**Вариант 11 конкурсных заданий *практического* этапа Конкурса**

В виртуальном мире видеоигр произошла непредвиденная ситуация. Внезапный программный сбой нарушил работу всех игровых персонажей, сделав их поведение хаотичным и непредсказуемым. Игровое сообщество столкнулось с массой проблем, поскольку многие игры стали практически непроходимыми из-за возникающих ошибок и лагов. Ваша задача разработать программу, способную обнаруживать и исправлять подобные ошибки, возвращая игровым персонажам должное поведение и делая прохождение игр вновь комфортным и приятным.

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 *game.csv,* который хранит в себе информацию о название игры; имени персонажа, имени ошибки, дате фиксации бага.

Столбцы: *GameName, characters, nameError(буквенное значение:числовое значение), date(ГГГГ-ММ-ДД).*

Разделитель *«,».*

***Задача 1.***

Самым важным персонажем для нашего игрового мира является Avery. Найдите все игры в которых есть данный персонаж и составьте отчет в виде файла: game\_new.txt, где каждая строка будет представлена в формате: “У персонажа <characters> в игре <GameName> нашлась ошибка с кодом: <nameError>. Дата фиксации: <date>”. После предоставления отчета выведите первую игру в списке с Avery.

*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***

У вас снова запросили отчет по ошибкам в играх. Теперь отчет необходимо предоставить о количестве багов для каждого персонажа. Для этого отсортируйте данные из файла *game.csv* по столбцу персонаж в алфавитном порядке с помощью быстрой сортировки. После этого предоставьте отчет в формате:  
<Персонаж 1> - количество багов: <count>

….

<Персонаж N> - количество багов: <count>

Где N - количество уникальных персонажей.

*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 Вас имя ошибки, а на выход будет выдавать список игр, в которых встречается эта ошибка, если ничего не найдено будет выводить: “Этой ошибки не существует”. Программа должна всегда запрашивать название ошибки. Прекратить свою работу она сможет только после ввода “game”.

Поиск необходимо осуществить с помощью последовательного алгоритма поиска**.**

**Формат ответа на запрос пользователя:**

“Ошибка <nameError> встречается в игре <GameName> у персонажа <characters>.”

Поиск необходимо осуществлять в файле game.csv

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

***Задача 4***

Необходимо собрать статистику по играм. Сруппируйте данные по названию игры и сделайте подсчет количества ошибок для каждой из них. Необходимо реализовать вывод списка игр с указанием суммарного количества ошибок и списком персонажей с их ошибками по следующей структуре:

*Game 1:*

*Total errors: X*

*Character 1:*

*- Error 1*

*- Error 2*

*Character 2:*

*- Error 1*

*...*

*Game 2:*

*Total errors: Y*

*Character 1:*

*- Error 1*

*...*

Пример работы программы:

Предположим, что файл game.csv имеет следующее содержимое:

Game1,Character1,Error1,2022-01-01

Game1,Character1,Error2,2022-01-10

Game1,Character2,Error1,2022-01-15

Game2,Character1,Error1,2022-01-20

Результат работы программы:

Game 1:

Total errors: 3

Character 1:

- Error 1

- Error 2

Character 2:

- Error 1

Game 2:

Total errors: 1

Character 1:

- Error 1

*Полученные структуры данных необходимо записать в новый файл game\_analysis.txt*

*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 получили файл game.csv, который содержит информацию о названии игры, имени персонажа, имени ошибки и дате фиксации бага. Для решения этой задачи реализуйте хэш-таблицу, в которой значением будет являться название игры и имя персонажа, а ключом имя ошибки. Затем программа должна анализировать данные и предоставлять следующую информацию:

1. Список всех игр, в которых были обнаружены ошибки.

2. Список всех ошибок, отсортированный по дате их фиксации.

Требования:

1. Используйте хеш-таблицы для эффективного хранения и поиска данных.

2. Программа должна выводить информацию в соответствии с описанными пунктами.

Пример содержимого файла game.csv:

Game,Character,Error,Date

Game1,Character1,Error1,2022-01-01

Game1,Character2,Error2,2022-01-02

Game1,Character1,Error3,2022-01-03

Game2,Character3,Error4,2022-01-01

Game2,Character3,Error5,2022-01-04

Game3,Character4,Error6,2022-01-05

Ожидаемый вывод программы:

Игры с ошибками:

Game1

Game2

Game3

Список ошибок по датам:

2022-01-01: Error1, Error4

2022-01-02: Error2

2022-01-03: Error3

2022-01-04: Error5

2022-01-05: Error6

*Полученный список ошибок запишите в новый файл error.txt в формате:*

*date: ['nameError']*

*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