

1. Gross Generated Revenue by Campaign/Store/Customer/Membership Type:

- **Fact Table:** Order Fact Table
(with measures Total = total+(tax_total_1+tax_total_2)-(discount_total_1+discount_total_2))
- **Dimensions:** Time, Customer, Membership_Type, Campaign, Store
- SQL for Gross Revenue:

```
SELECT SUM(total) AS total_gross_revenue, campaign_id, store_id, customer_id,
membership_type_id
FROM orders_fact
GROUP BY campaign_id, store_id, customer_id, membership_type_id;
```

2. How Long Until the First Customer Orders:

- **Fact Table:** Order Fact Table
(with measures first_order_date which when new order registered, calculate customer.created_date-order.created_date)
- **Dimensions:** customer, order_list, order
- SQL for First Order Time:

```
SELECT customer_id, AVG(since_created_to_order) AS first_order_date
FROM orders_fact
GROUP BY customer_id;
```

3. Most/Least Popular Items:

- **Fact Table:** Order Item Fact Table
(with measures item_quantity = sum(item_id) AS quantity group by (order_id))
- **Dimensions:** order, items, order_list
- SQL for Most Popular Items:

```
SELECT item_id, SUM(popularity) AS popularity
FROM orders_fact
GROUP BY item_id
ORDER BY popularity DESC (ASC)
LIMIT 1;
```

4. Gross per Sale:

- **Fact Table:** Order Fact Table (with measures such as order amount)
- **Dimensions:** order, order_list
- SQL for Gross per Sale:

```
SELECT count(gross_per_sale), order_id,  
FROM fact_orders  
GROUP BY order_id;
```

5. Items to Remove Based on Customer Preferences:

- **Fact Table:** Order Fact Table (with measures such as order processing time, delivery time)
- **Dimensions:** customer_preference, customer, preference, order, order_list
- SQL for Order Processing Time:

```
SELECT DISTINCT item_id  
FROM orders_fact  
WHERE preferences_value = 'Remove';
```

6. Improving Targeting of Marketing Campaigns:

- .

7. Order Processing/Delivery Time:

- **Fact Table:** Order Fact Table (with measures such as order processing time, delivery time)
- **Dimensions:** order, order_status
- SQL for Order Processing Time:

```
SELECT order_id, delivery_time_avg  
FROM fact_orders  
WHERE order_status_name = 'delivered';
```

8. Time for Customers to Add Their First Preference:

- **Fact Table:** Order Fact Table (with measures such as time to add first preference)
- **Dimensions:** customer, customer_preference

- SQL for Time to First Preference:

```
SELECT customer_id, since_created_till_add_preference
FROM fact_orders;
```

9. Time for Customers to Improve Their Membership:

- **Fact Table:** Order Fact Table (with measures such as time to improve membership)
- **Dimensions:** customer, membership_type
- SQL for Time to Membership Improvement:

```
SELECT customer_id, since_till_membership_improvement
FROM fact_orders;
```

10. Number of Campaigns to Improve a Customer's Membership:

- Requires analyzing the historical data on campaigns and their impact on membership improvements. No direct SQL command.

11. Store with Most Campaign Action:

- **Fact Table:** Campaign Fact Table (with measures such as campaign interactions)
- **Dimensions:** store, campaign
- SQL for Campaign Actions by Store:

```
SELECT store_id, COUNT(campaign_id) AS campaign_actions
FROM campaign_fact
GROUP BY store_id
ORDER BY campaign_actions DESC;
```

12. Creating New "Menu Meals" to Improve Sales:

- This can be done in different ways; like considering preferences and add most common to menu, or analyse which campaign have most profit and see which combination was most popular and add it to menu as default.

13. Measuring the Value of Membership Types for Returning Members:

- Yes and No;