Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Ордена Трудового Красного Знамени

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра «Информатики»

Курсовая работа по дисциплине

«Структуры и алгоритмы обработки данных»

Выполнил: студент группы БФИ2201

НДЗИЛА Полин

Проверил: СИМОНОВ С. Е

Москва 2024

**Введение**

В этом отчете описывается решение проблемы, связанной с анализом данных о продажах в магазине из файла CSV. Задачи следующие:

* Рассчитать общую выручку магазина.
* Определить самый продаваемый товар.
* Найти товар, который принес наибольшую выручку.

Для решения этой задачи мы будем использовать язык программирования ***Python*** и библиотеку ***pandas***, которая особенно хорошо подходит для манипулирования и анализа данных.

**Разработка**

Используемая библиотека

* *pandas*: для чтения и работы с данными в CSV-файле

**Шаги решения**

* Чтение CSV-файла: Мы будем использовать pandasp для чтения CSV-файла и обработки любых ошибок ввода/вывода.
* Вычисление общей выручки магазина: мы вычислим сумму столбца «Общая стоимость».
* Определение самого продаваемого товара: с помощью groupby и sump найдем товар с наибольшим общим количеством продаж.
* Определение товара, приносящего наибольший доход: мы используем groupbyet sump, чтобы найти товар с наибольшим общим доходом.
* Формирование отчета: мы создадим отчет, содержащий запрошенную информацию, отсортированную по общей выручке.

**Python-код**

**Описание программы**

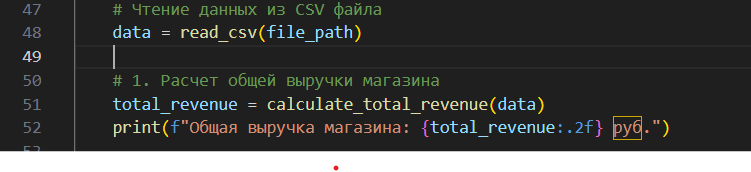
Программа написана на языке Python и состоит из нескольких функций, каждая из которых выполняет определенную задачу

****

**Pисунке 1** извлечение библиотеки и строки к CSV-файлу со статьями из журнала

**Расчет общей выручки магазина**

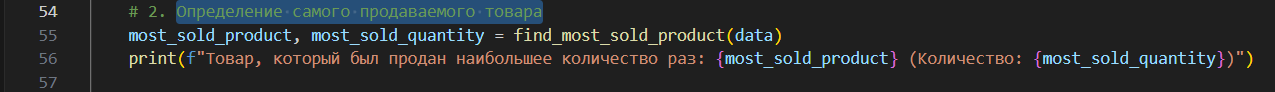
Эта функция позволяет запускать calculate\_total\_revenue с аргументами данных

****

**Pисунке 2** : Расчет общей выручки магазина

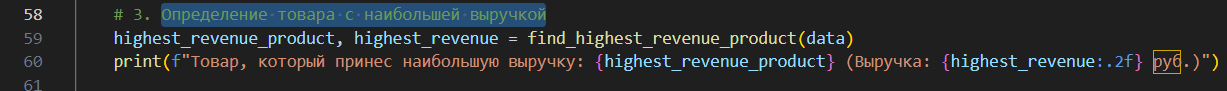
Определение самого продаваемого товара

Эта операция позволяет найти на экране ссылки на производство и редактирование.

****

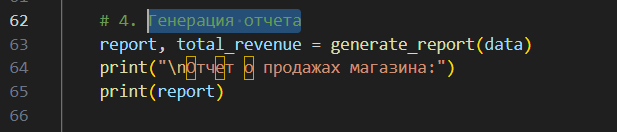
Pисунке 3 : Определение самого продаваемого товара

**Эти расширения позволяют определить продукт, приносящий наибольший доход.**

****

Pисунке 4 : Определение товара с наибольшей выручкой

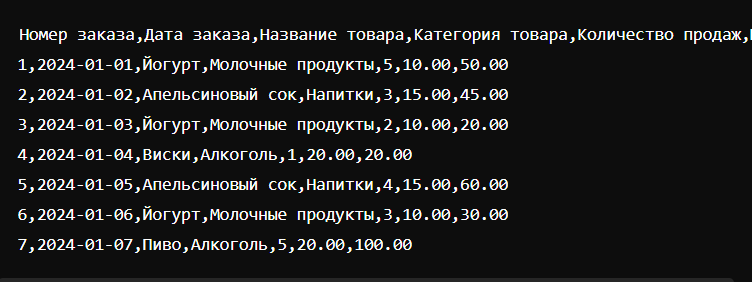
Эта кнопка появляется на экране дисплея «Отчет о продажах магазина:» и на нем появляется («Отчет»). Вы также найдете подробную информацию о товарах магазина.

****

Pисунке 5 : Генерация отчета

|  |
| --- |
| **import pandas as pd**  **def read\_csv(file\_path):**  **try:**  **return pd.read\_csv(file\_path)**  **except FileNotFoundError:**  **print("Ошибка: Файл не найден.")**  **exit()**  **except pd.errors.EmptyDataError:**  **print("Ошибка: Файл пуст.")**  **exit()**  **except pd.errors.ParserError:**  **print("Ошибка: Проблема при чтении файла.")**  **exit()**  **def calculate\_total\_revenue(data):**  **return data['Общая стоимость'].sum()**  **def find\_most\_sold\_product(data):**  **product\_sales = data.groupby('Название товара')['Количество продаж'].sum()**  **most\_sold\_product = product\_sales.idxmax()**  **most\_sold\_quantity = product\_sales.max()**  **return most\_sold\_product, most\_sold\_quantity**  **def find\_highest\_revenue\_product(data):**  **product\_revenue = data.groupby('Название товара')['Общая стоимость'].sum()**  **highest\_revenue\_product = product\_revenue.idxmax()**  **highest\_revenue = product\_revenue.max()**  **return highest\_revenue\_product, highest\_revenue**  **def generate\_report(data):**  **total\_revenue = calculate\_total\_revenue(data)**  **product\_sales = data.groupby('Название товара')['Количество продаж'].sum()**  **product\_revenue = data.groupby('Название товара')['Общая стоимость'].sum()**    **report = pd.DataFrame({**  **'Количество продаж': product\_sales,**  **'Общая выручка': product\_revenue,**  **'Доля в общей выручке (%)': (product\_revenue / total\_revenue) \* 100**  **}).sort\_values(by='Общая выручка', ascending=False)**    **return report, total\_revenue**  **def main():**  **file\_path = 'C:/024/devoir.csv'**    **# Чтение данных из CSV файла**  **data = read\_csv(file\_path)**    **# 1. Расчет общей выручки магазина**  **total\_revenue = calculate\_total\_revenue(data)**  **print(f"Общая выручка магазина: {total\_revenue:.2f} руб.")**    **# 2. Определение самого продаваемого товара**  **most\_sold\_product, most\_sold\_quantity = find\_most\_sold\_product(data)**  **print(f"Товар, который был продан наибольшее количество раз: {most\_sold\_product} (Количество: {most\_sold\_quantity})")**    **# 3. Определение товара с наибольшей выручкой**  **highest\_revenue\_product, highest\_revenue = find\_highest\_revenue\_product(data)**  **print(f"Товар, который принес наибольшую выручку: {highest\_revenue\_product} (Выручка: {highest\_revenue:.2f} руб.)")**    **# 4. Генерация отчета**  **report, total\_revenue = generate\_report(data)**  **print("\nОтчет о продажах магазина:")**  **print(report)**    **# Сохранение отчета в CSV файл**  **report.to\_csv('отчет\_о\_продажах\_магазина.csv', index=True, encoding='utf-8-sig')**  **print("\nОтчет сохранен в файл 'отчет\_о\_продажах\_магазина.csv'.")**  **if \_\_name\_\_ == "\_\_main\_\_":**  **main()** |

**Результат**

****

Pисунке 6 : подробная информация о каждом предмете

После использования скрипта результаты отображаются на консоли, и вы должны следовать им:

Общая выручка магазина: 325.00 руб.

Товар, который был продан наибольшее количество раз: Йогурт (Количество: 10)

Товар, который принес наибольшую выручку: Пиво (Выручка: 100.00 руб.)

**Отчет о продажах магазина:**

Количество продаж Общая выручка Доля в общей выручке (%)

*Название товара*

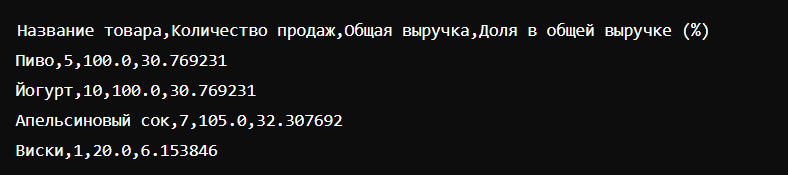
Пиво 5 100.0 30.769231

Йогурт 10 100.0 30.769231

Апельсиновый сок 7 105.0 32.307692

Виски 1 20.0 6.153846

Отчет сохранен в файл 'отчет\_о\_продажах\_магазина.csv'.



Pисунке 7 : результат консольного поиска

**Объяснение кода**

Чтение CSV-файла:

Функция read\_csv использует pandasp для чтения файла и обработки возможных ошибок (файл не найден, пустой файл, ошибка чтения).

**Вычисление общего дохода:**

Функция calculate\_total\_revenue вычисляет сумму столбца «Общая стоимость».

**Определение самого продаваемого товара:**

Функция find\_most\_sold\_product использует groupby для группировки данных по названию товара и sump для подсчета общего количества продаж. Затем она определяет товар с максимальным количеством продаж.

**Определение продукта, приносящего наибольший доход:**

Функция find\_highest\_revenue\_product использует groupbyp для группировки данных по названию товара и sump для расчета общей выручки по каждому товару. Затем она определяет товар с наибольшим общим доходом.

**Генерация отчета** :

Функция generate\_report генерирует DataFrame, содержащий количество продаж, общую выручку и долю каждого товара в общей выручке, отсортированный по убыванию общей выручки. Отчет сохраняется в CSV-файле.

**Заключение**

Эта программа на Python использует эффективные структуры данных и алгоритмы обработки данных для анализа продаж магазина из CSV-файла. Результаты включают подсчет общей выручки, определение самого продаваемого товара и самого высокодоходного продукта, а также создание подробного отчета о продажах. Ошибки ввода/вывода обрабатываются соответствующим образом, чтобы обеспечить надежность программы.