# Вариант 30 конкурсных заданий *практического* этапа Конкурса

Вы устроились программистом в компанию, которая занимается продажей компьютерной техники. На данный момент учет всех устройств ведется в Excel-таблице, что является сложным, так как объем продаж растет каждый месяц. Вам выгрузили данные в формате csv и попросили реализовать функционал, чтоб автоматизировать учет и подбор устройств.

Вы готовы приступать к задачам, 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 *devices.txt,* который хранит в себе информацию о компании, модели, типе, разрешении экрана, параметрах памяти и цене.

Столбцы: *Company, Product, TypeName, Inches, Cpu, Ram, Memory, Gpu, Price*

Разделитель *«\*».*

***Задача 1.***

Вам поручили задание подсчитать количество ноутбуков(Ultrabook, Notebook, Netbook) с различными характеристиками Ram. В исходном файле подсчитайте количество устройств и запишите в файл count\_company.txt, в формате:

Utrabook

<Ram 1> - <count>

…

<Ram N> - <count>

…

Notebook

<Ram 1> - <count>

…

<Ram N> - <count>

…

Netbook

<Ram 1> - <count>

…

<Ram N> - <count>

…

Где N - количество различных вариантов в этой категории. После этого выведите характеристику для Utrabook в формате:

Utrabook

<Ram 1> - <count>

…

<Ram N> - <count>

*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 комфортного пользования файлом вы решили его отсортировать по названию компании в обратном алфавитном порядке. Для этого воспользуйтесь сортировкой с средней сложностю O(n log n). В задаче нельзя использовать встроенные сортировки! После сортировки выведите первых пять компаний в формате:

<Company> - <Product> - <Price>

*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**

Пришло время сделать наработку для интерфейса, который будет взаимодействовать с базой данных. Для этого Вам необходимо написать консольную программу, которая будет запрашивать у пользователей название компании, а на выход будет выдавать информацию по самому дорогому устройству, если ничего не найдено будет выводить: “У нас нет данного устройства”. Программа должна всегда запрашивать данные о модели товара. Прекратить свою работу она сможет только после ввода “=”.

Формат вывода ответа:

По вашему запросу: <Company> найдены следующие варианты:

<Company> <Product> - тип устройства: <TypeName>; Разрешение экрана - <Inches>; Цена - <Price>

Поиск необходимо осуществить с помощью двоичного поиска в исходном файле.

**Обратите внимание, что данные в файле не отсортированы. Для сортировки можете воспользоваться встроенными сортировками или из задачи 2.**

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

***Задача 4***

Руководство компании хочет посчитать за реализацию продуктов, какой из компаний можно получить максимальную прибыль. Сформируйте список, в котором будут содержаться названия компаний и сумма за продажу всей техники, имеющейся в базе.

Выведите получившуюся информацию на экран в формате:

Если продать все ноутбуки <Название компании> можно заработать <сумма цен>.

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

***Задача 5***

Вы разрабатываете интеллектуального помощника для менеджеров из отдела по холодным продажам. У менеджера есть скрипты, но количество техники достаточно большое, поэтому поиск работает неэффективно. Для быстрой выдачи подсказок реализуйте хэш-таблицу, где ключом является «компания»+ «продукт», а значением цена.

Выведите первые 10 значений сформированной таблицы в формате:

<Компания> <Продукт> <цена>

*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