

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
- 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:** Item
- SQL for Most Popular Items:

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

4. Gross per Sale:

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

```
SELECT order_id, order_amount  
FROM fact_orders;
```

5. Items to Remove Based on Customer Preferences:

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

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:** Time
- SQL for Order Processing Time:

```
SELECT order_id, (delivery_timestamp - order_timestamp) AS processing_time  
FROM fact_orders;
```

8. Time for Customers to Add Their First Preference:

- **Fact Table:** Customer Fact Table (with measures such as time to add first preference)
- **Dimensions:** Customer
- SQL for Time to First Preference:

```
SELECT customer_id, (preference_timestamp - first_order_timestamp) AS  
time_to_first_preference  
FROM fact_orders;
```

9. Time for Customers to Improve Their Membership:

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

```
SELECT customer_id, (membership_improved_timestamp - first_order_timestamp) AS  
time_to_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
- 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 various options; like considering preferences and addit to menu, or analyse which campaign have most profit.

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

- **Yes and No;**