**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный университет**

**Факультет прикладной математики-процессов управления**

**Кафедра “фундаментальная информатика и информационные технологии”**

**отчет**

**по лабораторной работе №1**

**по дисциплине «Алгоритмы и структуры данных»**

**на тему «генерация синтетических данных»**

**Вариант – 1**

|  |  |  |
| --- | --- | --- |
| Студент гр. 23Б16-пу |  | Сулимов А.С. |
| Преподаватель |  | Дик А.Г. |

**Санкт-Петербург**

**2024 г**

Оглавление

[Цель работы 3](#_Toc178589189)

[Формализация задачи 3](#_Toc178589190)

[Теоретическая часть 4](#_Toc178589191)

[Основные шаги программы 5](#_Toc178589192)

[Блок-схема программы 6](#_Toc178589193)

[Описание программы 7](#_Toc178589194)

[Рекомендации пользователя 10](#_Toc178589195)

[Рекомендации для программиста 11](#_Toc178589196)

[Контрольный пример 12](#_Toc178589197)

[Вывод 13](#_Toc178589198)

[Список использованной литературы 13](#_Toc178589199)

# **Цель работы**

Создать генератор синтетических данных о покупке товаров в магазинах Санкт-Петербурга.

# **Формализация задачи**

Задача состоит в создании датасета для железнодорожных билетов со следующими требованиями:

1. **Дата и время**: совпадают с расписание работы магазинов.
2. **Название магазина**: названия магазинов, существующих на территории СПб.
3. **Координаты**: магазина, с точностью до 6 знаков после запятой.
4. **Товар**: товар, который соответствует выбранному магазину.
5. **Бренд**: бренд, соответствующий выбранному товару.
6. **Выбор вагона и места**: Учет классов вагонов и типов мест.
7. **Количество**: от 1 до 5 штук.
8. **Стоимость**: В зависимости от количества и выбранного товара.
9. **Карта оплаты**: Генерация карт с возможностью многократного использования с повторением не больше пяти раз и возможностью настраивать вероятность к какому банку и платежной системе принадлежит карта.
10. **Количество строк датасета**: Не меньше 50000.

# **Теоретическая часть**

Для создания датасета использован программный модуль и набор csv таблиц:

1. main.py: Генерация всех данных, необходимых для решения задачи. Берет нужные данные из файлов general\_table.csv, shop\_locations.csv. После завершения программы создает файл output.csv
2. general\_table.csv: таблица, которая содержит глобальную категорию товаров (clothes, electronics, food), бренд, продающий данные товары, сами товары, среднюю стоимость каждого товара в долларах США.
3. shop\_locations.csv: таблица, которая содержит глобальную категорию товаров (clothes, electronics, food), название магазина, его координаты.

* Количество строк в датасете ограничивается вводом пользователя, но минимальное количество сгенерированных строк будет 1000.
* Веса банков и платежных систем определяются пользователем и должны в сумме давать число больше нуля.
* Логика выбора карт оплаты с ограничением на 5 повторов.
* Соответствие товара, его цены, бренда и магазина, где он был куплен
* Локации магазинов точно совпадают с реальными ( использовался API Яндекс карт)

# **Основные шаги программы**

1. Запуск программы (main.py):
2. Пользователь вводит процентные соотношения банков и платежных систем, а так же выбирает категорию товаров, количество строчек в файле output.csv.
3. Загружаются данные из файлов general\_table.csv, shop\_locations.csv
4. Выполняется файл main.py.
   1. Покупки генерируются с использованием данных о расположении магазинов, персональных данных и цене товаров, а также информации о платежных картах.
      1. Генерация случайного товара с учетом типа магазина и его стоимости.
      2. Генерация локации магазина и времени его работы.
      3. Генерация платежной карты с использованием весов для выбора платежной системы и банка.
5. Данные собираются и записываются в файл output.csv.

# **Блок-схема программы**

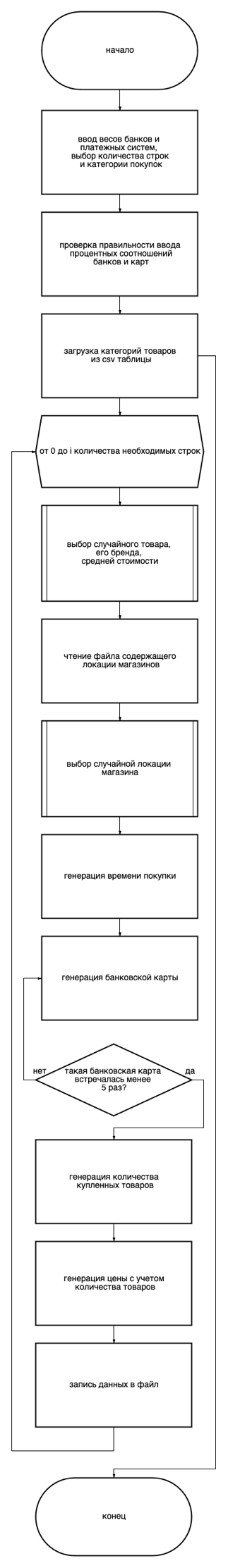


Рисунок блок-схема основной программы



Рис 2. Блок-схема подпрограмм

# **Описание программы**

Программа реализована на языке python 3.11.4, с использованием библиотек: csv, random, geopy, datetime, tkinter. В программе использовались 10 функций и 6 структур данных. В таблице 1 представлено описание функций

Таблица 1. Описание функций

|  |  |  |
| --- | --- | --- |
| Имя функции | Тип возвращаемого значения | Описание функции |
| read\_data | list | Читает csv таблицу и записывает данные в массив |
| generate\_random\_datetime | datetime | Генерирует случайную дату и время в заданном диапазоне |
| generate\_one\_card\_2 | str | Генерирует номера банковских карт, с которых человек может произвести платеж |
| possibility\_generator | array | Возвращает случайный элемент массива с учетом таблицы вероятностей |
| choose\_one\_row | list | Возвращает случайную элемент датасета с учетом фильтров |
| choose\_item\_from\_row | list | Возвращает случайную элемент массива, генерирует цену товара |
| generate\_one\_output | list | Генерирует единый элемент датасета. Является основной функцией программы |
| write\_into\_csv\_file | None | Записывает строчку в csv файл |
| generate\_dataset | None | Генерирует датасет, отвечает за правильное количество данных в датасете |
| clicked | None | Запускает последовательность действий при нажатии кнопки в графическом интерфейсе (проверяет ввод пользователя, отправляет данные в главную функцию) |

В таблице 2 представлено описание структур данных, используемых в программе.

Таблица 1. Описание структур

|  |  |  |  |
| --- | --- | --- | --- |
| Имя структуры | Тип структуры | Тип хранимых данных | Описание |
| painment\_system\_names | list | string | Хранит имена возможных платежных систем |
| bank\_names | list | string | Хранит имена возможных банков |
| possibility\_painment\_sys | list | string | Хранит поля, в которые вводятся вероятности выпадения определенных платежных систем |
| possibility\_banks | list | string | Хранит поля, в которые вводятся вероятности выпадения определенных банков |
| random\_row | list | string | Хранит использованные случайную строчку начального датасета |
| dataset | list | array | Хранит строки датасета, которые были сгенерированы |

# **Рекомендации пользователя**

Для запуска программы необходимо наличие устройства с установленной операционной системой Linux, macOS или Windows, а также среды разработки, поддерживающей запуск python версии 3.11.4

Необходимо установить python библиотеки: datetime, tkinter. Для этого необходимо открыть терминал и прописать следующие команды:

pip install -r requirements.txt

Далее необходимо запустить программу с помощью команды (находитесь в той же директории, что и программа)

python main.py

Под строкой «платежная система» находятся названия платежных систем, а снизу от них, соответствующая каждой системе, комбинированный список. В каждом из комбинированных списков необходимо выбрать значения вероятности генерации для каждой платежной системы (рисунок. 7). Если оставить поле пустым или данные, введенные в него, будут некорректными, то при запуске программы вылезет ошибка.

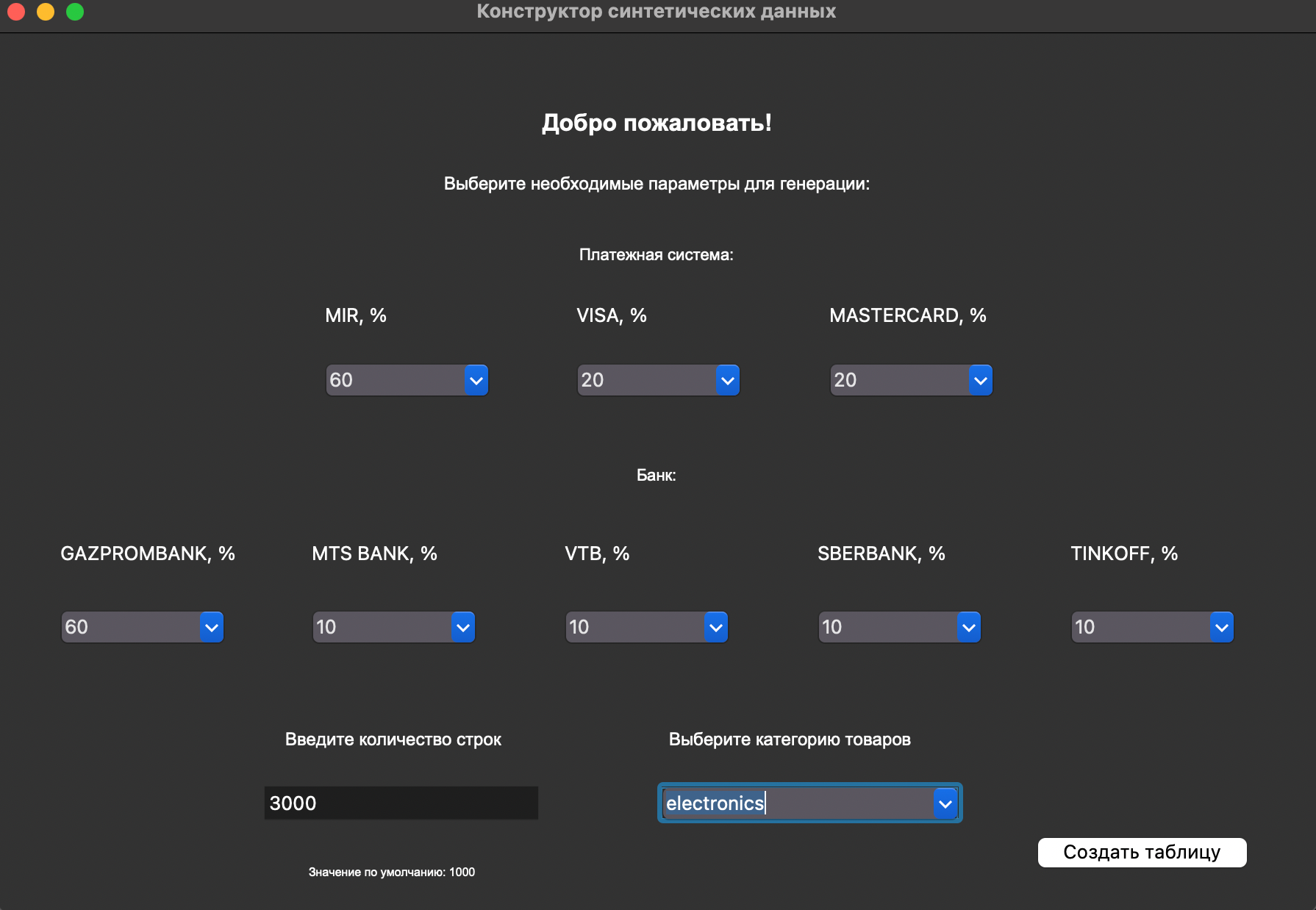


Рисунок 3

Под строкой “Введите значения вероятностей генерации банков” находятся названия банков, а справа от них, соответствующий каждой системе, комбинированный список. В каждом из комбинированных списков необходимо выбрать значения вероятности генерации для каждой платежной системы. (рисунок. 7) Если оставить поле пустым или данные, введенные в него, будут некорректными, то при запуске программы вылезет ошибка (рис 8).

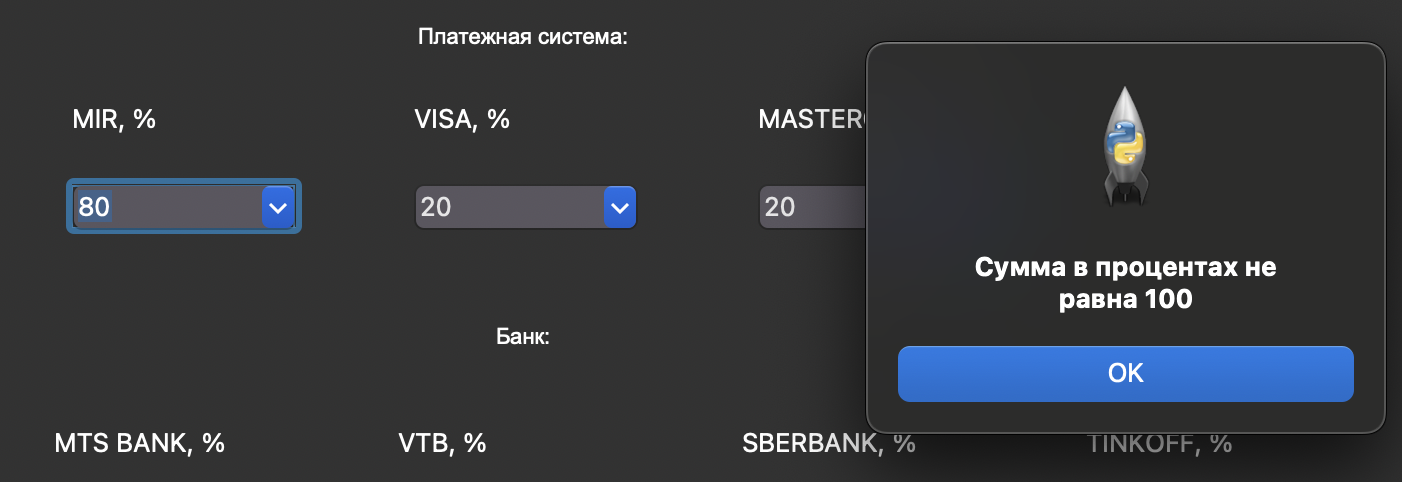


Рисунок 4

В поле расположенное под строкой “Выберите количество строк” необходимо ввести количество строк, которые нужно сгенерировать для датасета (рисунок. 7). Если оставить поле пустым или данные, введенные в него, будут некорректными, то выдаст 1000 строк.

После того, как вы ввели данные, для генерации датасета необходимо нажать на кнопку “Создать таблицу” и дождаться уведомления об успехе. Csv файл будет сохранен в ту директорию ( в папку output ), в которой находится программа.

# **Рекомендации для программиста**

Для внесения изменений в исходный код программы необходимо наличие устройство с установленное операционной системой Linux, macOS или Windows, а также среды разработки, поддерживающей запуск python версии 3.11.4

Исходный код программы доступен по ссылке ниже:

<https://github.com/alice3e/fake_shopping_log_list/tree/main>

# **Контрольный пример**

В этом разделе представлен пример работы программы.

На рисунке 10 представлены параметры, установленные при генерации.

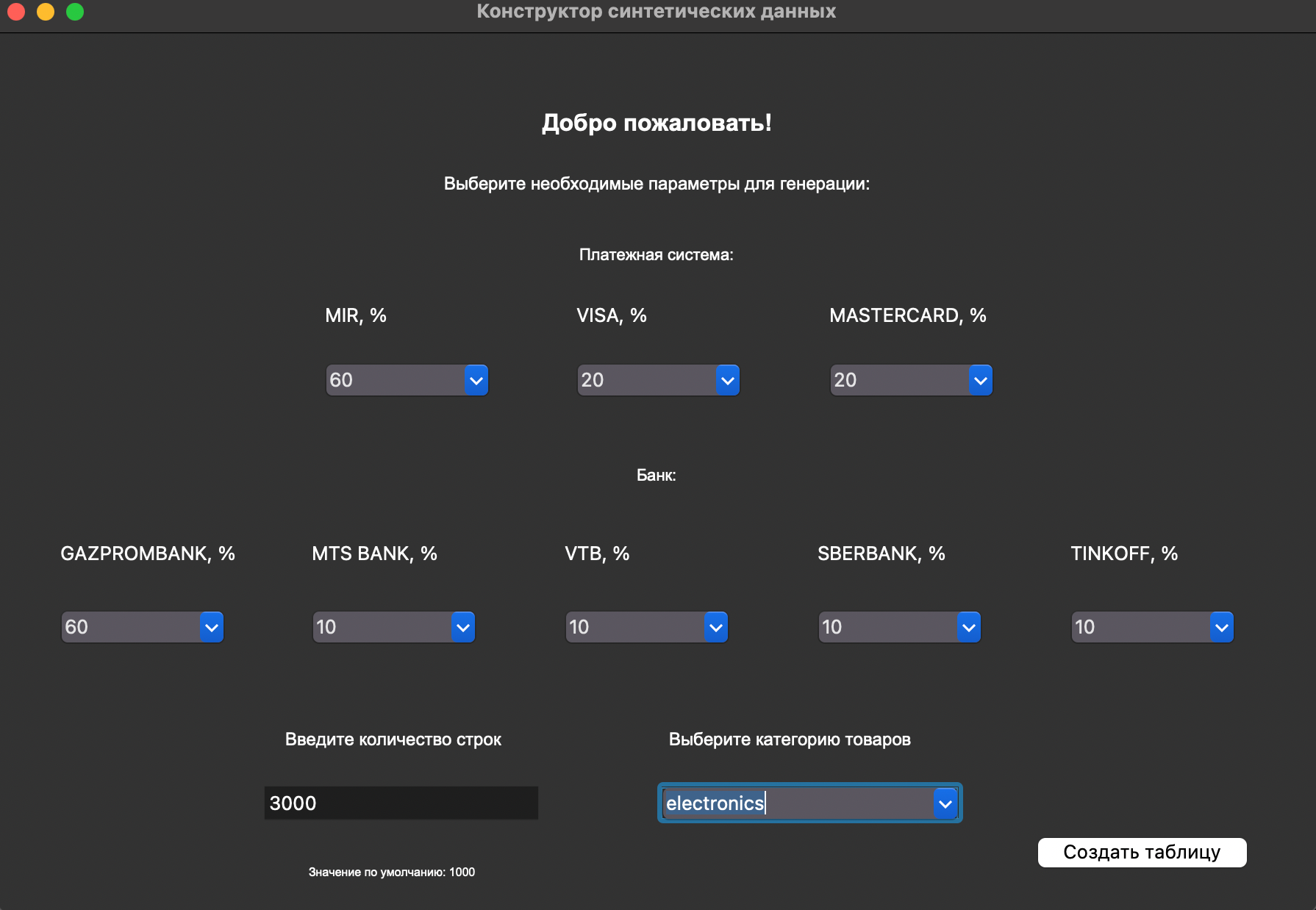


Рисунок 5. Меню программы

Рисунок 5. Меню программы

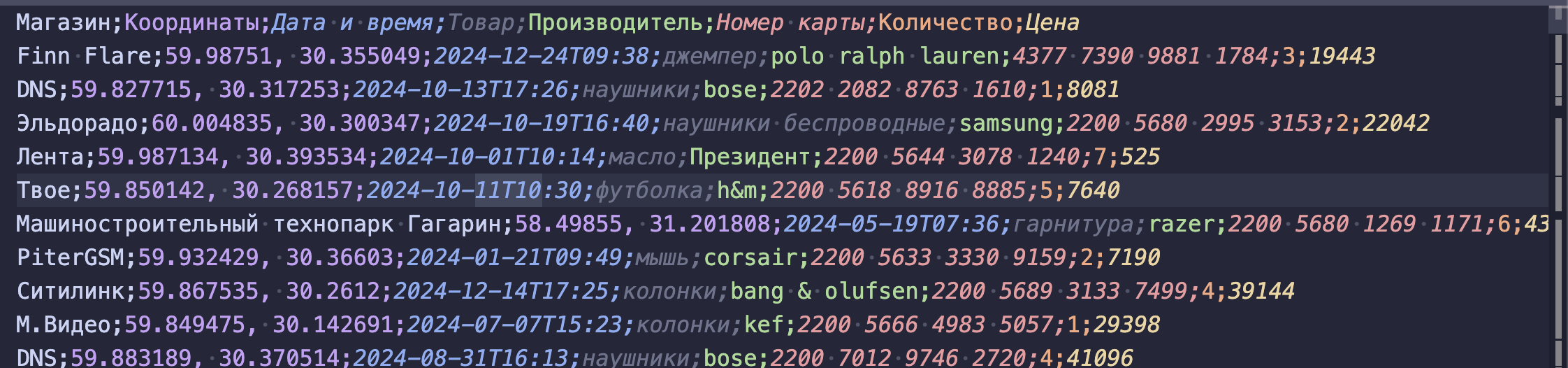


Рисунок 6. Результат работы программы

# **Вывод**

В ходе выполнения данной работы были изучены принципы генерации синтетических данных о покупке различных товаров, разработан алгоритм генерации синтетических данных данного типа и написана программа, которая при вводе определенных параметров генерирует датасет покупки различных товаров на территории СПб.

# Список использованной литературы

[1] Определение локации магазинов: (https://yandex.com/maps-api/)