# Вариант 28 конкурсных заданий *практического* этапа Конкурса

Ваши родители решили приобрести новый автомобиль, но не знают с чего начать подбор. В интернете вы нашли файл с описанием автомобилей, имеющихся в наличии. Ваша задача состоит в том, чтобы реализовать функционал, помогающий определиться с выбором.

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 cars.txt, который хранит в себе информацию о марке, модели, пробеге, цене, дате и цвете.

Столбцы: price, year, manufacturer, model, odometer, paint\_color

Разделитель «$»*.*

*Кодировка: “utf-8”*

***Задача 1.***

Ваши родители определили для себя, что точно хотят машину с пробегом меньше 10000. Составьте для них файл odometer\_car.txt со всеми машинами подходящий под их запрос, где каждая строчка файла будет в формате:

<manufacturer> <model>; Цвет: <paint\_color>; Пробег: <odometer>; Цена: <price>

После этого выведите родителям пять машин с пробегом меньше 10000 и цветом «серебро» в формате:

<manufacturer> <model> есть машина серебряного цвета. Ее стоимость <price> и пробег: <odometer>

*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 машины по возрастанию их цены. Для этого воспользуйтесь быстрой сортировкой. В задаче нельзя использовать встроенные сортировки! После сортировки выведите первые три самые дешевые машины машины в формате:

Вам могут подойти:

<manufacturer> <model>; Цвет: <paint\_color> ; Пробег: <odometer>; Цена: <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 их бюджет(верхнюю и нижнюю границу включительно через пробел), а на выход будет выдавать все подходящие модели в рамках бюджета с их характеристиками, если ничего не найдено будет выводить: “К сожалению, под ваш бюджет ничего не удалось найти”. Программа должна всегда запрашивать данные о бюджете. Прекратить свою работу она сможет только после ввода “стоп”.

Формат вывода ответа:

Исходя из вашего бюджета: <min\_input> - <max\_input> найдены следующие варианты:

1. <manufacturer> <model> цена <price>, пробег данной машины составляет <odometer>

…

N. <manufacturer> <model> цена <price>, пробег данной машины составляет <odometer>

где N - кол-во записей подходящих под условие

**Гарантируется, что первое вводимое число меньше второго.**

Поиск необходимо осуществить с помощью двоичного поиска в исходном файле.

*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 заметило, что клиенты часто выбирают машины по цвету. Ваша задача состоит в том, чтобы сформировать списки машин каждого цвета. По каждому из формированных списков необходимо вывести следующую информацию:

<Количество машин цвета 1> машин цвета <название цвета 1> есть сегодня в наличии.

<Количество машин цвета 2> машин цвета <название цвета 2> есть сегодня в наличии.

…

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

***Задача 5***

Вам необходимо посчитать сколько моделей каждой марки представлено в наборе данных. Составьте хэш-таблицу, где ключом будет название марки автомобиля, а значением – количество автомобилей этой марки.

Выведите информацию из сформированной таблице в формате:

<название марки> <количество моделей>

для следующих марок: buick , chrysler, volvo, infiniti, lincoln, acura, hyundai, mercedes-benz, audi, bmw.

*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