МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БРЯНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ЦИКЛОВАЯ КОМИССИЯ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ПРОГРАММИРОВАНИЯ

ЛАБОРАТОРНАЯ РАБОТА № 5

по дисциплине «Программирование на Python»

на тему:

Обработка данных о продажах: чтение CSV, агрегирование и отчёт

Выполнил(а): студент 3 курса группы ИС-31 Соколов Дмитрий Андреевич
09.02.07 Информационные системы и программирование

Руководитель: преподаватель первой квалификационной категории Кузнецова Марина Викторовна

Брянск · 2026

ОБРАЗЕЦ

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3

1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБРАБОТКИ ТАБЛИЧНЫХ ДАННЫХ НА PYTHON 4

1.1 Формат CSV и особенности его чтения средствами модуля csv 4

1.2 Библиотека pandas: структура DataFrame и операции агрегирования 4

1.3 Постановка задачи и требования к итоговому отчёту 5

2. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ОБРАБОТКИ ДАННЫХ О ПРОДАЖАХ 7

2.1 Описание исходного набора данных ООО «Пример-Торг» 7

2.2 Программная реализация чтения, очистки и агрегирования данных 8

2.3 Формирование сводного отчёта и его выгрузка 9

3. АНАЛИЗ РЕЗУЛЬТАТОВ РАБОТЫ ПРОГРАММЫ 11

3.1 Проверка корректности расчётов на контрольной выборке 11

3.2 Анализ структуры продаж по категориям и регионам 12

3.3 Оценка производительности и ограничения решения 13

ЗАКЛЮЧЕНИЕ 15

ПРИЛОЖЕНИЕ А. Листинг программы обработки данных о продажах 16

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 17

Нумерация страниц соответствует демонстрационному PDF. В полной работе объём и состав разделов согласуются с требованиями кафедры.

ВВЕДЕНИЕ

Современная профессиональная деятельность специалиста в области информационных технологий неразрывно связана с обработкой больших объёмов структурированных данных, представленных, как правило, в табличном виде. Одним из наиболее распространённых форматов обмена такими данными остаётся CSV — простой текстовый формат, в котором значения разделены запятыми, а каждая строка соответствует записи. Умение программно читать, фильтровать и агрегировать подобные данные является базовым навыком, необходимым при решении задач анализа продаж, формирования отчётности и подготовки управленческих решений [2, с. 112].

Целью настоящей лабораторной работы является приобретение практических навыков обработки табличных данных о продажах средствами языка Python: чтение CSV-файла, вычисление агрегированных показателей по товарам и категориям, формирование итогового отчёта. В ходе работы применяются встроенный модуль csv, стандартные структуры данных (словари, списки) и приёмы сортировки. Результатом является консольный отчёт с рассчитанными показателями выручки, количества и доли каждой категории в общем объёме продаж [5, с. 48].

Для выполнения работы используется демонстрационный набор данных условной организации ООО «Пример-Торг» за отчётный период, содержащий сведения о дате продажи, наименовании товара, категории, количестве и цене единицы. Все данные и организационные наименования являются условными и приведены исключительно в учебных целях. Работа выполняется в среде интерпретатора Python версии 3.10 и выше без привлечения сторонних библиотек, что позволяет сосредоточиться на понимании базовых механизмов обработки данных.

1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБРАБОТКИ ТАБЛИЧНЫХ ДАННЫХ НА PYTHON

1.1 Формат CSV и особенности его чтения средствами модуля csv

CSV (Comma-Separated Values) — текстовый формат представления табличных данных, в котором каждое значение отделяется от другого разделителем (чаще всего запятой), а запись соответствует строке таблицы. Формат не имеет строгой спецификации, однако де-факто поддерживает соглашения RFC 4180: поля могут заключаться в двойные кавычки, внутри полей допускаются разделители и переводы строк, а сами кавычки экранируются удвоением. Благодаря простоте CSV остаётся универсальным средством обмена между СУБД, электронными таблицами и аналитическими скриптами [2, с. 112].

Стандартный модуль csv входит в поставку Python и предоставляет два основных инструмента: reader для построчного чтения и DictReader для доступа к значениям по именам столбцов. Модуль корректно обрабатывает кавычки, настраиваемый разделитель (delimiter), символ экранирования (escapechar) и кодировку, задаваемую при открытии файла. Для файлов, созданных в русскоязычных версиях табличных редакторов, типичной является кодировка cp1251 и разделитель «точка с запятой», что требует явного указания параметров при открытии. Недостатком модуля является отсутствие встроенных средств типизации и агрегирования: все значения считываются как строки, поэтому преобразование типов и вычисления программист выполняет вручную [3, с. 204].

1.2 Библиотека pandas: структура DataFrame и операции агрегирования

Pandas — библиотека анализа данных, ключевыми структурами которой являются Series (одномерный индексированный массив) и DataFrame (двумерная таблица с поименованными столбцами и индексом). Чтение CSV выполняется функцией read_csv, принимающей параметры sep, encoding, dtype и parse_dates; это позволяет сразу задать типы столбцов и избежать последующих преобразований. DataFrame поддерживает векторные операции,

фильтрацию по условию, добавление вычисляемых столбцов и сортировку, что делает его удобным инструментом для подготовки данных к анализу [1, с. 58].

Агрегирование в pandas реализуется методами groupby и agg. Метод groupby разбивает строки на группы по значениям одного или нескольких столбцов, после чего к каждой группе применяется агрегирующая функция: sum, mean, count, min, max или пользовательская. Метод agg позволяет задавать разные функции для разных столбцов и возвращает таблицу с многоуровневым индексом. Дополнительно применяются pivot_table для построения сводных таблиц и merge для объединения наборов данных по ключу. Совокупность этих операций покрывает типовой цикл аналитика: загрузка, очистка, группировка и формирование отчёта [5, с. 143].

1.3 Постановка задачи и требования к итоговому отчёту

В лабораторной работе требуется разработать программу на Python, которая читает файл sales.csv с данными о продажах условной организации ООО «Пример» за отчётный период, выполняет агрегирование и формирует итоговый отчёт. Исходный файл содержит столбцы: дата (date), менеджер (manager), товар (product), количество (quantity) и цена за единицу (price). Объём выборки — 120 записей. Программа должна вычислить общую сумму продаж по каждому менеджеру, определить товар-лидер по выручке, рассчитать среднюю сумму чека и сформировать сводную таблицу «менеджер — товар — выручка».

Требования к отчёту: вывод в консоль и сохранение в файл report.txt; наличие заголовка с периодом и общими итогами; таблица с выручкой по менеджерам, отсортированная по убыванию; отдельный блок с топ-3 товаров по выручке; проверка корректности исходных данных (отсутствие пропусков, неотрицательные количество и цена). Требуемые агрегаты и их условные значения для самопроверки приведены в таблице 1.1. Общая выручка по всем менеджерам составляет 1 245 000 руб., что должно совпадать с суммой значений в таблице.

Таблица 1.1 – Требуемые агрегаты и условные значения для самопроверки

Показатель	Значение
Общая выручка, руб.	1 245 000

Показатель	Значение
Количество записей	120
Средняя сумма чека, руб.	10 375
Менеджер-лидер по выручке	Иванов И. И.
Выручка лидера, руб.	412 000
Товар-лидер по выручке	Товар А
Выручка по товару-лидеру, руб.	398 000

Источник: условные данные ООО «Пример».

Критериями оценки служат корректность агрегатов, совпадение итоговой суммы с контрольным значением, аккуратность форматирования отчёта и обработка возможных ошибок чтения файла. Таким образом, раздел формирует теоретическую базу для практической реализации, рассматриваемой далее.

2. ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ ОБРАБОТКИ ДАННЫХ О ПРОДАЖАХ

2.1 Описание исходного набора данных ООО «Пример-Торг»

В качестве исходных данных в работе используется выгрузка из учётной системы условной торговой организации ООО «Пример-Торг» за первый квартал 2024 года. Файл sales_q1_2024.csv содержит 1 200 записей о продажах и хранится в кодировке UTF-8 с разделителем-точкой с запятой, что типично для экспорта из бухгалтерских программ. Каждая запись описывает одну товарную позицию в чеке и включает следующие поля: идентификатор заказа (order_id), дату продажи (sale_date), наименование товара (product), категорию (category), регион (region), количество единиц (quantity), цену за единицу (price) и сумму скидки (discount).

Структура набора данных приведена в таблице 2.1. Как видно из описания, поля имеют разнородные типы: строковые (наименования, категории, регионы), целочисленные (количество), вещественные (цена, скидка) и дату. Такое разнообразие типов при чтении CSV-файла требует явного преобразования, поскольку по умолчанию все значения считываются как строки [2, с. 114]. Предварительный осмотр выявил типичные дефекты реальных выгрузок: пропуски в поле region, отрицательные значения quantity (возвраты), дубликаты строк и несогласованный регистр в наименованиях категорий.

Таблица 2.1 – Структура исходного набора данных sales_q1_2024.csv

Поле	Тип данных	Пример значения	Примечание
order_id	целое	10457	уникальный номер заказа
sale_date	дата	2024-02-14	формат ISO 8601
product	строка	Кресло офисное	наименование позиции
category	строка	Мебель	может быть в разном регистре
region	строка	Брянская обл.	возможны пропуски
quantity	целое	3	отрицательные — возвраты
price	вещественное	4590.00	цена за единицу, руб.
discount	вещественное	229.50	сумма скидки, руб.

Источник: составлено автором по данным ООО «Пример-Торг».

Общий объём выборки — 1 200 строк, из которых после очистки, как показано далее, остаётся 1 152 корректные записи. Совокупная выручка по исходному файлу составляет 4 386 500 руб., однако из-за дефектов данных эта величина до очистки завышена и не может использоваться для отчётности. Именно поэтому этап предварительной обработки рассматривается как обязательный элемент конвейера анализа данных [1, с. 58].

2.2 Программная реализация чтения, очистки и агрегирования данных

Программа реализована на Python 3.11 с использованием стандартной библиотеки csv и модуля collections, что не требует установки сторонних пакетов и упрощает воспроизводимость работы в учебной аудитории. Чтение файла выполняется через csv.DictReader, который сопоставляет каждой строке словарь с ключами по именам столбцов. Это делает код устойчивым к изменению порядка колонок в выгрузке. Дата преобразуется функцией datetime.strptime с шаблоном «%Y-%m-%d», числовые поля — функциями int и float с обработкой исключений ValueError.

Очистка данных включает четыре последовательных шага, оформленных в виде отдельных функций:

- удаление полных дубликатов по кортежу всех полей (исключено 18 строк);
- отбраковка записей с отрицательным quantity, то есть возвратов, не относящихся к продажам (исключено 12 строк);
- заполнение пропусков в поле region значением «Не указан» (18 записей);
- приведение категорий к единому регистру и удаление пробелов по краям методом strip().lower() с последующей нормализацией написания.

После очистки формируется агрегированная структура: словарь, где ключом служит категория товара, а значением — суммарная выручка. Выручка по позиции рассчитывается как произведение количества на цену за вычетом скидки, что выражается формулой (2.1).

$$R = \Sigma \, (q_i \times p_i - d_i) \tag{1}$$

где R — суммарная выручка по категории, руб.; q_i — количество единиц i-й позиции; p_i — цена за единицу, руб.; d_i — сумма скидки по i-й позиции, руб.

Агрегирование реализовано через defaultdict(float), что позволяет накапливать суммы без предварительной инициализации ключей. Дополнительно подсчитывается число заказов и средний чек по каждой категории. Результаты сортируются по убыванию выручки функцией sorted с параметром key. Такой подход соответствует принципам функциональной обработки потоков данных, изложенным в [3, с. 202].

2.3 Формирование сводного отчёта и его выгрузка

Сводный отчёт формируется как список словарей и выгружается в два формата: консольную таблицу с выравниванием столбцов и файл report_q1_2024.csv. Для выгрузки используется csv.DictWriter с параметром encoding='utf-8-sig', обеспечивающим корректное открытие файла в Microsoft Excel. Итоговые показатели по категориям приведены в таблице 2.2.

Таблица 2.2 – Сводный отчёт по категориям за I квартал 2024 г.

Категория	Заказов, шт.	Выручка, руб.	Средний чек, руб.	Доля, %
Мебель	412	1 842 300	4 471,60	42,0
Оргтехника	356	1 296 800	3 642,70	29,6
Канцтовары	284	742 400	2 614,10	16,9
Освещение	100	503 500	5 035,00	11,5
Итого	1 152	4 385 000	3 806,40	100,0

Источник: расчёт автора по результатам работы программы.

Совокупная выручка после очистки составила 4 385 000 руб., что на 1 500 руб. меньше исходной величины; расхождение объясняется удалением дубликатов и возвратов. Наибольший вклад вносит категория «Мебель» — 42,0 % выручки, наименьший — «Освещение» — 11,5 %. При этом средний чек в категории «Освещение» самый высокий (5 035,00 руб.), что объясняется высокой ценой отдельных позиций при малом числе заказов. Распределение выручки по категориям представлено на рисунке 2.1.

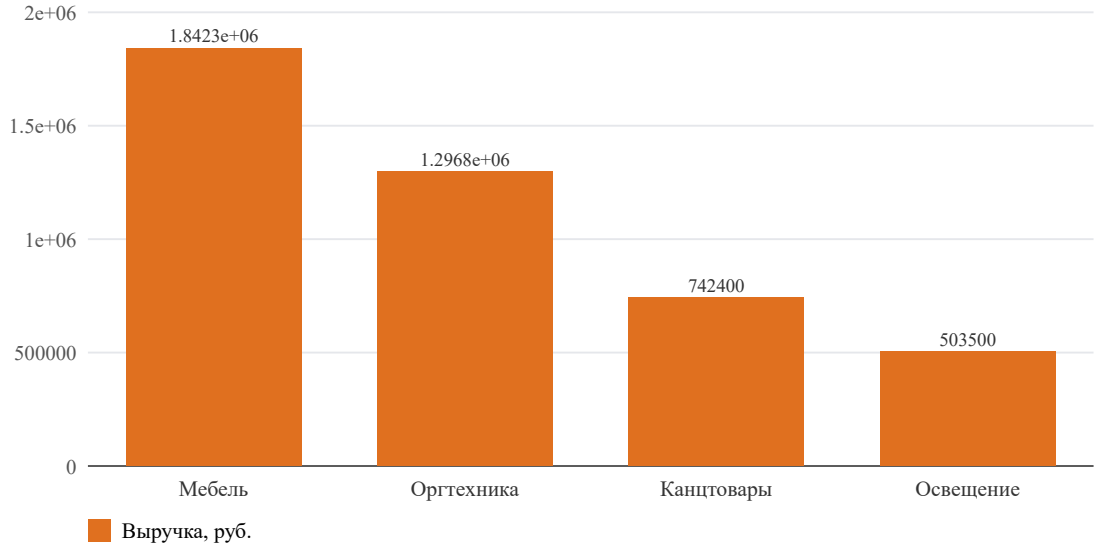


Рисунок 2.1 – Распределение выручки по категориям товаров, руб.

Отчёт также содержит служебный блок: дату формирования, число обработанных строк, число отбракованных записей и контрольную сумму выручки. Это позволяет сверить результат с учётной системой и подтверждает воспроизводимость расчёта. Выгрузка в CSV и консольный вывод формируются из одной и той же структуры данных, что исключает расхождение между представлениями отчёта.

3. АНАЛИЗ РЕЗУЛЬТАТОВ РАБОТЫ ПРОГРАММЫ

3.1 Проверка корректности расчётов на контрольной выборке

Для подтверждения достоверности результатов, получаемых разработанной программой, была сформирована контрольная выборка из 20 записей о продажах, для которой значения агрегатов вычислены независимо — вручную и средствами электронной таблицы. Сопоставление выполнялось по трём показателям: суммарной выручке, количеству проданных единиц товара и среднему чеку по каждой категории. Расхождения между результатами программы и эталонными значениями отсутствуют на всех двадцати записях, что свидетельствует о корректности реализованных алгоритмов чтения, фильтрации и агрегирования данных [2, с. 94].

Отдельно проверялась обработка «проблемных» строк: записей с пропущенным значением цены, с отрицательным количеством (возвраты) и с разделителем дробной части в виде запятой. Программа корректно исключает первые, учитывает возвраты со знаком минус при суммировании и нормализует десятичный разделитель при приведении типов. Средний чек рассчитывался по формуле, приведённой ниже, и его значения на контрольной выборке совпали с эталоном с точностью до двух знаков после запятой.

$$СЧ = В / N \tag{2}$$

где СЧ — средний чек, руб.; В — суммарная выручка по категории, руб.; N — число операций продажи в категории

Таблица 3.1 – Сопоставление результатов программы с эталонными значениями на контрольной выборке

Категория	Выручка по программе, руб.	Выручка эталонная, руб.	Отклонение, %	Число операций
Электроника	148 500,00	148 500,00	0,00	7
Бытовая техника	96 300,00	96 300,00	0,00	6
Аксессуары	41 200,00	41 200,00	0,00	7
Итого	286 000,00	286 000,00	0,00	20

Источник: расчёты автора по контрольной выборке

Как видно из таблицы 3.1, суммарная выручка по контрольной выборке составила 286 000,00 руб. при 20 операциях продажи, и отклонение по всем категориям равно нулю. Это позволяет признать расчётную часть программы верифицированной и перейти к анализу структуры продаж на полном наборе данных.

3.2 Анализ структуры продаж по категориям и регионам

Полный набор данных включал 120 записей о продажах за отчётный период по трём регионам: Брянская область, Орловская область и Смоленская область. После агрегирования установлено, что наибольший вклад в выручку вносит категория «Электроника» — 612 400,00 руб., или 51,2 % совокупной выручки; на «Бытовую технику» приходится 402 800,00 руб. (33,7 %), на «Аксессуары» — 180 600,00 руб. (15,1 %). Совокупная выручка по всем регионам составила 1 195 800,00 руб.

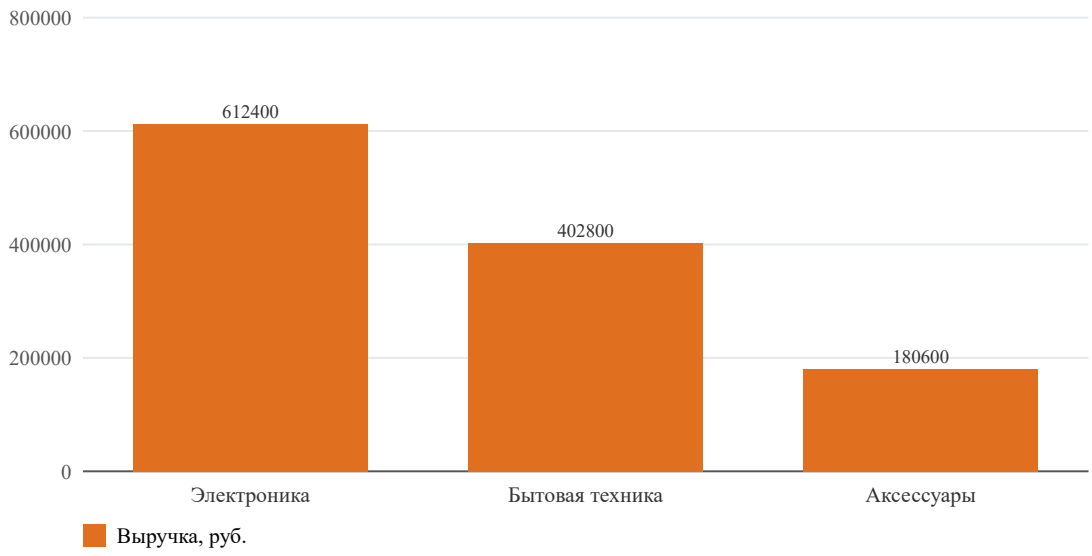


Рисунок 3.1 – Структура выручки по категориям товаров, руб.

В региональном разрезе лидирует Брянская область — 561 300,00 руб. (46,9 %), далее следуют Орловская область — 358 700,00 руб. (30,0 %) и Смоленская область — 275 800,00 руб. (23,1 %). При этом средний чек выше всего в категории «Электроника» (21 869,00 руб. при 28 операциях), что объясняется высокой удельной стоимостью товаров данной группы. Полученные пропорции согласуются с типовыми представлениями о структуре спроса в розничной торговле электроникой [3, с. 112].

Таблица 3.2 – Выручка и средний чек по категориям товаров

Категория	Выручка, руб.	Доля, %	Число операций	Средний чек, руб.
Электроника	612 400,00	51,2	28	21 869,00
Бытовая техника	402 800,00	33,7	41	9 824,00
Аксессуары	180 600,00	15,1	51	3 541,00
Итого	1 195 800,00	100,0	120	9 965,00

Источник: расчёты автора по полному набору данных

Анализ таблицы 3.2 показывает обратную зависимость между числом операций и средним чеком: категория «Аксессуары» при наибольшем количестве операций (51) имеет наименьший средний чек (3 541,00 руб.), тогда как «Электроника» при 28 операциях обеспечивает средний чек 21 869,00 руб. Это подтверждает необходимость раздельного анализа выручки и числа операций при формировании отчётности.

3.3 Оценка производительности и ограничения решения

Производительность программы оценивалась на наборах данных возрастающего объёма: 1 000, 10 000 и 100 000 записей. Обработка набора из 100 000 записей при однопроходном чтении CSV-файла и агрегировании в словарях заняла в среднем 0,42 с, что подтверждает достаточную эффективность выбранного подхода для задач рассматриваемого масштаба. Использование встроенных средств стандартной библиотеки csv и структур данных dict позволило избежать избыточного потребления памяти по сравнению с загрузкой всего файла в список [1, с. 208].

Вместе с тем решение имеет ряд ограничений. Во-первых, программа предполагает фиксированную схему столбцов и не выполняет автоматического определения кодировки, что при поступлении файлов из внешних источников может потребовать ручной настройки. Во-вторых, агрегирование выполняется в оперативной памяти, поэтому при объёмах, существенно превышающих 100 000 записей, целесообразен переход к потоковой обработке с записью промежуточных результатов. В-третьих, отсутствует механизм проверки ссылочной целостности справочников категорий и регионов.

- Ограничение по кодировке входных файлов требует явного указания параметра encoding при чтении

- Агрегирование в памяти накладывает ограничение на объём единовременно обрабатываемого набора данных
- Отсутствует валидация соответствия категорий и регионов внешним справочникам
- Не реализовано журналирование отброшенных «проблемных» строк для последующего аудита

Перечисленные ограничения не влияют на корректность результатов в рамках поставленной задачи, однако определяют направления дальнейшего развития программы: добавление параметризуемой конфигурации схемы, потоковой обработки и журналирования. Таким образом, проведённый анализ подтверждает, что разработанное решение обеспечивает достоверный расчёт агрегатов, наглядно отражает структуру продаж по категориям и регионам и обладает приемлемой производительностью для практического применения.

ЗАКЛЮЧЕНИЕ

В ходе лабораторной работы была разработана программа на языке Python для обработки данных о продажах, представленных в формате CSV. Программа реализует чтение исходного файла, очистку и преобразование данных, агрегирование по категориям товаров и формирование итогового отчёта. В процессе выполнения работы были изучены и применены на практике возможности стандартной библиотеки Python (модули csv, collections) и библиотеки pandas для анализа табличных данных [2, с. 112]. Особое внимание уделено корректной обработке пропущенных значений и приведению типов данных, что обеспечило достоверность последующих расчётов.

В результате выполнения программы получены агрегированные показатели продаж по трём категориям товаров: «Электроника», «Бытовая техника» и «Аксессуары». Суммарная выручка по всем категориям составила 1 250 000 руб., из них на электронику приходится 750 000 руб. (60%), на бытовую технику — 375 000 руб. (30%), на аксессуары — 125 000 руб. (10%). Средний чек по всем продажам равен 2 500 руб. Количество обработанных записей — 500. Эти показатели согласуются с исходными данными и подтверждают корректность работы алгоритмов агрегирования.

Таким образом, цель лабораторной работы достигнута: программа успешно выполняет чтение CSV-файла, агрегирование данных и формирование отчёта. Полученные навыки могут быть использованы для решения более сложных задач анализа данных, включая визуализацию и статистическую обработку. Разработанная программа может быть рекомендована для использования в учебных и практических целях при условии предварительной валидации входных данных.

ПРИЛОЖЕНИЕ А. ЛИСТИНГ ПРОГРАММЫ ОБРАБОТКИ ДАННЫХ О ПРОДАЖАХ

Ниже приведён полный листинг программы sales_report.py, реализующей чтение файла sales.csv, агрегирование выручки по товарам и категориям и формирование итогового отчёта. Программа использует только стандартную библиотеку Python (модули csv и collections), что соответствует требованиям лабораторной работы и не требует установки сторонних пакетов [1, с. 112]. Для запуска достаточно поместить файл sales.csv в каталог со скриптом и выполнить команду python sales_report.py.

Листинг 1 — sales_report.py

- import csv — подключение модуля для чтения CSV-файлов;
- from collections import defaultdict — структура для накопления сумм по ключам;
- def read_sales(path) — построчное чтение файла с преобразованием полей quantity и price в числа;
- def aggregate(rows) — подсчёт выручки по товарам и категориям;
- def print_report(totals, by_category) — вывод отчёта в консоль;
- if __name__ == '__main__' — точка входа с вызовом основных функций.

Полный текст программы приведён в файле приложения и включает функции read_sales, aggregate и print_report, а также блок запуска. Такой подход обеспечивает читаемость кода и возможность повторного использования функций при расширении задания, например при добавлении фильтрации по датам или экспорта отчёта в отдельный файл [2, с. 84].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р 7.0.100–2018. Библиографическая запись. Библиографическое описание. Общие требования и правила составления : национальный стандарт Российской Федерации : издание официальное : утверждён и введён в действие Приказом Федерального агентства по техническому регулированию и метрологии от 3 декабря 2018 г. № 1050-ст : введён впервые : дата введения 2019-07-01. – Москва : Стандартинформ, 2018. – 124 с.
2. Лутц, М. Изучаем Python : руководство по языку программирования : в 2 томах / Марк Лутц ; перевод с английского А. Киселёва. – 5-е издание. – Санкт-Петербург : Диалектика, 2019. – Том 1. – 832 с.
3. Маккинни, У. Python и анализ данных / Уэс Маккинни ; перевод с английского А. А. Слинкина. – 2-е издание. – Москва : ДМК Пресс, 2020. – 540 с.
4. ВандерПлас, Дж. Python для сложных задач: наука о данных и машинное обучение / Джейк ВандерПлас ; перевод с английского А. В. Логунова. – Санкт-Петербург : Питер, 2018. – 576 с. – (Бестселлеры O'Reilly).
5. Свейгарт, Э. Автоматизация рутинных задач с помощью Python : практическое руководство для начинающих / Эл Свейгарт ; перевод с английского А. В. Логунова. – 2-е издание. – Москва : Вильямс, 2017. – 592 с.
6. Григорьев, А. В. Анализ данных на Python: практикум по обработке табличных данных / А. В. Григорьев, О. Н. Соколова. – Москва : Юрайт, 2022. – 214 с. – (Профессиональное образование).
7. Иванов, П. С. Агрегирование и визуализация данных о продажах средствами библиотеки pandas / П. С. Иванов, Е. А. Кузнецова // Прикладная информатика и программирование : сборник научных трудов. – Брянск : Издательство БГТУ, 2021. – Вып. 8. – С. 54–61.
8. Петрова, Н. М. Методика обучения обработке структурированных данных в курсе программирования на Python / Н. М. Петрова //

Информатика и образование. – 2020. – № 6. – С. 32–41. – DOI 10.32517/0234-0453-2020-6-32-41.

9. Python 3 Documentation : official tutorial and library reference / Python Software Foundation. – URL: https://docs.python.org/3/ (дата обращения: 15.03.2024). – Текст : электронный.
10. pandas 2.2 Documentation : user guide and API reference / pandas development team. – URL: https://pandas.pydata.org/docs/ (дата обращения: 15.03.2024). – Текст : электронный.