

TESTING QUERIES

List and compare food and drink types

```
SELECT 'Food' AS ItemType, name AS ItemName, type AS Type, price
FROM food
UNION
SELECT 'Drink' AS ItemType, description AS ItemName, type AS Type, price
FROM drink
ORDER BY ItemType, Type;
```

Monthly Expenditure on Food and Drink for Each Member

```
SELECT
    m.memberid,
    m.campusaddress,
    EXTRACT(MONTH FROM c.transactiondate) AS Month,
    EXTRACT(YEAR FROM c.transactiondate) AS Year,
    SUM(CAST(SUBSTRING(f.price FROM 2) AS NUMERIC) * c.foodquantity) AS TotalFoodPrice,
    SUM(CAST(SUBSTRING(d.price FROM 2) AS NUMERIC) * c.drinkquantity) AS TotalDrinkPrice
FROM
    member m
JOIN
    card c ON m.memberid = c.memberid
LEFT JOIN
    food f ON c.foodid = f.foodid
LEFT JOIN
    drink d ON c.drinkid = d.drinkid
GROUP BY
    m.memberid, m.campusaddress, Month, Year
ORDER BY
    Year, Month, m.memberid;
```

List foods and their allergen status

```
SELECT f.name AS FoodName, f.type AS FoodType, f.price, a.description AS AllergenDescription
FROM food f
LEFT JOIN allergen_status a ON f.allergenid = a.allergenid
ORDER BY f.name;
```

Frequency of Alcoholic vs Non-Alcoholic Drink Purchases

```
SELECT d.isalcoholic, COUNT(*) AS PurchaseCount
FROM card c
JOIN drink d ON c.drinkid = d.drinkid
GROUP BY d.isalcoholic;
```

List members with their latest purchase dates

```
SELECT
    m.memberid,
    m.campusadress,
    MAX(c.transactiondate) AS LastPurchaseDate
FROM
    member m
LEFT JOIN
    card c ON m.memberid = c.memberid
GROUP BY
    m.memberid;
```

Members who spent above average

```
SELECT
    m.memberid,
    m.campusadress,
    SUM(CAST(SUBSTRING(f.price FROM 2) AS NUMERIC) * c.foodquantity + CAST(SUBSTRING(d.price FROM 2) AS NUMERIC) *
    c.drinkquantity) AS TotalSpent
FROM
    member m
JOIN
    card c ON m.memberid = c.memberid
LEFT JOIN
    food f ON c.foodid = f.foodid
LEFT JOIN
    drink d ON c.drinkid = d.drinkid
GROUP BY
    m.memberid
HAVING
    SUM(CAST(SUBSTRING(f.price FROM 2) AS NUMERIC) * c.foodquantity + CAST(SUBSTRING(d.price FROM 2) AS NUMERIC) *
    c.drinkquantity) > (
        SELECT AVG(TotalSpent) FROM (
            SELECT
                SUM(CAST(SUBSTRING(f.price FROM 2) AS NUMERIC) * c.foodquantity + CAST(SUBSTRING(d.price FROM 2) AS
                NUMERIC) * c.drinkquantity) AS TotalSpent
            FROM
                card c
            LEFT JOIN
                food f ON c.foodid = f.foodid
            LEFT JOIN
                drink d ON c.drinkid = d.drinkid
            GROUP BY
                c.memberid
        ) AS AverageSpending
    );
```

Members ranked by spending

```

SELECT
    m.memberid,
    m.campusadress,
    RANK() OVER (ORDER BY TotalSpent DESC) AS SpendingRank,
    TotalSpent
FROM (
    SELECT
        c.memberid,
        SUM(CAST(SUBSTRING(f.price FROM 2) AS NUMERIC) * c.foodquantity + CAST(SUBSTRING(d.price FROM 2) AS NUMERIC) *
c.drinkquantity) AS TotalSpent
    FROM
        card c
    LEFT JOIN
        food f ON c.foodid = f.foodid
    LEFT JOIN
        drink d ON c.drinkid = d.drinkid
    GROUP BY
        c.memberid
) AS Spending
JOIN
    member m ON Spending.memberid = m.memberid;

```