**Вариант 3 конкурсных заданий *практического* этапа Конкурса**

Вы с друзьями решили открыть продуктовый магазин. Друзья отвечают за рекламу и операционную деятельность. Спустя несколько месяцев работы Вы поняли, что часть продуктов не распродается и ее необходимо списывать, что негативно влияет на общую прибыль. Ваша задача состоит в том, чтобы на основе накопленных данных написать функционал приложения для автоматизации учета продаж продуктов.

Думаю, что Вы готовы приступать, но сначала необходимо **создать репозиторий** для проекта и задач, которые Вы будете выполнять. Обязательно сделайте его **PUBLIC**, а то мы не сможем проверить решение вашей последней задачи. Каждую задачу вам необходимо будет правильно оформить и залить в репозиторий. *Код, который Вы напишете необходимо задокументировать, чтобы другие программисты могли понять, что делает код и за что отвечает. Примеры документирования приведены ниже.*

**Пример документирования кода на языке С++**

/\*\*

\* Это описание функции foo

\*

\* @param str это описание аргумента str

\* @param pattern это описание аргумента pattern

\* @return это описание того, что вернет функция

\*/

int foo(std::string str, std::string& pattern)

{

...

};

**Пример документирования на языке Python**

def complex(real=0.0, imag=0.0):

"""Описание функции complex.

Описание аргументов:

real – описание аргумента

imag – описание аргумента

"""

if imag == 0.0 and real == 0.0: return complex\_zero

...

Теперь можете приступать к решению задач!

К задачам прикреплен файл *products.csv,* который хранит в себе информацию о продукте, категории, цены за 1 кг, количество проданных товаров.

Столбцы: *Category, product, Date(в формате ДД.ММ.ГГГГ), Price per unit(вещественное число), Count(вещественное число)*

Разделитель *«;».*

*Кодировка: “utf-8-sig”*

***Задача 1.***

Вы собрали уже большое кол-во данных и хотите подвести итог полученной выручки за весь период. Для этого воспользуйтесь данными из таблицы *products.csv.* Все результаты по продуктам необходимо записать в новый столбец с названием total (значение расчитывать исходя из столбцов *Price per unit и Count)* и сохранить в таблицу products\_new.csv. В консоль необходимо вывести итоговую сумму по категории Закуски.

Значение в столбце total рассчитывается путем умножения стоимости единицы продукции на количество проданных единиц.

*В задаче запрещено использование сторонних библиотек(Pandas и др)*

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка (для языка Python – PEP 257). После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория.*

***Задача 2***

Для комфортной работы и эффективной автоматизации учета продаж следует отсортировать по группам в алфавитном порядке категории товаров в таблице *products.csv* и в первой категории вывести самый дорогой товар.

Формат вывода: “В категории: <Category> самый дорогой товар: <product> его цена за единицу товара составляет <Price per unit>”

В задаче нельзя использовать сторонние сортировки, для решения воспользуйтесь сортировкой вставками.

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория*

***Задача 3***

**Ввод: стандартный ввод**

**Вывод: стандартный вывод**

Напишите программу, которая по заданному названию категории выводит продукт с наименьшим кол-вом проданных единиц в заданной категории в формате:

“В категории: <Category> товар: <product> был куплен <Count> раз”, если по заданной категории ничего не найдено, то вывести “Такой категории не существует в нашей БД”

Программа должна постоянно самостоятельно запрашивать название категории и остановиться только в случае ввода слова ‘молоко’.

***Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория***

***Задача 4***

Вы решили устроить акцию в магазине. Для этого вам необходимо создать промокоды для каждого товара. Реализуйте методы/функции, которые будут генерировать промокоды для продуктов. Промокод должен состоять из первых 2-ух букв названия + день поступления + 2 предпоследних буквы названия в обратном порядке + месяц поступления в обратном порядке.

Например, для товара “Напитки для здоровья” поступившего 17.08.2022 промокод должен выглядеть как “НА17ЯЬ80”.

“Напитки;Напитки для здоровья;12.06.2022;297.0;25.0” → “Напитки;Напитки для здоровья;12.06.2022;297.0;25.0;НА12ЯЬ60”

*На вход подается CSV файл, который необходимо записать в список, для каждого элемента сгенерировать промокод, после чего дополнить список сгенерированными элементами, записать сгенерированные значения в столбец promocode. Последним этапом полученный список записать в новый product\_promo.csv файл.*

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория.*

***Задача 5***

Для проведения маркетинговой кампании необходимо определить, какие из продуктов продаются меньше всего. Реализуйте хэш-таблицу, где ключом будет являться название категории продукта, а значением количество его продаж.

Выведите список из 10 наименее продаваемых продуктов в формате:

<Название категории>, <количество>.

*Не забудьте сделать комментарии к коду согласно стандартам документирования кода выбранного языка. После выполнения необходимо сделать локальные и удаленные изменения Вашего репозитория.*

***Задача 6***

Ваш код будет использоваться программистом, которого возьмут на работу, поэтому он должен быть правильно оформлен и выложен на GitHub. Весь написанный код должен быть задокументирован согласно стандартам документирования кода выбранного языка.

Также необходимо оформить README.md для Вашего репозитория. Пункты, которые должны быть описаны:

1. Название проекта

2. Описание проекта

3. Оглавление (необязательно)

4. Как установить и запустить проект

5. Как использовать проект