Молдавский Государственный Университет

Факультет Математики и Информатики

Департамент Информатики

**Дисциплина:** JavaScript

**Отчёт**

**по лабораторной работе №2**

**Тема:** Основы работы с массивами, функциями и объектами в JavaScript

Выполнил: студент группы IA2403

Демченко Юрий

Проверил: преподаватель

A.Murineț

Кишинев, 2025

**Лабораторная работа №2**

**Цель работы:**

Изучить основы работы с массивами и функциями в JavaScript, применяя их для обработки и анализа транзакций.

**Условия работы:**

Создайте консольное приложение для анализа транзакций.

Шаг 1. Создание массива транзакций

Шаг 2. Реализация функций для анализа транзакций

Шаг 3. Тестирование функций

Документирование кода

**Ход работы:**

Я создал файл main.js и добавил массив объектов с транзакциями. Каждая транзакция содержит свойства: transaction\_id, transaction\_date, transaction\_amount, transaction\_type, transaction\_description, merchant\_name, card\_type.

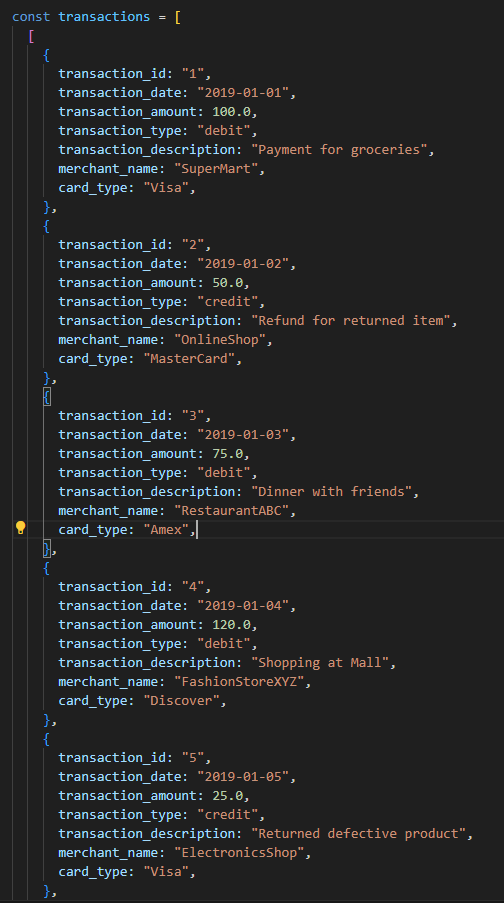
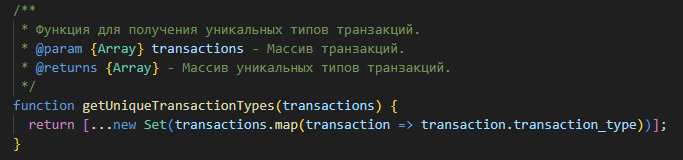


Рисунок 1 – Массив объектов с транзакциями

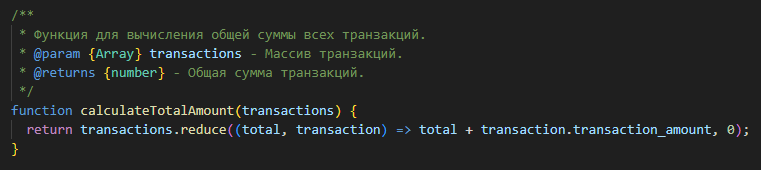
Далее я реализовал следующие функции для анализа транзакций:

1.getUniqueTransactionTypes(transactions)

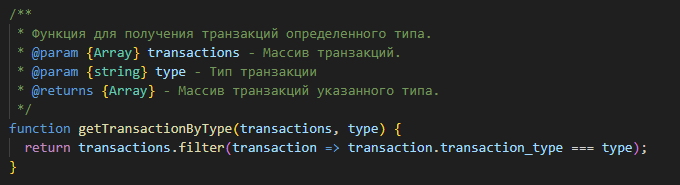
Возвращает массив уникальных типов транзакций. Я использовал Set() для выполнения задания.



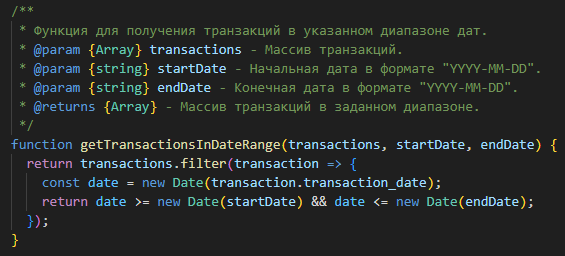
2.calculateTotalAmount(transactions) – Вычисляет сумму всех транзакций.



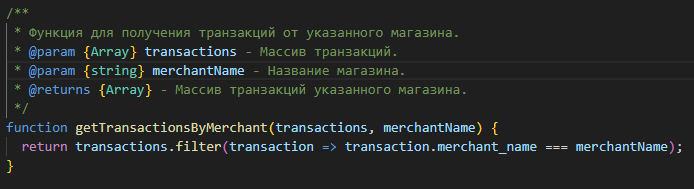
3.getTransactionByType(transactions, type) - Возвращает транзакции указанного типа (debit или credit).



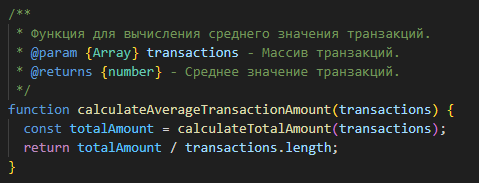
4.getTransactionsInDateRange(transactions, startDate, endDate) – Возвращает массив транзакций, проведенных в указанном диапазоне дат от startDate до endDate.



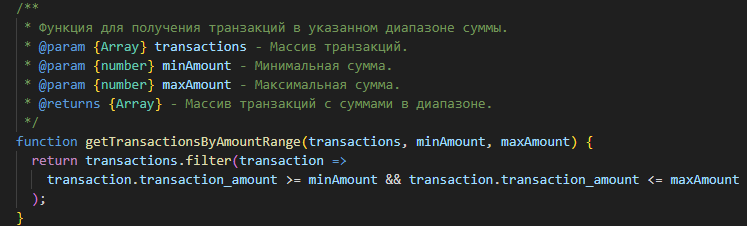
5.getTransactionsByMerchant(transactions, merchantName) – Возвращает массив транзакций, совершенных с указанным merchantName.



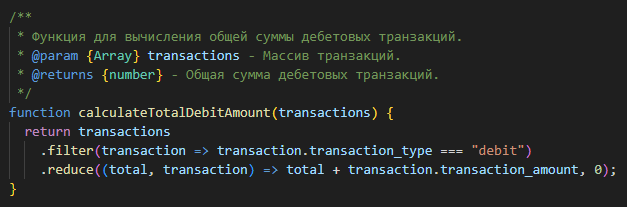
6.calculateAverageTransactionAmount(transactions) – Возвращает среднее значение транзакций.



7.getTransactionsByAmountRange(transactions, minAmount, maxAmount) – Возвращает массив транзакций с суммой в заданном диапазоне от minAmount до maxAmount.

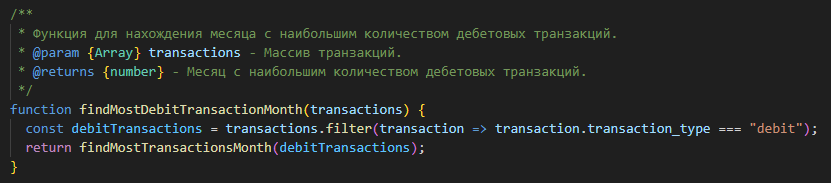


8.calculateTotalDebitAmount(transactions) – Вычисляет общую сумму дебетовых транзакций.



9.findMostTransactionsMonth(transactions) – Возвращает месяц, в котором было больше всего транзакций.

10.findMostDebitTransactionMonth(transactions) – Возвращает месяц, в котором было больше дебетовых транзакций.



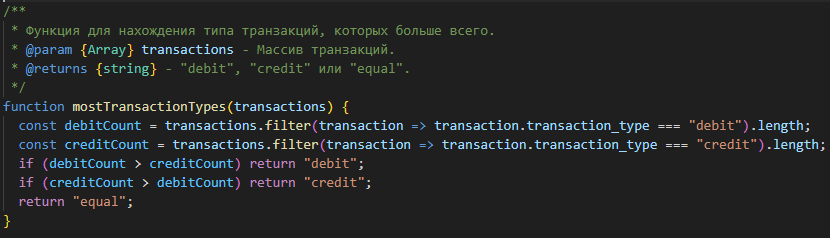
11.mostTransactionTypes(transactions)

Возвращает каких транзакций больше всего.

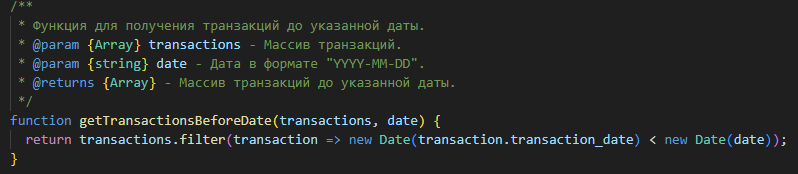
Возвращает debit, если дебетовых.

Возвращает credit, если кредитовых.

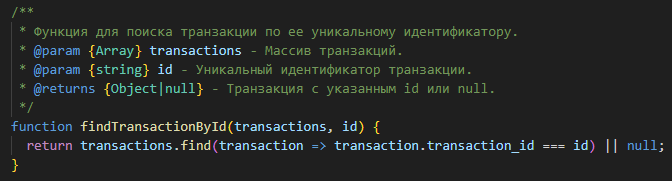
Возвращает equal, если количество равно.



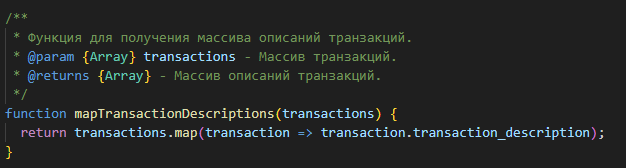
12.getTransactionsBeforeDate(transactions, date) – Возвращает массив транзакций, совершенных до указанной даты.



13.findTransactionById(transactions, id) – Возвращает транзакцию по ее уникальному идентификатору (id).



14.mapTransactionDescriptions(transactions) – Возвращает новый массив, содержащий только описания транзакций.



**Выводы**

В ходе выполнения лабораторной работы №2, я изучил основы работы с массивами и функциями в JavaScript, применяя их для обработки и анализа транзакций.

**Ответы на контрольные вопросы**

1. В JavaScript для обработки массивов объектов можно использовать map(), filter(), reduce(), find(), sort(), forEach().
2. В JavaScript даты, записанные в строковом формате ("YYYY-MM-DD"), можно сравнивать, преобразуя их в объект Date. Сравнивать даты можно с помощью методов, таких как getTime() или напрямую с использованием объектов Date.
3. Методы map(), filter() и reduce() выполняют разные операции с массивами. map() создает новый массив, преобразуя каждый элемент исходного массива. filter() возвращает новый массив, включающий только те элементы, которые удовлетворяют условию. reduce() сводит массив к одному значению, например, суммируя элементы или создавая объект.

**Библиография**

1. <https://github.com/MSU-Courses/javascript>
2. <https://moodle.usm.md/course/JavaScript>