	during the day		
Evaluation	Score that a client has given to an offer	-Must enter the date on which the score was made, that has to be after the day of the offer. -The score has to be between 1-5.	IdEvaluation
Dish photo	Picture of a dish		IdPhoto
Restaurant	Establishment were you can buy and consume prepared food	The forks of a restaurant have to be between 1-5.	IdRestaurant
Service	Score that each customer gives to every Staff	The mark has to be between 1-5	IdService
Waiter	Those attending to customers by supplying them with food and drink as requested		IdPerson
Chef	Is a trained professional cook and tradesman who is proficient in all aspects of food preparation		IdPerson
Manager	Managing waiters		IdPerson
Owner	Owner of the restaurant		IdPerson
Cook	Individuals who prepare food for consumption in restaurants.		IdPerson
Person	Individual who works in a restaurant		IdPerson

Relations

Relation	Description	External constraints	Classes associate
R_Area_City	Cities that an Area can have		Area_City
R_Subtype_ Restaurant			Kitchenstyle_ Restaurant
R_Restaurant _Offer	Offer that a restaurant have for a specific day		Restaurant _Offer
R_Restaurant _Offer	All the evaluation to that specific offer		Restaurant _Offe
R_Offer_ TimeZone	The time zone of the offer		Offer_ TimeZone
R_Dish_ DishPhoto	Photo of the dish		Dish_ DishPhoto
R_M_F	First dish of the menu		Menu_First
R_M_S	Second dish of the menu		Menu_Second
R_M_D	Dessert dish of the menu		Menu_Dessert
R_F_D	Dish assigned to the first dish		First_Dish
R_S_D	Dish assigned to the second dish		Second_Dish
R_D_D	Dish assigned to the dessert		Dish_Dessert
R_City_ Restaurant	Restaurants per city		City_ Restaurant
R_Dish_ DishDay	Dish assigned to dish of the day		Dish_ DishDay
R_S_K	All subtypes of a Kitchen style		Subtype_KitchenStyle
R_Restaurant_Dish	All dishes that a restaurant has		Restaurant_Dish
R_Service_Person	Scores of persons		Service_Person
R_Person_Restaurant	Restaurant’s Staff		Person_Restaurant

Table of Volumes

In this table i'm gonna write some examples of volumes, just to have an idea.

Name	Type	VOLUME
Restaurant	E	500
Offer	E	149400
Evaluation	E	1494000
Subtypes	E	42

Explanation of Volumes

Name	Operation
Restaurant	200 Restaurants in the database
Offer	200(rest)*249(Business days in Mallorca)*3(time zones)
Evaluation	373500(Number Offers)*10(approximate number of ratings)
Subtype	6(number continents)*7(average between subtypes of restaurant in mallorca)

Some continents like Asia or South America can have more than 12 subtypes against Australia, maximum 3-4, Antartica has no restaurants.

Table of Operations

We put 3 as an example.

Operation	Description	Frequency	Type
Op1	Insert of a new restaurant	20 per year	I
Op2	Check evaluation offers	1494000 per year	I
Op3	Check DishDay	4000 per week	I
Op4	Create the view with the restaurants that use Mallorquina Kitchen Style	80 per week	I
Op7	Create the view with the restaurants, from the highst to the lowest, calculating the average of their offers scores.	5000 per week	I

Tables of Accesses to Operations

Op1

The dish W or R depends if the dish is already created for another restaurant

Concept	Type	Access	Type
Restaurant	C	1	W
Subtype	C	1	R
Dish	C	1	R/W
City	C	1	R
R_Restaurant _Subtype	R	1	W
R_Restaurant_Dish	R	1	W
R_Restaurant_City	R	1	W

Op2

Minimum1494000 per year(1 access to every opinion) between all restaurants

Concept	Type	Access	Type
Evaluation	C	1494000	W
Offer	C	1494000	R
R_Evaluation_Offer	R	1494000	W

Op3
The clients check the daily dish of restaurants (week), we consider 1 check per dish day and per restaurant, 1check * 5 days a week * 200 restaurants = 1000 checks

Concept	Type	Access	Type
Restaurant	C	1000	R
Offer	C	1000	R
DishDay	C	1000	R
Dish	C	1000	R
R_Dish_DishDay	R	1000	R
R_Restaurant_Offer	R	1000	R

Op4
The average per week is about 80 checks per week

Concept	Type	Access	Type
Restaurant	C	80	R
City	C	80	R
Type Cooking	C	80	R
R_Restaurant_City	R	80	R
R_KitchenStyle_Restaurant	R	80	R

Op7
The average per week is about 5000 checks per week

Concept	Type	Access	Type
Restaurant	C	5000	R
Evaluation	C	5000	R
Offer	C	5000	R
R_Restaurant_Offer	R	5000	R
R_Evaluation_Offer	R	5000	R

Dictionary of External Constraints

First the requirement, then the translation to the database

Must enter the date on which the score was made, that has to be after the day of the offer.
Date of **Evaluation** >= of date of the **Offer**

The forks of a restaurant have to be between 1-5.
Forks of **Restaurant** >=1 AND Forks<=5

The score has to be between 1-5.
Score of **Evaluation** >=1 AND Forks<=5

This information has to be published 2 weeks before the offer.
Date of the **Offer** >= NOW()+2 weeks

Marks should be between 1-5.
Marks of **Service** >=1 AND Marks<=5

Restructuring of the Conceptual Scheme

ER Schema Changes

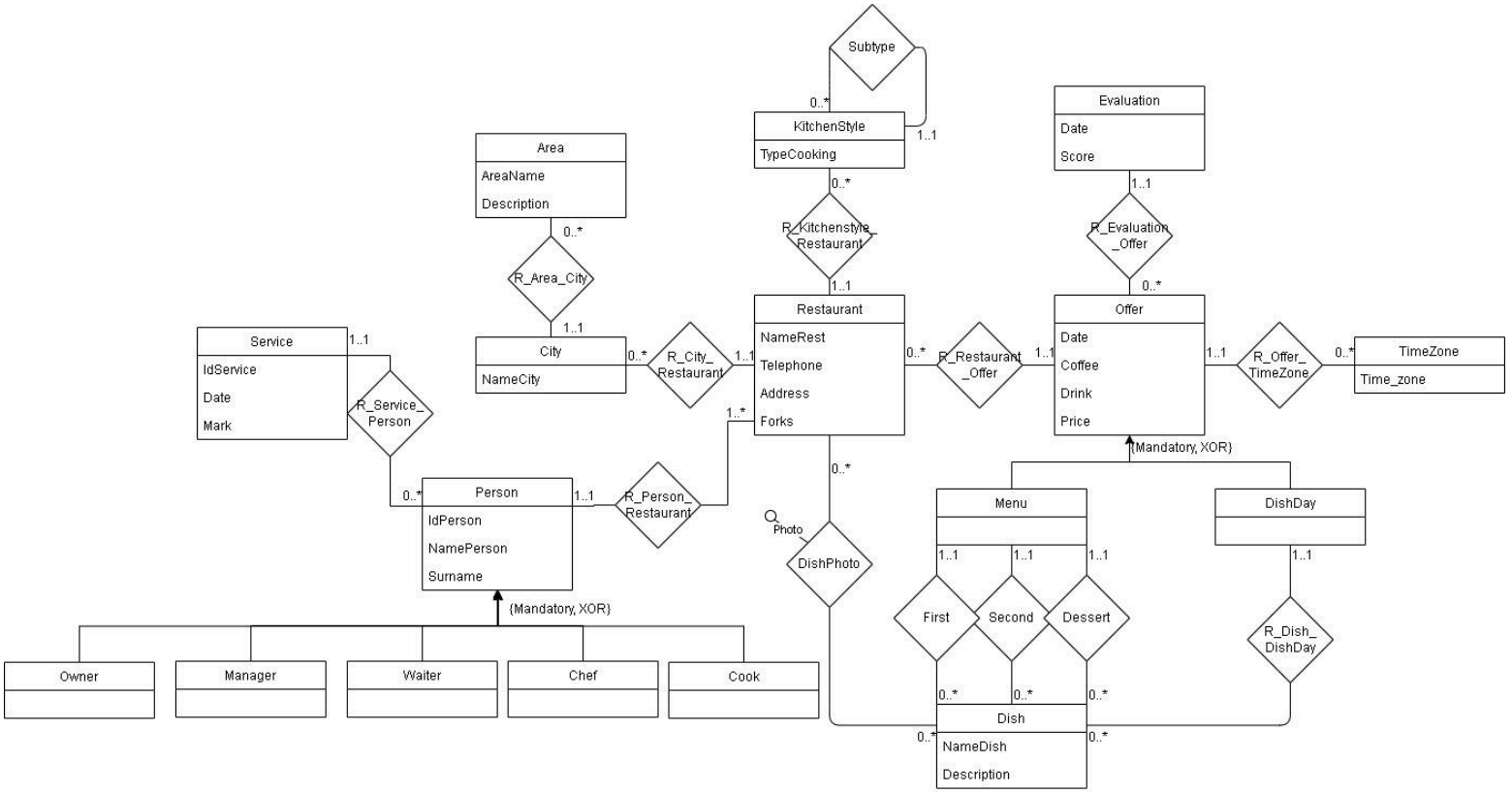
While analyzing the different operations and relations we realize this different changes

- If we keep the class **DishPhoto**, is going to provoke that all dishes from all restaurants have the same photo for common dishes, in order to allow every restaurant put their own pic, we are gonna **transform** the class **DishPhoto** into an attribute for the relation **R_Dish_DishPhoto.**

- We can avoid the excessive creation of classes for **Menu**, by **transform First**, **Second** and **Dessert** into a relation, it will make the same function and increase the efficiency by adding the NameDish as First, Second and Dessert as attributes of **Menu**.

- We can fuse **KitchenStyle** and **Subtype** into a general class and relation with itself to avoid creating another class, as both classes have their name as unique attribute.

Restructured ER Diagram



Data Dictionary (2nd Version)

Class

Class	Description	External links	Identificator
Area	Part of Mallorca's territory		NameArea
City	Large town		NameCity
KitchenStyle	Style of cooking		TypeCooking
Offer	Offer of a restaurant a specific day	This information has to be published 2 weeks before the offer.	IdOffer
Menu	Cart formed by the 3 dish of a full meal		IdMenu
Dish	A dish in gastronomy is a specific food preparation, ready to eat or to be served.		NameDish
DishDay	Unique dish that is offered apart of the menu		IdPdd
TimeZone	Differentiation of the moments for eating during the day		Time_Zone
Evaluation	Score that a client has given to an offer	-Must enter the date on which the score was made, that has to be after the day of the offer. -The score has to be between 1-5.	IdEvaluation
Restaurant	Establishment were you can buy and consume prepared food	The forks of a restaurant have to be between 1-5.	IdRestaurant
Service	Score that each customer gives to	The mark has to be between 1-5	IdService

	every Staff		
Waiter	Those attending to customers by supplying them with food and drink as requested		IdPerson
Chef	Is a trained professional cook and tradesman who is proficient in all aspects of food preparation		IdPerson
Manager	Managing waiters		IdPerson
Owner	Owner of the restaurant		IdPerson
Cook	Individuals who prepare food for consumption in restaurants.		IdPerson
Person	Individual who works in a restaurant		IdPerson

Relations

Relation	Description	External constraints	Classes associate
R_Area_City	Cities that an Area can have		Area_City
R_KitchenStyle_ Restaurant			Kitchenstyle_ Restaurant
R_Restaurant _Offer	Offer that a restaurant have for a specific day		Restaurant _Offer
R_Restaurant _Offer	All the evaluation to that specific offer		Restaurant _Offe
R_Offer_ TimeZone	The time zone of the offer		Offer_ TimeZone
R_Dish_ DishPhoto	Photo of the dish		Dish_ DishPhoto
First	First dish of the menu		Menu_Dish

Second	Second dish of the menu		Menu_Dish
Dessert	Dessert dish of the menu		Menu_Dish
R_City_ Restaurant	Restaurants per city		City_ Restaurant
R_Dish_ DishDay	Dish assigned to dish of the day		Dish_ DishDay
R_Subtype	All subtypes of a Kitchen style		KitchenStyle
R_Restaurant_DishPhoto_Dish	All dishes that a restaurant has + the attribute photo		Restaurant_Dish
R_Service_Person	Scores of persons		Service_Person
R_Person_Restaurant	Restaurant’s Staff		Person_Restaurant

Relations cardinalities

Restaurant 1..1 ← → 0..*City
There can be several restaurants in a City.
A restaurant is only located in one City.

Restaurant 1..1 ← → 0..* KitchenStyle
A restaurant can only have one Kitchen style..
A kitchen style can belong to many restaurants

Restaurant 0..* ← → 1..1 Offer
A restaurant can have many related offers.
An Offer can only be related to a specific restaurant.

Restaurant 0..* ← → 0..* Dish
A restaurant can offer various dishes.
A Dish can be offered at several restaurants..

City 1..1 ← → 0..* Area
A City Only belongs to a specific Geographic Area.
There may be several restaurants in an Area.

KitchenStyle 0..* ← → 1..1 KitchenStyle
This relationship is a reflective relationship. Relate a general style of cooking to a more specific Kitchen style. For example Asian ← → China.
A general kitchen style can have several cooking subtypes (eg Asian ← → Chinese, Japanese, etc.)
A kitchen subtype can only have a parent type of cooking style (e.g. Chinese ← → Asian)

Offer 0..* ← → 1..1 Valoració
An Offer can have several evaluations.
An evaluation is related to a specific Offer.

Offer 1..1 ← → 0..* TimeZone
An Offer has a specific Time zone.
A time zone may be related to several different offerings.

Dish (rFirst, rSecond, rDessert) 0..* ← → 1..1 Menu
In this case, three different relationships are described, one for each type of Dish included in the menu. In this way a menu is related to three Dishs: a First, a Second and a Dessert.
A Dish can be offered in several menus..
A menu can have only one dish for each relationship (First, Second, Dessert)

Dish 0..* ← → 1..1 DishDay
A Dish can often be a Dish of the day (it may also never be a Dish)
A Dish of the day is related to a particular Dish.

Inheritance: Offer ← → DishDay | {M,XOR} |Offer ← → Menú
An Offer can be a Dish of the Day or a Menu, which is why it is a mandatory XOR. Because Offer attributes are shared by menu and Dish of the day, a legacy relationship is considered the most appropriate option for good readability and understanding of the mode.

Inheritance: Person ← → (waiter, cook…etc| {M,XOR}
A person only can be a worker, mandatory, and a person only can have one function or work, XOR. As there are no relationship between “children” we will consider function, and classes will inheritance id from Person

Service 1..1 ← → 0..* Person
A person can have 0 or several.
A Service is designated to one person

Person 1..1 ← → 1..* Restaurant
A person can have various staff.
A Service is designated to one Restaurant

Dictionary of External Constraints (2nd Version)

First the requirement, then the translation to the database

Must enter the date on which the score was made, that has to be after the day of the offer.
Date of **Evaluation** >= of date of the **Offer**

The forks of a restaurant have to be between 1-5.
Forks of **Restaurant** >=1 AND Forks<=5

The score has to be between 1-5.
Score of **Evaluation** >=1 AND Forks<=5

This information has to be published 2 weeks before the offer.
Date of the **Offer** >= NOW()+2 weeks

Table of Volumes (2nd Version)

In this table i'm gonna write some examples of volumes, just to have an idea.

Name	Type	VOLUME
Restaurant	E	500
Offer	E	149400
Evaluation	E	1494000
Subtypes	E	42

Explanation of Volumes

Name	Operation
Restaurant	200 Restaurants in the database
Offer	200(rest)*249(Business days in Mallorca)*3(time zones)
Evaluation	373500(Number Offers)*10(approximate number of ratings)
Subtype	6(number continents)*7(average between subtypes of restaurant in mallorca)

Some continents like Asia or South America can have more than 12 subtypes against Australia, maximum 3-4, Antartica has no restaurants.

Table of Operations

We put 3 as an example and 2 from the queries below, this was used with the table before the final version of the tables.

Operation	Description	Frequency	Type
Op1	Insert of a new restaurant	20 per year	I
Op2	Check evaluation offers	1494000 per year	I
Op3	Check DishDay	4000 per week	I
Op4	Create the view with the restaurants that use Mallorquina Kitchen Style	80 per week	I
Op7	Create the view with the restaurants, from the highst to the lowest, calculating the average of their offers scores.	5000 per week	I

Table of Accesses to Operations

Op1
The dish W or R depends if the dish is already created for another restaurant

Concept	Type	Access	Type
Restaurant	C	1	W
Subtype	C	1	R
Dish	C	1	R/W
City	C	1	R
R_Restaurant _Subtype	R	1	W
R_Restaurant_Dish	R	1	W
R_Restaurant_City	R	1	W

Op2
Minimum 249000 (5 check opinion at the end of the day(5 days per week) * 200 every restaurant * 249 working days in a year)

Concept	Type	Access	Type
Evaluation	C	249000	R
Offer	C	249000	R
R_Evaluation_Offer	R	249000	R

Op3
The clients check the daily dish of restaurants (week), we consider 1 check per dish day and per restaurant, 1check * 5 days a week * 200 restaurants = 1000 checks

Concept	Type	Access	Type
Restaurant	C	1000	R
Offer	C	1000	R
DishDay	C	1000	R
Dish	C	1000	R
R_Dish_DishDay	R	1000	R
R_Restaurant_Offer	R	1000	R

Op4
The average per week is about 80 checks per week

Concept	Type	Access	Type
Restaurant	C	80	R
City	C	80	R
Type Cooking	C	80	R
R_Restaurant_City	R	80	R
R_KitchenStyle_Restaurant	R	80	R

Op4
The average per week is about 5000 checks per week

Concept	Type	Access	Type
Restaurant	C	5000	R
Evaluation	C	5000	R
Offer	C	5000	R
R_Restaurant_Offer	R	5000	R
R_Evaluation_Offer	R	5000	R

Direct translation to the Relational Model

Logical Diagram

Class

Area (**NameArea**, Description)

City (**NameCity**)

Restaurant (**IdRestaurant**, NameRest, Telephone, Address, Forks)

Dish (**NameDish**, Description)

Menu (**IdMenu**)

Offer (**IdOffer**, Coffee, Beguda, Date, Price)

DishDay (**IdPdd**)

Evaluation (**IdEvaluation** , Date, Score)

KitchenStyle (**TypeCooking**)

TimeZone(**Time_Zone**)

DishPhoto (**IdPhoto**, Photo)

Person (**IdPerson**, NamePerson, Surname)

Service(**IdService**, Date, Mark)

Owner (**IdPerson**)

Manager (**IdPerson**)

Waiter (**IdPerson**)

Chef (**IdPerson**)

Cook (**IdPerson**)

Relations

The criteria used for choosing the key is based on the cardinality of the relations.

R_Restaurant_DishPhoto_Dish (**IdRestaurant**, **NameDish**)

R_Area_City (NameArea, **NameCity**)

R_City_Restaurant (NameCity, **IdRestaurant**)

R_KitchenStyle_Restaurant (TypeCooking , **IdRestaurant**)

R_Subtype (OTipus_Cuina, **STipus_cuina**)

R_First (NameDish1, **IdMenu**)

R_Second (NameDish2, **IdMenu**)

R_Dessert (Dessert, **IdMenu**)

R_Restaurant_Offer (IdRestaurant, **IdOffer**)

R_Evaluation_Offer (IdOffer, **IdEvaluation**)

R_Offer_TimeZone**IdOffer**, Time_Zone)

R_Dish_DishDay(**IdPdd**, NameDish)

R_Service_Person(**IdService**, IdPerson)

R_Person_Restaurant (**IdPerson**, IdRestaurant)

Fusions

Now that we have the final class tables and relationships, we check if these tables can be merged. We painted the result of these fusions in green

- First look at the tables that cannot be merged with any, for example in this case Area has not been merged with any table, as the main key in the table of the relationship that affects Area is not NameArea. The same goes for TimeZone

Area (**NameArea**, Description)

TimeZone(**Time_Zone**)

- Now we will look at the tables that have been merged with another, we will put as an example City and R_Area_City have the same main key that could be merged, providing NameArea as a foreign key in the City table. As you can see in DishDay, Evaluation and Kitchen_style, the same process has been done.

City (**NameCity**)

R_Area_City (**NameCity**, NameArea)

Result:

City(**NameCity**, NameArea)

R_Dish_DishDay(**IdPdd**, NameDish)

Dish(**NameDish**, Descripció)

Result:

DishDay(**IdPdd**, NameDish)

Evaluation (**IdEvaluation**, Date, Score)

R_Evaluationn_Offer (IdOffer, **IdEvaluation**)

Result:

Evaluation (**IdEvaluation**, Date, Score, IdOffer)

KitchenStyle (**TypeCooking**)

R_Subtype (**TypeCooking**, SubtypeC)

Result:

KitchenStyle(**TypeCooking**, SubtypeC)

Person (**IdPerson**, NamePerson, Surname)

R_Person_Restaurant (**IdPerson**, IdRestaurant)

Result:

Person (**IdPerson**, NamePerson, Surname, IdRestaurant)

Service(**IdService**, Date, Mark)

R_Service_Person(**IdService**, IdPerson)

Result:

Service(**IdService**, Date, Mark, IdPerson)

- In these cases, the main key of a class table appears in two relationship tables, such as the Restaurant where its main key appears in two relationship tables. This causes two foreign keys (NameCity and TypeCooking) to appear on the Restaurant table after merging.

Restaurant (**IdRestaurant**, NameRest, Telephone, Address, Forks)

R_City_Restaurant (NameCity, **IdRestaurant**)

R_KitchenStyle_Restaurant (TypeCooking , **IdRestaurant**)

Result:

Restaurant (**IdRestaurant**, NameRest, Telephone, Address, Forks, NameCity, TypeCooking)

Offer(**IdOffer**, Coffee, Drink, Date, Price)

R_Offer_TIMEZONE(**IdOffer**, Time_Zone)

R_Restaurant_Offer (IdRestaurant, **IdOffer**)

Result:

Offer (**IdOffer**, Coffee, Beguda, Date, Price, IdRestaurant, Time_Zone)

-Photo is an attribute for the relationship R_Restaurant_DishPhoto_Dish

DishPhoto (**IdRestaurant**, **NameDish**, Photo)

- Finally, it is visible that the main menu key appears in 3 different relationships. This merge adds three foreign keys to the Menu table.
.

R_First (NameDish1, **IdMenu**)

R_Second (NameDish2, **IdMenu**)

R_Dessert (Dessert, **IdMenu**)

Menu (**IdMenu**)

Result:

Menu (**IdMenu**, NameDish1, NameDish2, Dessert)

Inheritance

1. A table for each of the entities and the primary key.

Offer (**IdOffer**, Coffee, Beguda, Date, Price, IdRestaurant, Time_Zone)

Menu (**IdMenu**, NameDish1, NameDish2, Dessert)

DishDay(**IdPdd**, NameDish)

- Note: Identifiers will be inherited from the Offer class.

2. Look at the possibilities of merging.

The Offer parent class can merge with all of its children, so Offer ceases to exist, and its daughter tables inherit all of its attributes and their respective relationships.

However, in this case we are not interested in applying the merger as it would complicate and increase the size of the database (the relationship with Restaurant would increase the size of the database), apart from having the Rating class with the identifiers without no control (a rating could be from a Menu and a DishDay). That said, the Offer parent class is related to the daughters, even if there is repeated data and therefore takes up more space in the database.

1. A table for each of the entities and the primary key.

Person (**IdPerson**, NamePerson, Surname, IdRestaurant)

Owner (**IdPerson**)

Manager (**IdPerson**)

Waiter (**IdPerson**)

Chef (**IdPerson**)

Cook (**IdPerson**)

- Note: Identifiers will be inherited from the Offer class.

2. Look at the possibilities of merging.

As it's an inheritance relationship of XOR, we can merge parent with children but we will conserve the original table as the relation with the class service will make it harder.

Person (**IdPerson**, NamePerson, Surname, IdRest)

Owner (**IdPerson**, NamePerson, Surname, IdRestaurant)

Manager (**IdPerson**, NamePerson, Surname, IdRestaurant)

Waiter (**IdPerson**, NamePerson, Surname, IdRestaurant)

Chef (**IdPerson**, NamePerson, Surname, IdRestaurant)

Cook (**IdPerson**, NamePerson, Surname, IdRestaurant)

Final tables

Format: FOREIGN KEY and **PRIMARY KEY**

Area (**NameArea**, Description)

TimeZone(**Time_Zone**)

City(**NameCity**, NameArea)

DishDay(**IdPdd**, NameDish)

Evaluation (**IdEvaluation**, Date, Score, IdOffer)

KitchenStyle(**TypeCooking**, SubtypeC)

Restaurant (**IdRestaurant**, NameRest, Telephone, Address, Forks, NameCity, TypeCooking)

Offer (**IdOffer**, Coffee, Drink, Date, Price, IdRest, Time_Zone)

DishPhoto (**IdRest**, **NameDish**, Photo)

Menu (**IdMenu**, NameDish1, NameDish2, Dessert)

Dish(**NameDish,** Description)

Person (**IdPerson**, NamePerson, Surname, IdRest)

Service(**IdService**, Date, Mark, IdPerson)

Owner (**IdPerson**, NamePerson, Surname, IdRestaurant)

Manager (**IdPerson**, NamePerson, Surname, IdRestaurant)

Waiter (**IdPerson**, NamePerson, Surname, IdRestaurant)

Chef (**IdPerson**, NamePerson, Surname, IdRestaurant)

Cook (**IdPerson**, NamePerson, Surname, IdRestaurant)

Database Specification in SQL

```
CREATE TABLE Area(
    NameArea char(20) PRIMARY KEY,
    Description char(150)
);
CREATE TABLE City(
    NameCity char(15) PRIMARY KEY,
    NameArea char(20) NOT NULL,
    FOREIGN KEY (NameArea) REFERENCES Area(NameArea)
);
CREATE TABLE KitchenStyle(
    TypeCooking char(15) PRIMARY KEY,
    SubtypeC  char(15),
    FOREIGN KEY (SubtypeC) REFERENCES KitchenStyle(TypeCooking)
);
```

```
CREATE TABLE Restaurant(
    IdRestaurant int PRIMARY KEY,
    NameRest char(15),
    Telephone int,
    Address char(30),
    Forks int,
    NameCity char(15) NOT NULL,
    TypeCooking char(15) NOT NULL,
    FOREIGN KEY (TypeCooking) REFERENCES KitchenStyle(TypeCooking),
    FOREIGN KEY (NameCity) REFERENCES City(NameCity)
);
CREATE TABLE TimeZone(
    Time_Zone char(15) PRIMARY KEY
);
CREATE TABLE Offer(
    IdOffer int PRIMARY KEY,
    Coffee boolean,
    Drink boolean,
    Date Date,
    Price float,
    IdRest int NOT NULL,
    Time_Zone char(15) NOT NULL,
    FOREIGN KEY (IdRest) REFERENCES Restaurant(IdRestaurant),
    FOREIGN KEY (Time_Zone) REFERENCES TimeZone(Time_Zone));
```

```
CREATE TABLE Evaluation(
    IdEvaluation int PRIMARY KEY,
    Date Date,
    Score int,
    IdOffer int NOT NULL,
    FOREIGN KEY (IdOffer) REFERENCES Offer(IdOffer)
);

CREATE TABLE Dish(
    NameDish char(50) PRIMARY KEY,
    Description char(150)
);

CREATE TABLE Menu(
    IdMenu int PRIMARY KEY,
    NameDish1 char(50) NOT NULL,
    NameDish2 char(50) NOT NULL,
    Dessert char(50) NOT NULL,
    FOREIGN KEY (NameDish1) REFERENCES Dish(NameDish),
    FOREIGN KEY (NameDish2) REFERENCES Dish(NameDish),
    FOREIGN KEY (Dessert) REFERENCES Dish(NameDish),
    FOREIGN KEY (IdMenu) REFERENCES Offer(IdOffer)
);
```

```
CREATE TABLE DishDay(
    IdPdd int PRIMARY KEY,
    NameDish char(50) NOT NULL,
    FOREIGN KEY (IdPdd) REFERENCES Offer(IdOffer),
    FOREIGN Key (NameDish) REFERENCES Dish(NameDish)
);



CREATE TABLE DishPhoto(
    IdRest int,
    NameDish char(50),
    Photo blob,
    PRIMARY KEY(IdRest,NameDish),
    FOREIGN KEY (IdRest) REFERENCES Restaurant(IdRestaurant),
    FOREIGN KEY (NameDish) REFERENCES Dish(NameDish)
);


CREATE TABLE Person(
    IdPerson int,
    NamePerson char(50),
    Surname char(50),
    IdRestaurant int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdRestaurant) REFERENCES Restaurant(IdRestaurant)
```

```
);
CREATE TABLE Service(
    IdService int,
    Date Date,
    Mark int,
    IdPerson int,
    PRIMARY KEY(IdService),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
CREATE TABLE Owner(
    IdPerson int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
CREATE TABLE Manager(
    IdPerson int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
CREATE TABLE Waiter(
    IdPerson int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
```

```
CREATE TABLE Chef(
    IdPerson int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
CREATE TABLE Cook(
    IdPerson int,
    PRIMARY KEY(IdPerson),
    FOREIGN KEY (IdPerson) REFERENCES Person(IdPerson)
);
```

Relational Scheme

As the relational scheme has been very optimized, I don't find any more modifications.

Triggers

- Must enter the date on which the score was made, that has to be after the day of the offer.

```
DELIMITER $$
CREATE TRIGGER CorrectDateEvaluation BEFORE INSERT ON Evaluation
 FOR EACH ROW
BEGIN
IF (NEW.Date < (SELECT Date FROM Offer Where Evaluation.IdOffer = Offer.IdOffer))
THEN
		SIGNAL SQLSTATE '45004' SET MESSAGE_TEXT = 'Evaluation have to be
after or same day as the offer'';
	END IF;
END$$
DELIMITER;
```

- The forks of a restaurant have to be between 1-5.

```
DELIMITER $$
CREATE TRIGGER CorrectF BEFORE INSERT ON Restaurant
 FOR EACH ROW
BEGIN
		IF (5 < NEW.Forks AND NEW.Forks < 1) THEN
		SIGNAL SQLSTATE '45001' SET MESSAGE_TEXT = 'Forks have to be
between 1-5';
	END IF;
END$$
DELIMITER;
```

- The score has to be between 1-5.

```
DELIMITER $$
CREATE TRIGGER CorrectScore BEFORE INSERT ON Evaluation
 FOR EACH ROW
BEGIN
IF (5 < NEW.Score AND NEW.Score < 1) THEN
		SIGNAL SQLSTATE '45002' SET MESSAGE_TEXT = 'Score have to be
between 1-5';
	END IF;
END$$
DELIMITER;
```

- This information has to be published 2 weeks before the offer.

```
DELIMITER $$
CREATE TRIGGER CorrectDate BEFORE INSERT ON Offer
 FOR EACH ROW
 BEGIN
 IF (DATE_ADD(NOW(), INTERVAL 2 WEEK) > NEW.Date) THEN
                SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Date of offer must be 2
or more from now';
        END IF;
END$$
DELIMITER;
```

- Service marks need to be between 1-5

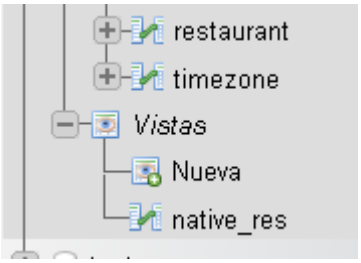
```
DELIMITER $$
CREATE TRIGGER CorrectMark BEFORE INSERT ON Service
 FOR EACH ROW
BEGIN
IF (5 < NEW.Mark AND NEW.Mark < 1) THEN
                SIGNAL SQLSTATE '45003' SET MESSAGE_TEXT = 'Mark have to be
between 1-5';
        END IF;
END$$
DELIMITER;
```

Queries on operations

Op4

Create the view with the restaurants that use Mallorquina Kitchen Style in Llevant area

```
CREATE VIEW Native_Res AS

SELECT Restaurant.NameRest, Restaurant.Telephone, Restaurant.Address, Restaurant.Forks

FROM City

JOIN Restaurant ON City.NameArea = 'Llevant'

AND Restaurant.TypeCooking = 'Mallorquina'

AND Restaurant.NameCity = City.NameCity;
```



Op 5

Calculate the total of the Japanese restaurants of an specific area, by the description of the area

```
SELECT COUNT(Restaurant.IdRestaurant) AS RestaurantsJapanese

FROM Restaurant

JOIN City ON Restaurant.TypeCooking = 'Japanese'

AND City.NameCity = Restaurant.NameCity

JOIN Area ON Area.Description = 'Llevant is a Mallorcan region that includes the east of the island of Mallorca'

AND Area.NameArea = City.NameArea;
```

RestaurantsJapanese
2

Op6

List of restaurants that offer in a specific date, a menu with ‘Arros Brut’ as a first, ‘Porcella’ as second and ‘Pijama’ as dessert

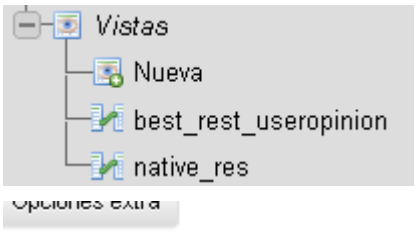
```
SELECT Restaurant.NameRest,Restaurant.Telephone,Restaurant.Address

FROM menu

JOIN Offer ON Offer.Date = '2024-03-02'

AND menu.NameDish1 = 'Arros Brut'

AND menu.NameDish2 = 'Porcella'

AND menu.Dessert = 'Pijama'

AND Offer.IdOffer = menu.IdMenu

JOIN Restaurant ON Restaurant.IdRestaurant = Offer.IdRest;
```

NameRest	Telephone	Address
Ca na Rosa	683728472	Carrer Vidriell 23
Langostas Pepe	629304768	Carrer Carajillo
ComidaEsComida	683721422	Carrer Margalida 72

Op7

Create the view with the restaurants, from the highst to the lowest, calculating the average of their offers scores.

```
CREATE VIEW Best_Rest_UserOpinion AS

SELECT Restaurant.NameRest, Restaurant.Telephone, Restaurant.Address,
Restaurant.NameCity,AVG(Evaluation.Score) AS AVGScore

FROM Evaluation

JOIN Offer ON Offer.IdOffer = Evaluation.IdOffer

JOIN Restaurant ON Restaurant.IdRestaurant = Offer.IdRest

GROUP BY Restaurant.NameRest

ORDER BY AVGScore DESC;
```



NameRest	Telephone	Address	NameCity	AVGScore
Can Patatilla	618733129	Carrer Bonnin, 22	Mancor	5.0000
Kebab Tomeu	627910002	Carrer Trèbol 44	Mancor	4.0000
Shinnosuke	681375638	Carrer Tomeu Sastre 23	Arta	3.0000
AntonioDiForno	718939261	Carrer Capmenut, 51	Palma	2.6667
Ca na Rosa	683728472	Carrer Vidriell 23	Arta	2.6000
Sugoku Umai	365355864	Carrer es Xino, 10	Arta	2.5000
Langostas Pepe	629304768	Carrer Carajillo	Mancor	2.5000
Ganges Flow	62716393	Carrer Turmeda 7	Llucmajor	2.5000
Il tagliatiano	512188799	Carrer Joanot, 65	Andratx	2.0000
DiverSo	637623219	Carrer Gil 45	Felanitx	2.0000
Top Kebab	678452392	Carrer Eusebi Estada 65	Palma	1.0000

Op8

Show list of the Restaurants that don't offer Paella as a first

```
SELECT DISTINCT Restaurant.IdRestaurant, Restaurant.NameRest, Restaurant.Telephone,
Restaurant.Address, Restaurant.NameCity, Restaurant.Forks FROM

Restaurant

LEFT JOIN (

        SELECT Offer.IdOffer, Offer.IdRest FROM

        menu

        JOIN Offer ON menu.NameDish1 = 'Paella'

        AND Offer.IdOffer = menu.IdMenu

) AS OfertesPrPaella

ON Restaurant.IdRestaurant = OfertesPrPaella.IdRest

WHERE OfertesPrPaella.IdOffer IS NULL
```

IdRestaurant	NameRest	Telephone	Address	NameCity	Forks
0	Bar Paco	634593827	Carrer Sargantana 13	Palma	1
1	Top Kebab	678452392	Carrer Eusebi Estada 65	Palma	2
2	DiverSo	637623219	Carrer Gil 45	Felanitx	5
3	Ca na Rosa	683728472	Carrer Vidriell 23	Arta	2
4	Langostas Pepe	629304768	Carrer Carajillo	Mancor	3
5	ComidaEsComida	683721422	Carrer Margalida 72	Lloseta	2
6	Shinnosuke	681375638	Carrer Tomeu Sastre 23	Arta	3
7	Ganges Flow	62716393	Carrer Turmeda 7	Llucmajor	4
9	31FAM	621328202	Carrer Elefant 2	Bunyola	5
10	Parrilla Patxi	637415202	Carrer Agón 32	Binissalem	5
11	Kebab Tomeu	627910002	Carrer Trèbol 44	Mancor	4
12	Can Patatilla	618733129	Carrer Bonnin, 22	Mancor	2
13	Llepadits	971352185	Carrer Pocapalla, 18	Son Servera	1
14	AntonioDiForno	718939261	Carrer Capmenut, 51	Palma	5
15	Sugoku Umai	365355864	Carrer es Xino, 10	Arta	2
16	Mallorqueta	598233516	Carrer Madrid, 69	Son Servera	3
17	Wan Tun Tri	661381275	Carrer Capmenut, 52	Palma	2
18	La Guarida	781631863	Carrer Jaume I, 1	Llucmajor	5
19	El raconet	196816515	Carrer Ratoli, 22	Sineu	2
20	Il tagliatiano	512188799	Carrer Joanot, 65	Andratx	2

Op9

Show the restaurants that don’t offer Paella at all, group by city.

```
SELECT DISTINCT Restaurant.IdRestaurant, Restaurant.NameRest, Restaurant.Telephone,
Restaurant.Address, Restaurant.NameCity, Restaurant.Forks

FROM Restaurant

LEFT JOIN(

        (

        SELECT Offer.IdOffer, Offer.IdRest, menu.IdMenu FROM

        menu

        JOIN Offer ON (

                menu.NameDish1 = 'Paella'

                OR menu.NameDish2 = 'Paella'

                OR menu.Dessert = 'Paella'

                )

        AND Offer.IdOffer = menu.IdMenu

        ) UNION (

        SELECT Offer.IdOffer, Offer.IdRest, DishDay.IdPdd FROM

        DishDay

        JOIN Offer ON DishDay.NameDish = 'Paella'

        AND Offer.IdOffer = DishDay.IdPdd

        )

)AS OffersWithPaella

ON Restaurant.IdRestaurant = OffersWithPaella.IdRest

WHERE OffersWithPaella.IdOffer IS NULL

ORDER BY (Restaurant.NameCity)
```

IdRestaurant	NameRest	Telephone	Address	NameCity	Forks
20	Il tagliatiano	512188799	Carrer Joanot, 65	Andratx	2
3	Ca na Rosa	683728472	Carrer Vidriell 23	Arta	2
6	Shinnosuke	681375638	Carrer Tomeu Sastre 23	Arta	3
10	Parrilla Patxi	637415202	Carrer Agón 32	Binissalem	5
9	31FAM	621328202	Carrer Elefant 2	Bunyola	5
2	DiverSo	637623219	Carrer Gil 45	Felanitx	5
5	ComidaEsComida	683721422	Carrer Margalida 72	Lloseta	2
7	Ganges Flow	62716393	Carrer Turmeda 7	Llucmajor	4
18	La Guarida	781631863	Carrer Jaume I, 1	Llucmajor	5
4	Langostas Pepe	629304768	Carrer Carajillo	Mancor	3
11	Kebab Tomeu	627910002	Carrer Trèbol 44	Mancor	4
12	Can Patatilla	618733129	Carrer Bonnin, 22	Mancor	2
0	Bar Paco	634593827	Carrer Sargantana 13	Palma	1
17	Wan Tun Tri	661381275	Carrer Capmenut, 52	Palma	2
19	El raconet	196816515	Carrer Ratoli, 22	Sineu	2
13	Llepadits	971352185	Carrer Pocapalla, 18	Son Servera	1
16	Mallorqueta	598233516	Carrer Madrid, 69	Son Servera	3

Op10

Show restaurants, and kitchen style that an a specific date, they offer the same dish in menu and dish of the day

```
SELECT Restaurant.NameRest, Restaurant.TypeCooking

FROM(

        (

        SELECT Offer.*, menu.NameDish1, menu.NameDish2, menu.Dessert FROM menu

        JOIN Offer ON Offer.IdOffer = menu.IdMenu

        WHERE Offer.Date = '2024-03-02'

        ) AS OfertesMenu

        JOIN(
```

```
        SELECT Offer.*,DishDay.NameDish from DishDay

        JOIN Offer ON Offer.IdOffer = DishDay.IdPdd

        WHERE Offer.Date = '2024-03-02'

        ) AS OfertesPdd

        ON (

        OfertesMenu.NameDish1 = OfertesPdd.NameDish

        OR OfertesMenu.NameDish2 = OfertesPdd.NameDish

        OR OfertesMenu.Dessert = OfertesPdd.NameDish

        )

        AND OfertesMenu.IdRest = OfertesPdd.IdRest

        JOIN Restaurant ON Restaurant.IdRestaurant = OfertesMenu.IdRest

)
```

NameRest	TypeCooking
Langostas Pepe	Castilian

Op11

Best afgan restaurant in Palma by the evaluations.

```
SELECT  Rest_Afgan_Palma.NameRest, Scorens_Restaurants
FROM (
        SELECT Restaurant.NameRest, Restaurant.IdRestaurant
        FROM Restaurant
        WHERE Restaurant.TypeCooking = 'Afgan' AND Restaurant.NameCity = 'Palma'
) AS Rest_Afgan_Palma
JOIN (
        SELECT Offer_Amb_Score.IdRest, AVG(Score) AS Scorens_Restaurants
        FROM
        (
        SELECT Offer.IdRest, AVG(Evaluation.Score) AS Score
        FROM Evaluation
        JOIN Offer ON Offer.IdOffer = Evaluation.IdOffer
        GROUP BY (Evaluation.IdOffer)
        ) AS Offer_Amb_Score

        GROUP By (Offer_Amb_Score.IdRest)
) AS AVG_Offer_Amb_Score
ON Rest_Afgan_Palma.IdRestaurant = AVG_Offer_Amb_Score.IdRest
ORDER BY (Scorens_Restaurants) DESC
LIMIT 1
```

NameRest	Scorens_Restaurants
AntonioDiForno	2.75000000

Op 12

For night time zone, and a specific date, show all the restaurants of Mallorcan style with 3 forks from Binissalem, with their dish day menu and the menu.

```
SELECT Restaurant.IdRestaurant, Restaurant.NameRest,
Restaurant.NameCity,Restaurant.Forks,Restaurant.TypeCooking, menu.*,DishDay.*

FROM Restaurant

JOIN Offer ON Restaurant.Forks > 3

AND Restaurant.NameCity = 'Binissalem'

AND Offer.Time_Zone = 'Night'

AND Restaurant.TypeCooking = 'Mallorquina'

AND Offer.Date = '2024-10-08'

AND Restaurant.IdRestaurant = Offer.IdRest

LEFT JOIN DishDay ON DishDay.IdPdd = Offer.IdOffer

LEFT JOIN menu ON menu.IdMenu = Offer.IdOffer
```

IdRestaurant	NameRest	NameCity	Forks	TypeCooking	IdMenu	NameDish1	NameDish2	Dessert	IdPdd	NameDish
10	Parrilla Patxi	Binissalem	5	Mallorquina	NULL	NULL	NULL	NULL	19	Chinese rice
10	Parrilla Patxi	Binissalem	5	Mallorquina	41	Sushi	Porcella	Orange Cake	NULL	NULL

Op13

Asiatic restaurants by area

```
CREATE VIEW Q_Rest_byArea AS

SELECT COUNT(Restaurant.IdRestaurant) AS Quantitity,Restaurant.Forks,City.NameArea

FROM Restaurant

JOIN City ON Restaurant.NameCity=City.NameCity

JOIN KitchenStyle ON (KitchenStyle.TypeCooking='Asian' OR KitchenStyle.SubtypeC='Asian')

AND Restaurant.TypeCooking=KitchenStyle.TypeCooking

GROUP BY Restaurant.Forks,City.NameArea

ORDER BY City.NameArea, Restaurant.Forks
```

Quantity	Forks	NameArea ▲ 1
1	2	Es Raiguer
1	2	Llevant
1	3	Llevant
1	4	Migjorn
2	2	Palma
1	5	Palma
1	2	Pla de Mallorca

Op14

Restaurants that only have five score in their offers.

```
SELECT NameRest
FROM restaurant
WHERE (EXISTS (SELECT *
                FROM offer
                WHERE (offer.IdOffer = ANY (SELECT evaluation.IdOffer
                                                FROM evaluation
                                                WHERE Score = 5)
                    ) AND (offer.IdRest = restaurant.IdRestaurant)
                )

        );
```

	NameRest
Borrar	Can Patatilla