ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

"ВЫСШАЯ ШКОЛА ЭКОНОМИКИ"

**Московский институт электроники и математики им. А. Н. Тихонова**

Марков Д. Э., Милосердов А. В., Фурса Д. Д., Хусаенов Т. И. – авторы проекта, Группа БИВ184

***Информационно-аналитическое приложение для работы с базой данных видеокарт Nvidia***

**Руководство пользователя**

Проект по дисциплине «Проектный семинар “Python в науке о данных”»

студентов образовательной программы бакалавриата

«Информатика и вычислительная техника»

Студенты: Марков Д. Э.

Милосердов А. В.

Фурса Д. Д.

Хусаенов Т. И.

Москва 2019 г.

Оглавление

[1 Описание решаемой задачи 3](#_Toc10922270)

[2 Описание структуры базы данных 3](#_Toc10922271)

[3 Описание интерфейса программы 4](#_Toc10922272)

[4 Описание подкаталогов 9](#_Toc10922273)

[5 Требуемые пакеты и библиотеки 9](#_Toc10922274)

[6 Системные требования 9](#_Toc10922275)

# Описание решаемой задачи

Информационно-аналитическое приложение, разработанное на языке Python позволяет управлять базой данных видеокарт Nvidia.

Приложение обладает следующим функционалом:

* Загрузка базы данных из каталога Data и выведение данных на экран
* Редактирование записей базы данных
* Удаление записей базы данных
* Сортировка данных: по объему видеопамяти, по кол-ву FPS в играх и т.п.
* Анализ записей: подсчет среднего арифметического по значениям числовых полей; подсчет выборочной дисперсии по значениям числовых полей
* Формирование отчетного файла результатов анализа записей, сохранение файла в каталоге Output

# Описание структуры базы данных

Каждая запись в базе данных состоит из полей:

* Название видеокарты
* Дата релиза
* Архитектура
* Базовая тактовая частота
* Конфигурация памяти
* Поддержка технологии Nvidia SLI
* Поддержка технологии RTX
* Энергопотребление
* Производительность в игре Far Cry 5
* Производительность в игре Fallout 4
* Производительность в игре Witcher 3
* Результаты тестирования в бенчмарке 3D Mark Cloud Gate
* Результаты тестирования в бенчмарке 3D Mark Fire Strike
* Средняя цена

# Описание интерфейса программы

* 1. Главное окно

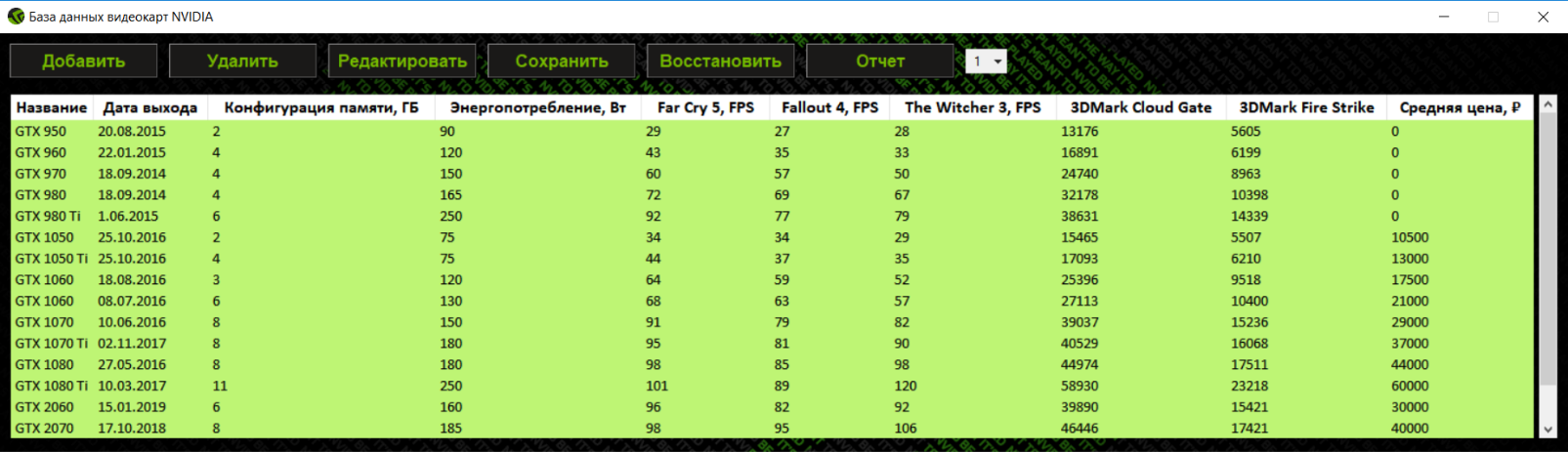
Приложение запускается с помощью скрипта main, находящегося в папке Scripts. Запускать приложение необходимо из командной строки, находясь в папке Scripts. База данных автоматически загружается из двоичного формата при наличии соответствующего файла, в ином случае загружается исходная база данных (из csv).

Рис. 1. Главное окно

На верхней панели главного окна расположены 6 кнопок для работы с базой данных и меню для переключения между справочниками:

Рис. 2. Переключение между справочниками

* 1. Функция «Сохранить»

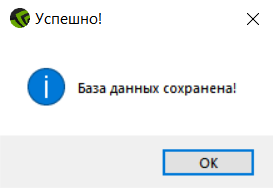
Для сохранений изменений после редактирования базы данных нужно нажать на кнопку «Сохранить», файл будет сохранен в папку data, и появится сообщение об успешном сохранении изменений:

Рис. 2. Сохранение

* 1. Функция «Добавить»

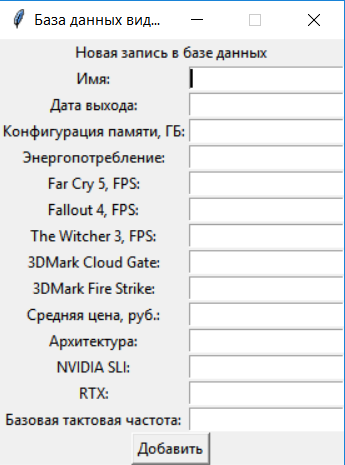
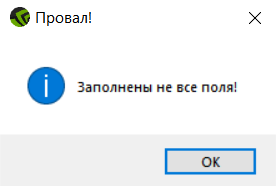
Чтобы добавить новую запись в базу данных нужно нажать кнопку «Добавить» и заполнить все поля для ввода данных:

Рис. 3. Функция добавить

В случае попытки добавления записи с хотя бы одним незаполненным полем будет выведено соответствующее сообщение об ошибке:

Рис. 4. Сообщение об ошибке



* 1. Функция «Удалить»

Выберите одну или несколько записей, нажмите кнопку «Удалить», записи будут удалены из всех пяти справочников. Если при попытке использования функции Удалить» не выбрана ни одна из записей, появится соответствующее сообщение об ошибке.

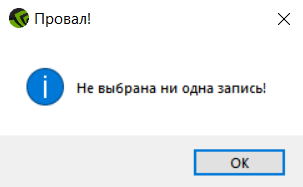


Рис. 5. Сообщение об ошибке 2

* 1. Функция «Редактировать»

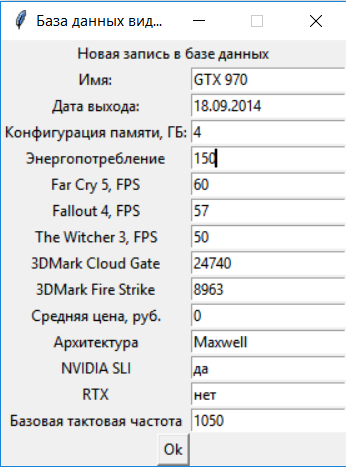
Для редактирования записи выберете нужную запись и нажмите кнопку «Редактировать». Откроется окно редактирования записи, измените желаемые характеристики и нажмите кнопку «ок»:

Рис. 6. Окно редактирования записи

* 1. Функция «Восстановить»

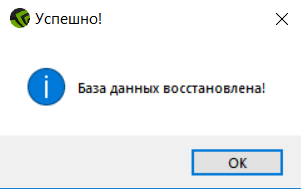
При нажатии кнопки «Восстановить» загружается исходная версия базы данных, все изменения и удаления записей откатываются. Восстановление базы данных сопровождается выводом соответствующего сообщения.

Рис. 7. Сообщение о восстановлении

* 1. Функция «Отчет»

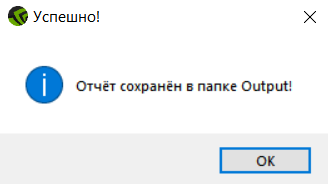
Для формирования отчета нажмите кнопку «отчет». Успешное формирование отчета сопроводится выводом следующего сообщения:

Рис. 8. Создание отчета

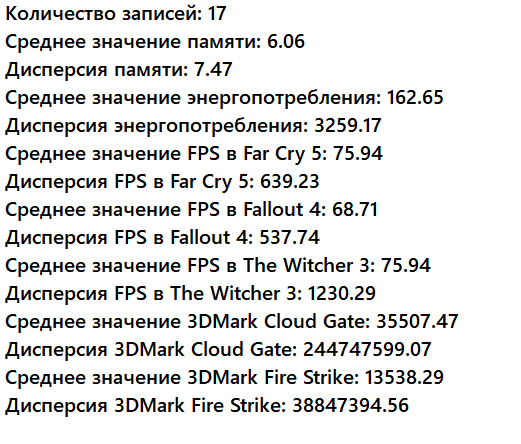
Создание отчета подразумевает создание текстового файла и четырех диаграмм в папке Output. В текстовом файле содержатся результаты подсчета количества записей, средние значения некоторых параметров и результаты вычисления дисперсии. В папке Output также будут сформированы: гистограмма, диаграмма Бокса-Вискерса, диаграмма рассеивания и столбчатая диаграмма. Пример текстового отчета представлен на рисунке ниже:

Рис. 9. Пример текстового отчета

# Описание подкаталогов

Составные части программы хранятся в каталоге с именем «Work», внутри которого хранятся шесть директорий:

* Data – директория для хранения база данных
* Graphics – директория для хранения графических изображений, используемых программой
* Library – директория для хранения пользовательской библиотеки
* Notes – каталог для документации, в нем хранится руководство пользователя и руководство разработчика
* Output – директория для сохранения отчетов по анализу отобранных записей
* Scripts – каталог для хранения специализированных программных блоков, в частности файла с определением параметров приложения и файла с основным скриптом приложения.

# Требуемые пакеты и библиотеки

Библиотека tkinter для создания графического интерфейса, sys для работы с путями к файлам и модулям, numpy, mathplotlib, pandas, pickle.

# Системные требования

ОС система Windows 7/8/10 (32 или 64 бит) Linux, MacOC, процессор с тактовой частотой не ниже 1,5 ГГц, монитор с разрешением экрана не ниже 1280x720, установленный интерпретатор Python 3.6 и выше, установленные библиотеки pandas 0.24.2, pickle numpy 1.16.4, tkinter 1.1.0, matplotlib 3.1.0