ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»  
МОСКОВСКИЙ ИНСТИТУТ ЭЛЕКТРОНИКИ И МАТЕМАТИКИ

**РУКОВОДСТВО РАЗРАБОТЧИКА**

Студент(ы) группы БИВ224:

* Половников Дмитрий Владимирович, dvpolovnikov@edu.hse.ru;
* Сиротин Артём Иванович, aisirotin@edu.hse.ru;
* Бородин Илья Дмитриевич, idborodin@edu.hse.ru.

Тема работы:

**Приложение для анализа автомобилей**

Руководитель:

Полякова Марина Васильевна

Москва 2023г.

**Оглавление**

[**Введение** 3](#_Toc136956529)

[**Описание необходимого окружения** 4](#_Toc136956530)

[**Установка и запуск** 4](#_Toc136956531)

[**Обращение к базе данных** 4](#_Toc136956532)

[**Структура приложения** 5](#_Toc136956533)

[**Описание функций анализа базы данных** 6](#_Toc136956534)

[**brands** 6](#_Toc136956535)

[**countries** 7](#_Toc136956536)

[**price\_dependency\_boxplot** 7](#_Toc136956537)

[**price\_dependency\_scatter** 8](#_Toc136956538)

[**correlation** 9](#_Toc136956539)

[**kruskal\_fun** 9](#_Toc136956540)

[**correlation\_graphic** 10](#_Toc136956541)

[**hist\_fun** 11](#_Toc136956542)

[**Обращение к программе** 12](#_Toc136956543)

# **Введение**

Это руководство предназначено для разработчиков, которые желают создать мощное и удобное приложение, способное анализировать и предоставлять информацию о характеристиках автомобилей с использованием графиков, текстовых отчетов и встроенной базы данных автомобилей с возможностью фильтрации.

В мире автомобилей информация о характеристиках и особенностях различных моделей играет важную роль для потенциальных покупателей и автолюбителей. Однако найти и проанализировать эту информацию может быть трудоемким и сложным процессом. Ваше приложение призвано решить эту проблему, предоставляя пользователям удобный и понятный способ получения информации о характеристиках автомобилей.

Цель проекта «Приложение для анализа автомобилей» заключается в создании инновационного приложения, которое позволит пользователям легко и быстро получать доступ к разнообразным характеристикам автомобилей, анализировать их с помощью графиков и текстовых отчетов, а также использовать встроенную базу данных автомобилей для фильтрации и поиска конкретных моделей.

В итоге ваше приложение поможет пользователям легко и удобно получать информацию о характеристиках автомобилей, сравнивать различные модели и принимать информированные решения при покупке или анализе автомобилей. Мы надеемся, что данное руководство поможет вам в разработке приложения и реализации вашей идеи.

Приступим к созданию приложения для анализа характеристик автомобилей и облегчим процесс получения информации об автомобилях.

**Характеристики программы**

## **Описание необходимого окружения**

Для того чтобы приложение запустилось необходимо установить дистрибутив Anaconda и встроенные в неё библиотеки.

Anaconda — это популярный дистрибутив Python и R, который обеспечивает удобную среду для разработки и управления пакетами научных вычислений и анализа данных. Он включает в себя не только сам интерпретатор Python или R, но и множество научных библиотек, инструментов и пакетов, таких как NumPy, Pandas, Matplotlib и многие другие. Anaconda также предоставляет удобную систему управления пакетами с помощью Conda, которая позволяет легко устанавливать, обновлять и управлять зависимостями пакетов. Благодаря своей мощности и простоте использования, Anaconda стал популярным выбором для разработчиков и исследователей данных, позволяя им быстро начать работу с анализом данных и машинным обучением.

## **Установка и запуск**

1. Установить Anaconda на свой ПК;
2. Открыть поиск и ввести «Anaconda prompt», далее открыть терминал по результату поиска;
3. С помощью команды cd войти в директорию /work/scripts;
4. Запустить приложение, используя команду «python main.py».

## **Обращение к базе данных**

Программа обращается к базе данных с помощью библиотеки pandas. Файл, содержащий базу данных, читается программой pd.read\_excel(io=path\_to\_data, sheet\_name=0), где path\_to\_data содержит путь до базы в директории проекта work, sheet\_name=0 – параметр, который берет в этом .xlsx файле первый лист.

## **Структура приложения**

├───Data

│ Cars\_Python.xlsx

│

├───Graphics

│ boxplot.png

│ brands.png

│ correlation\_1.png

│ correlation\_2.png

│ countries.png

│ scatter.png

│

├───Library

│ │ help\_functions.py

│ │ main\_functions.py

│ │

│ └───\_\_pycache\_\_

│ help\_functions.cpython-39.pyc

│ main\_functions.cpython-39.pyc

│

├───Notes

│ Руководство пользователя.docx

│ Руководство разработчика.docx

│ ТЗ.docx

│

├───Output

│ correlation.xlsx

│ kruskal.txt

│

└───Scripts

main.py

# **Описание функций анализа базы данных**

В данном разделе будут описываться функции, их параметры и что следует ожидать на выходе.

Приложение предоставляет функции поиска, фильтрации и сортировки автомобилей в базе данных. Оно также позволяет визуализировать данные с помощью графиков и генерировать текстовые отчеты о корреляции количественных характеристик автомобилей. Эти функции помогают пользователям находить нужные автомобили, анализировать данные и принимать информированные решения при выборе автомобиля.

## **brands**

Обращение:

**brands**(excel, home)

Описание:

Данная функция строит диаграмму марок автомобилей от количества уникальных автомобилей, относящихся к этой марке. Функция не нуждается в дополнительных параметрах и строит график исходя из всех данных, находящихся в базе.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

## **countries**

Обращение:

**countries**(excel, home)

Описание:

Данная функция строит диаграмму стран от количество различных марок в этих странах. Функция не нуждается в дополнительных параметрах и строит график исходя из всех данных, находящихся в базе.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

## **price\_dependency\_boxplot**

Обращение:

**price\_dependency\_boxplot**(excel, arg, home)

Описание:

Данная функция создает график Бокса-Вискера: цена от характеристики автомобиля. Функция принимает 1 дополнительный параметр arg, то есть характеристику, по которой будет строиться зависимость.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* arg – тип: строка, описание: колонка в базе данных (характеристика автомобиля), по которой будет строиться зависимость от цены;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

## **price\_dependency\_scatter**

Обращение:

**price\_dependency\_scatter**(excel, arg, home)

Описание:

Данная функция создает график-разброс: цена от характеристики автомобиля. Функция принимает 1 дополнительный параметр arg, то есть характеристику, по которой будет строиться зависимость.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* arg – тип: строка, описание: колонка в базе данных (характеристика автомобиля), по которой будет строиться зависимость от цены;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

## **correlation**

Обращение:

**correlation**(excel, home)

Описание:

Данная функция создает текстовый отчёт, в котором содержится корреляция всех количественных параметров автомобилей, находящихся в базе данных. Функция не нуждается в дополнительных параметрах и строит график исходя из всех данных, находящихся в базе.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Текстовый отчёт, который сохраняется в папке output, с расширением .xlsx.

## **kruskal\_fun**

Обращение:

**kruskal\_fun**(excel, home, arg)

Описание:

Данная функция создает текстовый отчёт, в котором содержится анализ корреляции между количественной целевой переменной и качественной объясняющей, то есть ценой. Функция не нуждается в дополнительных параметрах и строит график исходя из всех данных, находящихся в базе.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work;
* arg – тип: строка, описание: колонка в базе данных (характеристика автомобиля) с количественными данными, по которой будет строиться зависимость от цены;

Выходные данные:

Текстовый отчёт, который сохраняется в папке output, с расширением .txt.

## **correlation\_graphic**

Обращение:

**correlation\_graphic**(excel, home)

Описание:

Данная функция создает графический отчёт, в котором содержится корреляция всех количественных параметров автомобилей, находящихся в базе данных. Функция не нуждается в дополнительных параметрах и строит график исходя из всех данных, находящихся в базе. Также функция использует дополнительные функции, которые находятся в файле help\_functions.py, взятые с сайта matplotlib, документация по библиотеке.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

## **hist\_fun**

Обращение:

**hist\_fun**(excel, home, arg, bins)

Описание:

Данная функция создает гистограмму частот количественных переменных. Принимает 2 дополнительных параметра.

Параметры:

* excel – тип: DataFrame, описание: база данных, прочитанная библиотекой pandas;
* home – тип: строка, описание: путь до директории work;
* arg – тип: строка, описание: колонка в базе данных (характеристика автомобиля) с количественными данными, по которой будет строиться частота;
* bins – тип: целое число, описание: количество дроблений на графике.

Выходные данные:

Графический отчет, построенный на основе базы данных. .png файл с данным графиком в папке graphics.

# **Обращение к программе**

Обращение к программе происходит при помощи интерфейса. Он реализован средствами библиотеки Tkinter. Интерфейс имеет 3 вкладки: база, анализ, настройки.

Во вкладке «База» реализован просмотр автомобилей по всей базе. Созданы фильтры для упрощения поиска нужного автомобиля по заданным критериям.

Во вкладке «Анализ» представлены функции анализа. Они подразделяются на графические и текстовые данные. Во избежание неверного ввода со стороны пользователя сделаны специальные выпадающие списки, чтобы пользователь выбрал среди них необходимый и не вводил название самостоятельно. Также реализован ползунок для выбора значения от 10 до 300. Графические и текстовые отчеты автоматически сохраняются в папку Graphics и Output, соответственно.

Во вкладке «Настройки» пользователь может выбрать темную либо светлую тему.