You have given a case study that consists of two divert sets of the domain.

Part 1: About Sales and Delivery

Part 2: About Restaurant.

**Part 1 – Sales and Delivery:**

**Here you can find the schema file below:**

****

Composite data of a business organization, confined to the ‘sales and delivery’ domain is given for the period of the last decade. From the given data retrieve solutions for the given scenario.

List of tables:

Cust\_dimen: customer details

Market fact : details about sales , discount, profit and profit base margin with all other table ids

**Orders\_Dimen : order details**

**Prod\_Dimen: product details**

**Shipping Dimen: shipping details**

**Cust\_dimen:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Comments |
| Customer\_name | Varchar | Name of the customer |
| Province | Varchar | ID for Province |
| Region | Varchar | ID for Region |
| Customer\_Segment | Varchar | Types of the customer segments |
| Cust\_id | Varchar | Id to the customers |

**Market Fact:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Types | Comments |
| Ord id | Varchar | Id for the Order |
| Prod id | Varchar | Id for the product |
| Ship id | Varchar | ID for the shipping |
| Cust id | Varchar | ID for the customer |
| Sales | Float | The sales price for the product |
| Discount | Float | Discount for the product |
| Order Quantity | Float | Number of products have been ordered |
| Profit | Float | Profit that has been gained from the product |
| Shipping cost | Float | Shipping cost for the product |
| Product Base Margin | Float | Base margin value for the product |

**Orders\_Dimen:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Order id | integer | Id for the order |
| Order Date | date | The order date for that order has been ordered |
| Order Priority | varchar | Priority for the orders |
| Ord id | varchar | Order id as a varchar |

**Prod\_Dimen:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Product\_Category | Varchar | Type of the product |
| Product\_Sub\_category | Varchar | Name of the sub-category |
| Prod\_id | Varchar | Id of Product |

**Shipping Dimen:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| Order\_ID |  | Id for the orders |
| Ship\_Mode |  | Type of the shipping |
| Ship\_Date |  | Shipping date |
| Ship\_ID | Varchar | ID for the shipping |

**Question 1:** Find the top 3 customers who have the maximum number of orders

**Question 2:** Create a new column DaysTakenForDelivery that contains the date difference between Order\_Date and Ship\_Date.

**Question 3:** Find the customer whose order took the maximum time to get delivered.

**Question 4:** Retrieve total sales made by each product from the data (use Windows function)

**Question 5:** Retrieve the total profit made from each product from the data (use windows function)

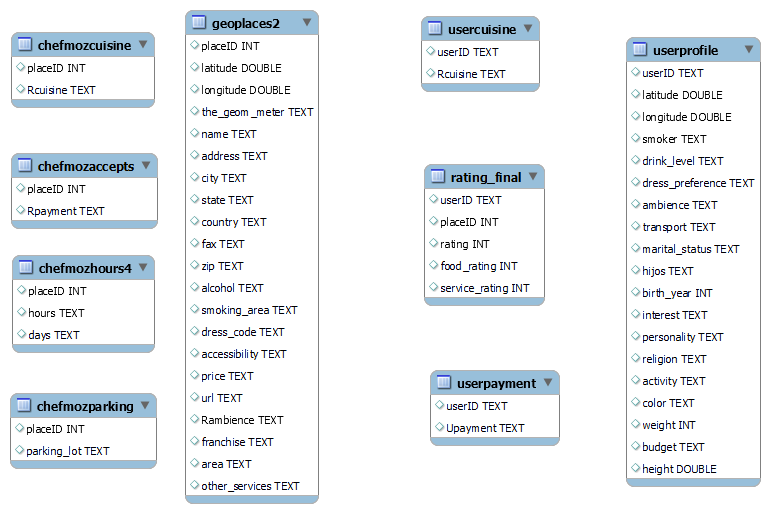
**Question 6:** Count the total number of unique customers in January and how many of them came back every month over the entire year in 2011

**Part 2 – Restaurant:**

# Above is the schema of the Restaurant Dataset.

**Overview of Restaurant Dataset**

* Chefmozaccepts (Location Wise availability of Payment Modes)
* Chefmozcuisine (Location Wise availability of Cuisine)
* Chefmozhours4(Working Hours of Restaurant)
* Chefmozparking (Parking availability at restaurants at different places)
* Geoplaces2(Location Wise Summary of dress code, country, state, etc.)
* Rating\_final (User wise rating to the restaurants in diff locations)
* Usercuisine (User had which Cuisine)
* User payment (User used which payment mode)
* Userprofile (Users personal details like a smoker, drink level, interest, religion, etc.)

**ER Diagram: **

If could see the ER diagram has no relationship representation map between the tables but we can join the relationship between the tables with ease using the basic domain understanding. I encourage you to set the relationship between the tables using a primary and foreign key if you feel it's needed.

**List of Tables:**

**geoplaces2:**

|  |  |  |
| --- | --- | --- |
| Column name: | Data Type | Describe |
| Place id | Integer | ID for place of the restaurants |
| Latitude | real | Latitude measurement values of the place |
| Longitude | real | Longitude measurement values of the place |
| the\_geom\_meter | text | The geo meter letters |
| Name | Varchar | Restaurants name |
| Address | Text | Address of place (restaurants) |
| City | Varchar | Name of the city |
| State | Varchar | Name of the State |
| Country | Varchar | Name of the Country |
| Fax | varchar | No data inside it. |
| Zip | integer | Zip code number |
| Alcohol | Varchar | Whether alcohol is served or not and what type of alcohol served |
| Smoking\_area | varchar | Whether smoking allowed or not and where they can smoke. |
| Dress\_code | Varchar | Type of dress |
| Accessibility | Varchar | Tells the accessibility whether have complete access or partially etc |
| Price | Varchar | Tells price medium, Low and High. |
| URL | Varchar | USR of the place restaurant |
| Rambience | Varchar | What type of ambiance |
| franchise | varchar | Type of franchise |
| Area | Varchar | Type of area whether it is closed or open |
| Other\_Services | varchar | Other services provided by the restaurant |

**Chefmozaccepts:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Place ID | Integer | Place id for the place |
| payment | Varchar | Type of the payments |

**Chefmozcuisine:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Place ID | Integer | Place id for the place |
| Rcuisine | Varchar | Type of the cuisine |

**Userprofile:**

|  |  |  |
| --- | --- | --- |
| Column name: | Data Type | Describe |
| User ID | Integer | ID for User |
| Latitude | real | Latitude measurement values of the place of the user |
| Longitude | real | Longitude measurement values of the place of the user |
| Smoker | Varchar | Boolean type True or False |
| Drive Level | Varchar | Type of the drinking condition |
| dress\_preference | Varchar | Type of dress preference |
| ambience | Varchar | Type of ambience |
| transport | Varchar | Type of transport the user is using |
| marital\_status | Varchar | Type of Martial Status |
| hijos | varchar | Type of Hijos like independent or kids |
| birth\_year | integer | Year of the birth |
| interest | Varchar | User is interested |
| personality | varchar | Type of personality of the user |
| religion | Varchar | Type of user religion |
| activity | Varchar | User activity what he does |
| color | Varchar | User interest color |
| Weight | Integer | Weight of the user |
| Budget | Varchar | Type Budget medium, Low and High. |
| Height | Integer | Height of the user in meters |

**chefmozhours4:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Place ID | Integer | Place id for the place |
| Hours | Time | Time of the working in a day |
| Days | Text | Working days |

**Chefmozparking:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| Place ID | Integer | Place id for the place |
| Parking lot | Varchar | Describes the type of the parking |

**rating\_final:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| User ID | Varchar | Id for the user |
| Place ID | Integer | Place id for the place |
| Rating | integer | Rating given by the user |
| Food\_Rating | Integer | Food rating given by the user |
| Service\_Rating | integer | Service rating given by the user |

**Usercuisine:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| User ID | Varchar | User id for the user |
| Rcuisine | Varchar | Type of the cuisines |

**Userpayment:**

|  |  |  |
| --- | --- | --- |
| Column Name | Data type | Description |
| User ID | Varchar | User id for the user |
| Upayment | Varchar | Type of the payment |

**Below mentioned are a few questions based on performances of different restaurants, based on different options.**

# Questions:

**Question 1: -** We need to find out the total visits to all restaurants under all alcohol categories available.

**Question 2: -**Let's find out the average rating according to alcohol and price so that we can understand the rating in respective price categories as well.

**Question 3:**  Let’s write a query to quantify that what are the parking availability as well in different alcohol categories along with the total number of restaurants.

**Question 4: -**Also take out the percentage of different cuisine in each alcohol type.

**Let us now look at a different prospect of the data to check state-wise rating.**

**Questions 5: -** let’s take out the average rating of each state.

**Questions 6: -**' Tamaulipas' Is the lowest average rated state. Quantify the reason why it is the lowest rated by providing the summary on the basis of State, alcohol, and Cuisine.

**Question 7: -** Find the average weight, food rating, and service rating of the customers who have visited KFC and tried Mexican or Italian types of cuisine, and also their budget level is low.

We encourage you to give it a try by not using joins.

**Part 3: Triggers**

**Question 1:**

Create two called Student\_details and Student\_details\_backup.

|  |  |
| --- | --- |
| Table 1: Attributes | Table 2: Attributes |
| Student id, Student name, mail id, mobile no. | Student id, student name, mail id, mobile no. |

You have the above two tables Students Details and Student Details Backup. Insert some records into Student details.

**Problem:**

Let’s say you are studying SQL for two weeks. In your institute, there is an employee who has been maintaining the student’s details and Student Details Backup tables. He / She is deleting the records from the Student details after the students completed the course and keeping the backup in the student details backup table by inserting the records every time. You are noticing this daily and now you want to help him/her by not inserting the records for backup purpose when he/she delete the records.write a trigger that should be capable enough to insert the student details in the backup table whenever the employee deletes records from the student details table.

Note: Your query should insert the rows in the backup table before deleting the records from student details.