**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**ОТЧЕТ**

**по лабораторной работе №2**

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

**на тему «Обезличивание** **датасета»**

**Вариант – 1**

**Студент гр. 23Б15-пу**

**Сурин И.С.**

**Преподаватель**

**Дик А.Г.**

**Санкт-Петербург**

**2024 г.**

Оглавление

1. [Цель работы 3](#_Toc177901232)
2. [Описание задачи (формализация задачи) 3](#_Toc177901233)
3. [Теоретическая часть 4](#_Toc177901234)
4. [Основные шаги программы 6](#_Toc177901235)
5. [Блок схема программы 7](#_Toc177901236)
6. [Описание программы 9](#_Toc177901237)
7. [Рекомендации пользователя 10](#_Toc177901238)
8. [Рекомендации программиста 10](#_Toc177901239)
9. [Исходный код программы 11](#_Toc177901240)
10. [Контрольный пример 11](#_Toc177901241)
11. [Вывод 13](#_Toc177901242)
12. [Источники 13](#_Toc177901243)

### **Цель работы:**

Целью лабораторной работы является разработка механизма для **обезличивания** входящего датасета, содержащего информацию о покупках в магазинах. При создании и генерации такого набора данных необходимо учитывать различные требования и условия, связанные с конфиденциальностью данных. Генератор данных должен уметь создавать большие наборы, которые включают:

* Название магазина
* Координаты покупки (включая дату, время и географические координаты)
* Категорию и бренд товара
* Номер платежной карты
* Количество товаров и стоимость покупки

Для достижения этой цели код выполняет следующие задачи:

1. **Генерация обезличенных данных** — обеспечивает создание датасета, соответствующего реальным транзакциям, с настройкой обезличивания полей для защиты конфиденциальной информации.
2. **Обезличивание ключевых полей** — скрывает или обобщает такие атрибуты, как название магазина, географические координаты, дата и время транзакции, категория и бренд товара, номер карты и другие данные, чтобы предотвратить возможность идентификации отдельных покупателей.
3. **Поддержка требований k-анонимности** — код оценивает уровень анонимности данных с помощью показателя **k-анонимности**, чтобы гарантировать, что в полученном наборе данных будет достаточно высокая степень защиты приватности.
4. **Гибкость настройки** — пользователю предоставляется возможность выбора, какие именно данные будут обезличены, что позволяет настроить процесс в зависимости от конкретных задач.
5. **Сохранение итогового датасета** — после завершения обезличивания, итоговый набор данных сохраняется в формате XML, что обеспечивает его удобное использование для анализа, моделирования или дальнейшей обработки, при этом гарантируется защита приватной информации.

Итоговый набор данных должен соответствовать требованиям по объему, качеству, конфиденциальности и содержать **реалистичные транзакции** для использования в исследованиях или передаче данных без угрозы нарушения приватности покупателей.

### **Теоретическая часть**

Для создания и обезличивания датасета, описывающего транзакции покупок в магазинах, используется несколько программных модулей, которые работают с различными аспектами данных. Основное внимание уделено обезличиванию данных с помощью таких методов, как генерализация и замена конкретных значений на более обобщённые, а также оценке анонимности с помощью метрики k-анонимности.

1. **Загрузка данных**: Данные загружаются из XML-файла с помощью библиотеки pandas. Этот файл содержит записи транзакций, которые включают такие поля, как название магазина, координаты, время и дата покупки, категория товаров, бренд, номер карты, количество товаров и стоимость. Загруженный DataFrame копируется в df\_anon для последующей анонимизации.
2. **Функции обезличивания**: В данном коде используются несколько функций для обезличивания данных:
   * **Обезличивание названия магазина**: Функция anon\_Store\_Name() заменяет конкретные названия магазинов на их категории, что помогает скрыть информацию о точных названиях торговых точек, сохраняя при этом связь с их типом.
   * **Обезличивание координат**: В anon\_coordinat() все географические координаты заменяются на одно фиксированное значение — "Санкт-Петербург", что позволяет скрыть фактическое местоположение покупок.
   * **Обезличивание даты и времени**: Функция anon\_Date\_Time() оставляет только год транзакции, удаляя точное время и дату покупки, что способствует защите личных данных.
   * **Обезличивание категорий и брендов**: Категория товара заменяется на символы \*\*\* **d** anon\_category()***,* а бренд — также на *\*\*\**** с помощью функции anon\_brand(), что позволяет скрыть детализированную информацию о типах и производителях товаров.
   * **Обезличивание номеров карт**: В anon\_Card\_Number() номера карт заменяются на названия банков, к которым они относятся (например, "Сбербанк", "Т-банк", "ВТБ"), чтобы сохранить связь с банками, но скрыть реальные номера карт.
   * **Обезличивание количества товаров и стоимости**: В anon\_Quantity() и anon\_Cost() количество товаров и их стоимость разделяются на диапазоны, что позволяет скрыть точные данные, но сохранить информацию о масштабах покупок.
3. **Гибкость обезличивания**: Пользователю предоставляется возможность выбора, какие поля необходимо обезличить через функцию what\_want(). Это позволяет адаптировать процесс обезличивания под конкретные задачи, сохраняя баланс между конфиденциальностью данных и полезностью для анализа.
4. **K- анонимити**: Для оценки уровня обезличивания данных используется метрика **k- анонимити**. Эта метрика вычисляется по квази-идентификаторам (название магазина, координаты, дата и время покупки, количество товаров, номер карты, стоимость, категория товара). Функция calculate\_k\_anonymity() считает, сколько уникальных записей можно найти по указанным полям. Чем выше k, тем лучше данные защищены от идентификации. Это важно для обеспечения конфиденциальности информации в итоговом датасете.
5. **Процентное соотношение K-анонимити**: После вычисления k- **анонимити** выводится процентное соотношение записей, соответствующих каждому значению k. Это помогает оценить распределение уникальных и дублирующихся записей в наборе данных и степень обезличивания.
6. **Сохранение результатов**: После завершения процесса обезличивания данные сохраняются в формате XML с использованием метода to\_xml(). Этот формат поддерживает структурированное хранение информации и широко используется для передачи данных между системами.

Таким образом, разработанный код позволяет гибко обезличить данные о покупках в магазинах, сохраняя структурированность и полезность данных для анализа, одновременно обеспечивая высокий уровень конфиденциальности.

### **Основные шаги программы**

1. **Запуск программы**: Пользователь запускает программу, и загружается датасет из файла dataset.xml.
2. **Создание копии DataFrame**: Создается копия оригинального DataFrame для дальнейшего обезличивания данных.
3. **Обезличивание названий магазинов**: В зависимости от выбора пользователя, происходит замена названий магазинов на их категории с использованием функции find\_category.
4. **Обезличивание координат**: Если пользователь выбирает эту опцию, все координаты заменяются на фиксированное значение "Санкт-Петербург".
5. **Обезличивание даты и времени**: Дата и время покупок обрезаются до года, оставляя только год покупки.
6. **Обезличивание категорий**: Если выбрано, все категории товаров заменяются на фиксированное значение "\*\*\*".
7. **Обезличивание брендов**: В случае выбора пользователем, все бренды заменяются на фиксированное значение "\*\*\*".
8. **Обезличивание номеров карт**: Номера карт заменяются на названия банков (Сбербанк, Т-банк, ВТБ) в зависимости от первых четырех цифр номера карты.
9. **Обезличивание количества товаров**: Количество товаров делится на категории (например, 5-29, 30-50) для анализа.
10. **Обезличивание стоимости**: Итоговая стоимость товаров делится на категории (например, 0-100000, 100000-1000000) для анализа.
11. **Ввод пользователя**: Пользователь выбирает, какие данные он хочет обезличить, через последовательный ввод (да/нет).
12. **Расчет K-анонимности**: Происходит группировка и расчет K-анонимности для выбранных квази-идентификаторов, чтобы определить уровень анонимности данных.
13. **Вывод результата**: Выводятся первые 10 значений K-анонимности и процентное соотношение этих значений относительно общего количества записей.
14. **Запись обезличенных данных**: Все обезличенные данные сохраняются в новый файл dataset\_anon.xml в формате XML.
15. **Отчет о выполнении программы**: Выводится информация о выполнении программы, включая примерный результат и статистику.



Рис 1. Блок-схема основной программы

### **Описание программы**

Программная реализация написана на языке Python 3.11.9 с использованием библиотеки: pandas. Программа организована через модульную структуру, акцентируясь на обезличивании данных о покупках в магазинах. В процессе разработки программы использовался следующий модуль, который отвечает за определенные аспекты обработки данных:

**Таблица1 1.py**

|  |  |  |
| --- | --- | --- |
| Структура данных | Описание | Возвращаемое значение |
| find\_category | Находит категорию магазина по его названию, сопоставляя с заданными данными в store. | str |
| anon\_Store\_Name | Обезличивает названия магазинов в DataFrame, заменяя их на категории. | None |
| anon\_coordinat | Обезличивает координаты, заменяя их на фиксированное значение "Санкт-Петербург". | None |
| anon\_Date\_Time | Обезличивает дату и время, оставляя только год. | None |
| anon\_category | Обезличивает категорию товара, заменяя ее на "\*\*\*". | None |
| anon\_brand | |  | | --- | | Обезличивает бренд товара, заменяя его на "\*\*\*". |  |  | | --- | |  | | None |
| anon\_Card\_Number | |  | | --- | | Обезличивает номера карт, заменяя их на названия банков на основе первых 4 цифр номера. |  |  | | --- | |  | | None |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| anon\_Quantity | |  | | --- | | Обезличивает количество товаров, классифицируя их по заданным диапазонам. |  |  | | --- | |  | | None |
| anon\_Cost | |  | | --- | | Обезличивает стоимость товаров, классифицируя их по заданным диапазонам. |  |  | | --- | |  | | None |
| what\_want | Запрашивает у пользователя, какие данные необходимо обезличить, и вызывает соответствующие функции. | None |
| calculate\_k\_anonymity | |  | | --- | | Вычисляет значения K-анонимности для указанных квазиидентификаторов. |  |  | | --- | |  | | list |

### **Рекомендации пользователя**

Для успешного запуска программы убедитесь, что у вас установлен Python и необходимые библиотеки, такие как pandas. Все модули программы должны находиться в одной директории для корректной работы. Запуск программы осуществляется через файл, содержащий основной код. Перед запуском убедитесь, что ваш файл dataset.xml правильно отформатирован и содержит минимум 50,000 строк для корректной работы с данными.

При запуске программы вам будет предложено обезличить различные аспекты данных, такие как названия магазинов, координаты и даты. Пожалуйста, вводите ответы в соответствии с предложенными вариантами (1 - да, 0 - нет). Убедитесь, что вы понимаете, какие данные будут обезличены, так как это может повлиять на анализ данных. После завершения работы программы обезличенные данные будут сохранены в dataset\_anon.xml.

### **Рекомендации программиста**

Для поддержания актуальности и работоспособности программы используйте последние версии библиотек, особенно pandas. Следите за правильной структурой данных в файле dataset.xml, чтобы избежать ошибок при загрузке и анонимизации данных. Применяйте практики надлежащего именования переменных и функций для улучшения читаемости кода.

Регулярно тестируйте программу на различных наборах данных, проверяя корректность сгенерированных данных, таких как номера карт, стоимости товаров и выбор магазинов. Обратите внимание на возможность добавления новых функций для улучшения обезличивания данных, например, возможность выбора специфических категорий для обезличивания или расширение диапазонов значений для количеств и стоимости.

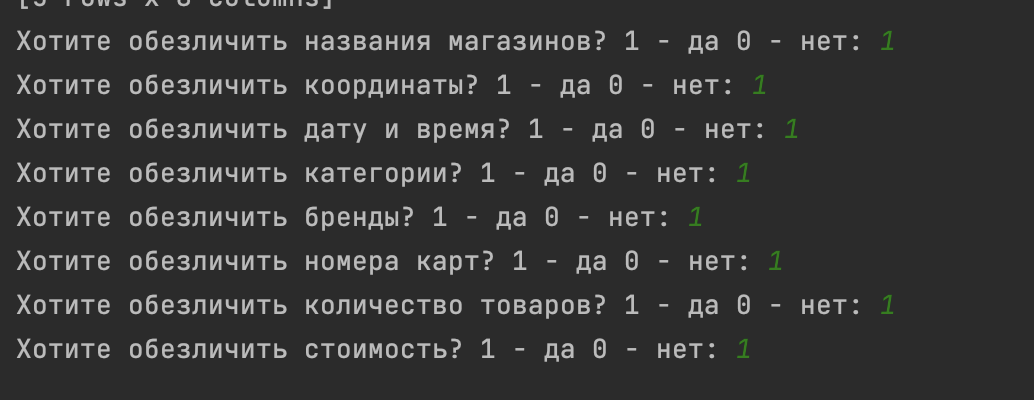
Также рекомендуется комментировать код для облегчения его понимания другими разработчиками или вами в будущем.

# **Исходный код программы:**

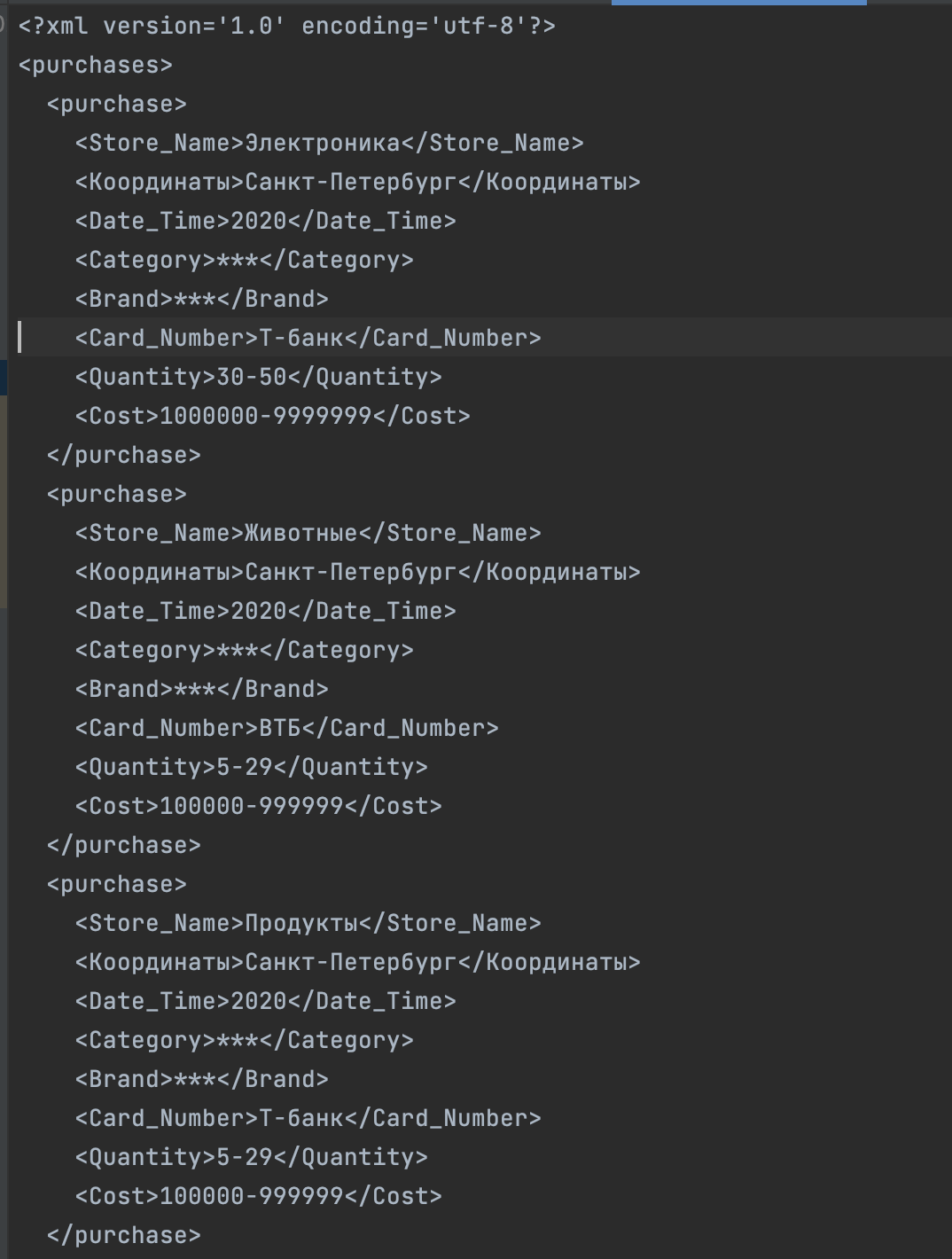
# **Контрольный пример**

1. **Запуск программы**  
   Для запуска программы используйте файл 1.py. Этот скрипт отвечает за обезличивание данных о покупках на основе информации из файла dataset.xml. Программа загружает данные и позволяет пользователю выбрать, какие поля обезличить.
2. **Выбор параметров Обезличивания**  
   После запуска программы пользователю будет предложено ввести, какие параметры он хочет обезличить (Рис. 2). Пользователь может выбрать следующие опции:
   * Обезличивание названий магазинов
   * Обезличивание координат
   * Обезличивание даты и времени
   * Обезличивание категорий
   * Обезличивание брендов
   * Обезличивание номеров карт
   * Обезличивание количества товаров
   * Обезличивание стоимости

Каждую из этих опций можно включить, введя 1, или отключить, введя 0.

Рис 2. Пример ввода критериев

1. **Обработка данных и вывод результатов**  
   После выбора параметров программа обрабатывает данные, а затем выводит изменённый DataFrame и сохраняет обезличенные данные в файл dataset\_anon.xml (Рис. 3). Программа также рассчитывает значения K-анонимности и выводит их на экран, чтобы пользователь мог оценить уровень обезличенных данных.

Рис 2. Пример датасета

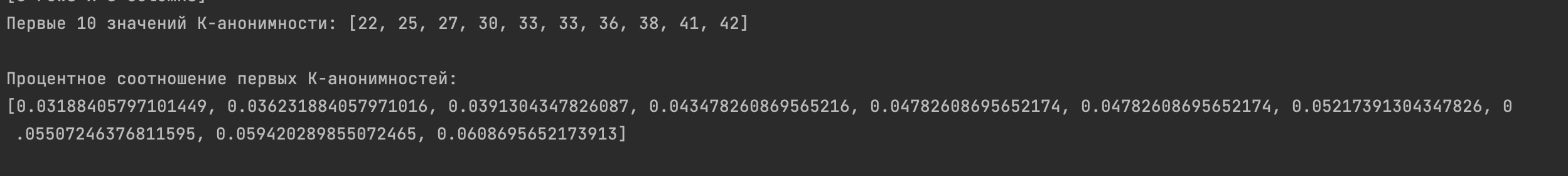


Рис 3. Пример вывод К

### **Вывод**

В рамках данной работы был разработан алгоритм для обезличивания данных о покупках в магазинах. Программа анализирует существующий датасет, который включает в себя такие параметры, как названия магазинов, координаты, дата и время покупок, категории и бренды товаров, номера карт, количество и стоимость покупок. Реализованный алгоритм обеспечивает возможность обезличивания различных аспектов данных, что повышает уровень конфиденциальности информации.

В процессе работы программа позволяет пользователю настраивать параметры обезличивания, выбирая, какие именно данные необходимо скрыть. Это обеспечивает гибкость в обработке данных и позволяет адаптировать программу под специфические требования к обезличиванию. Обезличенные данные сохраняются в формате XML, что упрощает их последующую обработку и анализ. Программа также рассчитывает значения K-анонимности, что позволяет пользователю оценить степень защиты обезличенных данных. Таким образом, разработанное программное обеспечение отвечает современным требованиям к безопасности и конфиденциальности данных

**Источники**

1. pandas — Python Data Analysis Library // pandas URL: <https://pandas.pydata.org/docs/> .
2. XML Processing Modules // Python URL: <https://docs.python.org/3/library/xml.html>