ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ КАЗЕННОЕ

ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

«КРАСНОДАРСКОЕ ПРЕЗИДЕНТСКОЕ КАДЕТСКОЕ УЧИЛИЩЕ»

**ИТОГОВЫЙ ИНДИВИДУАЛЬНЫЙ ПРОЕКТ**

за курс основного общего образования

**Тема:  
«Создание базы данных»**

Выполнил Фомин И.Н.,

обучающийся 10 класса «Ж»

Руководитель проекта: Радь Ю.О.,

преподаватель информатики

Работа допущена к защите: «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_\_\_г.

Подпись руководителя проекта: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_(Радь Ю.О.)

**Краснодар**

**2022 г.**

**ПАСПОРТ**

**итогового индивидуального проекта**

* Создание базы данных
* Цель: Создание базы данных и её анализ.

Задачи:

* Перевести базу данных из типа json в тип SQLite.
* Реализовать систему простых запросов к базе данных и ответов с неё через консоль.
* Провести исследование информации, содержащийся в базе данных, с помощью системы запросов и ответов.
* Получить знания и умения.
* Создать презентацию, показывающую реализацию моей цели и её итог.
* Подготовить полученные материалы к защите проекта.
* Защитить проект.

Исполнитель проекта: Фомин Иван Николаевич, обучающийся 10 «Ж» класса

* Руководитель (тьютор) проекта: Радь Ю.О., преподаватель информатики
* Сроки реализации проекта: с 01.09.2022 до 01.02.2023 г.
* Тип проекта:

По доминирующей

в проекте деятельностиРазработка и исследованиеПо предметно-содержательной областиИнформатикаПо количеству участников проектаИндивидуальныйПо широте охвата содержанияПредметныйПо характеру контактов (степени охвата)В рамках информации из социальной сети

7. Образовательная область, с которой связано содержание проекта: информатика, программирование.

8. Методы, методики, технологии, использованные в работе над проектом:

Высокоуровневый язык программирования - Python, текстовый формат обмена данными – json, библиотеки: SQLite3, argparse, json, matplotlib.

9. Результат (продукт) проекта и форма его представления: база данных с программным кодом для её изучения и презентация вместе с исследованиями содержащийся в базе информации.

10. Образовательные и культурно-просветительские учреждения, на базе которых выполнялся проект: база училища.

11. Источники информации, использованные автором в процессе выполнения проекта: выложенная в открытом доступе в сети интернет информация о части пользователей социальной сети twitter, информация о используемых библиотеках https://github.com/, книга «Data Science» от Joel Grus.

12. Использованное оборудование: компьютер.

**Введение**

Данный проект является как учебным, изучение новых библиотек и различных способов парсинга (систематизации данных), так и информационным, анализ большого объёма данных с будущим построением графиков по интересующей нас функции. Данная работа имеет большой простор для будущей доработки и, конечно, может применяться и для других сфер программирования.

**Актуальность:** В последние десятилетия стали очень популярны социальные сети. Миллионы людей каждый день заходят в свои приложения для того, чтобы узнать: что же произошло за последнее время. Мой проект оценивает данную активность за различные промежутки времени, оценивая количество регистраций и общее число пользователей сети. Это также поможет Российскому правительству знать, когда люди больше отдыхают, а когда заняты работой.

**Гипотеза:** основана на предположении о том, что возможно узнать, когда люди свободны, и когда они не могут зайти в социальную сеть.

**Цель:** научиться создавать свою базу данных, проводить её анализ и получать из этого информацию об активности людей.

**Задачи:**

* Ознакомиться с информацией по теме проекта.
* Систематизировать полученную информацию.
* Создать свою базу данных на основе полученной информации.
* Проанализировать большой объём информации на основе созданной базы.
* Сделать удобный способ принятия запроса и вывода на него ответа.
* Сделать выводы и создать презентацию на основании проведенного исследования.
* Подготовить полученные материалы к защите проекта.
* Защитить проект.

**Объект анализа:** Созданная база на 5 миллионов человек.

**Методы анализа:** аналитический и сравнительный.

**Практическая значимость**: оценка активности людей той или иной сети, с выявлением наибольшей, наименьшей и средней активностей.

**Основная часть:**

**1. Создание базы данных**:

**1.1. Что такое база данных:**

Для создания базы, сначала разберёмся - что это такое? База данных – совокупность данных, хранящихся и упорядоченных в соответствии с определённой структурой. Структура, в свою очередь, определяет то, каким образом данные будут располагаться и как к ним будет предоставляться доступ. Проще говоря — обычная таблица с возможностью настройки условий на столбцы.

**1.1. Создание и настройка пустой базы:**

Для создания пустой базы я воспользовался простеньким браузером баз данных на SQL, он находиться в открытом доступе в интернете и в целом имеется множество аналогов, созданных для упрощения данного процесса. Упрощением пользуюсь из-за того, что помощью кода неудобно создавать и настраивать столбцы, тем более, что это понадобится лишь единожды — для создания и настройки столбцов базы.

**2. Заполнение созданный базы:**

**2.1. Что такое СУБД и SQL:**

Файл, найденный в интернете, имеет расширение json, т.е текстовый формат обмена данными, основанный на JavaScript. Файл был плохо структурирован, что мешало анализировать данные. Для выхода из этой ситуации создадим базу данных в удобном для анализа расширении db (data base) и наполним её данными из найденного файла, параллельно структурируя данные. Структурирование и управление базы происходит с помощью СУБД (системы управления базой данных), она позволяет создавать, удалять, изменять и делать любые манипуляции с данными. Для отправления запросов в СУБД будет использоваться отдельный язык - SQL, язык запросов, предоставляющий возможность манипулировать информацией в базе данных.

**2.2. Обход ограничений ресурсов компьютера:**

Подключившись к пустой базе в своей программе открываем найденный файл в оперативной памяти и начинаем её парсить в ранее открытое хранилище, но продлевается данный процесс не долго, файл, выгруженный в память, весит так много, что занимает все ресурсы системы и тормозит весь компьютер вплоть до системной остановки процесса или отключения компьютера. Для избежания данной проблемы выгружаем в оперативную память только одну строку исходного файла, достаём из неё нужные данные, сохраняем в нашу базу и освобождаем память для нового цикла, пока не будут перенесены все строки найденного в интернете файла.

**2.3. Решение проблемных мест исходного файла:**

В процессе заполнения новой базы оказалось, что в исходном файле иногда столбцы с email отсутствовали, а вместо них был столбец с номером телефона. Выходом стала обычная проверка на наличие или отсутствие того или иного столбца перед отправлением запроса. Также находились строки, заполненные арабским в несколько десятков символов. Данных строк было четыре из пяти миллионов, так что самым простым решением был их пропуск.

**3. Поиск информации внутри созданной базы:**

**3.1 Количество регистраций за год, день или час:**

Итак, у меня есть систематизированная база данных, как теперь на достать из неё информацию? Изначально я хотел отправлять запросы в нашу базу построчно, но ещё немного изучив виды запросов SQL нашёл функцию, позволяющую делать запрос на поиск по всей базе, при том, что он будет построчным и не станет сильно использовать ресурсы компьютера. Поиск производился только по одному столбцу, который содержал в себе полную дату регистрации пользователя, и считал только те строки, которые содержали нужный год, день или час. Теперь я мог одним запросом найти количество пользователей в нужную мне дату.

**3.2 Количество регистраций за нужный промежуток:**

Чтобы получить ещё больше информации логичнее всего было рассматривать нужный нам промежуток и строить график по частоте регистраций. Простым способом будет сделать цикл, перебирающий выбранные года, дни или часы, и отправляющий по очереди по одному запросу. Теперь мы просто сохраняем ответы и строим по ним график с помощью простой библиотеки — matplotlib, она требует на вход данные точек и названия осей и строит по ним график.

**4. Простая реализация удобного способа отправления запросов:**

**4.1. Что такое argparse:**

Наша программа успешно выполняет свою работу, но для её использования нам требуется изменять файл либо использовать отдельно файлы для нужной функции. Также требуется запуск файла из специальной среды. Для облегчения данного процесса нужно реализовать простой и удобный способ отправки запросов с возможностью выбора функции. В этом мне помогла библиотека argparse, представляющая из себя набор функций для принятия запросов из командной строки консоли.

**4.2. Создание шаблона запроса:**

Идея заключается в том, чтобы одной строкой в консоли можно было получить нужный нам график. Представим, что мы не знаем, что может выполнять наша программа, для этого мы отправляем простую команду помощи, выводящую варианты функций и возможные для исследования промежутки. Теперь мы вводим нужную нам функцию и промежуток, отправляем запрос в нашу программу и после обработки получаем ответ. Для реализации данной программы я выношу все действия в функции и ,при получении запроса, условиями отправляю данные в нужную часть программы.

**4.2**

**Заключение**

После данной работы я глубже разобрался в создании базы данных и её настройке, изучил библиотеки json, sqlite3, matplotlib и argparse, они многофункциональны и могут использоваться во многих сферах информатики. В целом очень много сил было потрачено на изучение материалов, из-за чего осталось много простора для будущего развития проекта. Например: можно рассматривать конкретные даты регистраций, находить наибольшие всплески и отслеживать регистрации не настоящих пользователей, которые могут сильно портить будущую статистику приложения, благодаря данной функции возможно будет пресекать данную деятельность блокировкой данных аккаунтов и нахождением создателя данной проблемы.

В следующем году я планирую развить дальше эту тему. А сейчас я считаю, что мой проект вполне удался.

**Список литературы**

Книги:

1. Джоэл Грас **— «**Data Science**»**

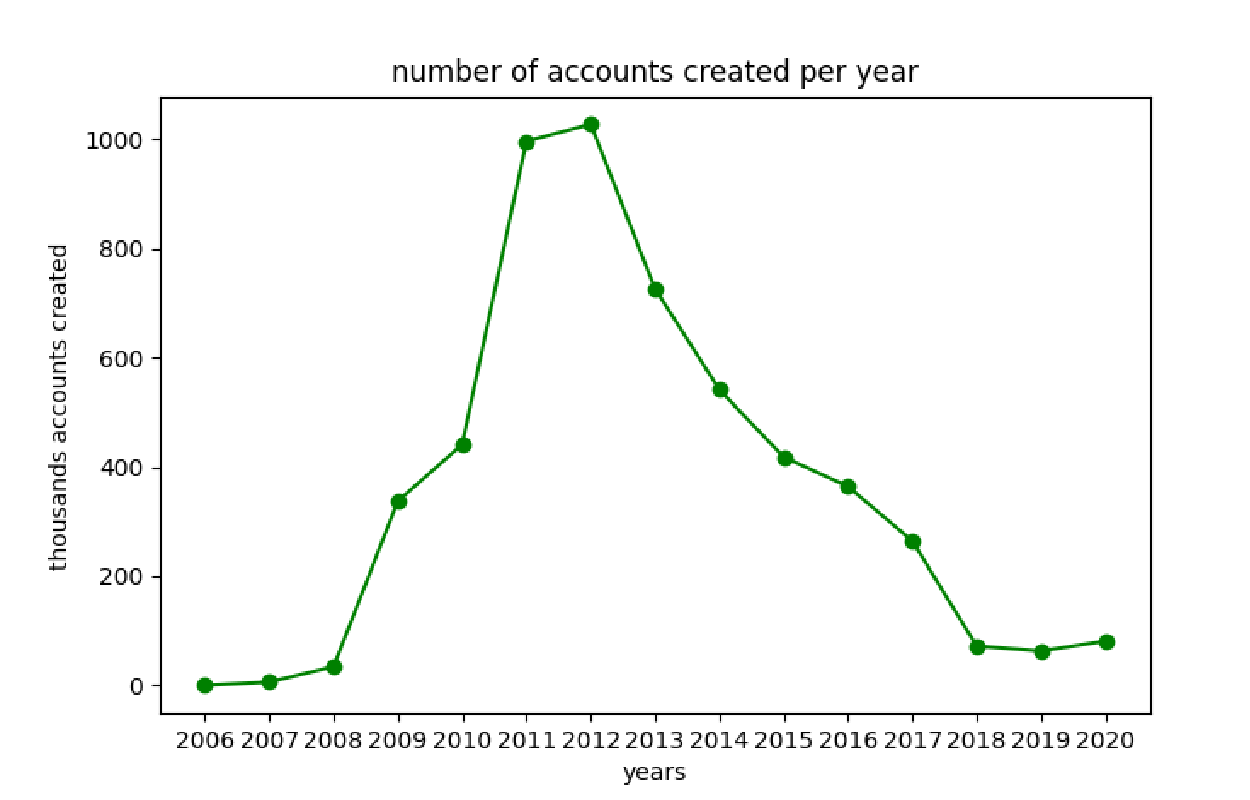
Интернет источники:

* docs.python.org
* docs.python.org/3/library/argparse.html
* docs.python.org/3/library/sqlite3.html
* matplotlib.org
* github.com

**Приложения**

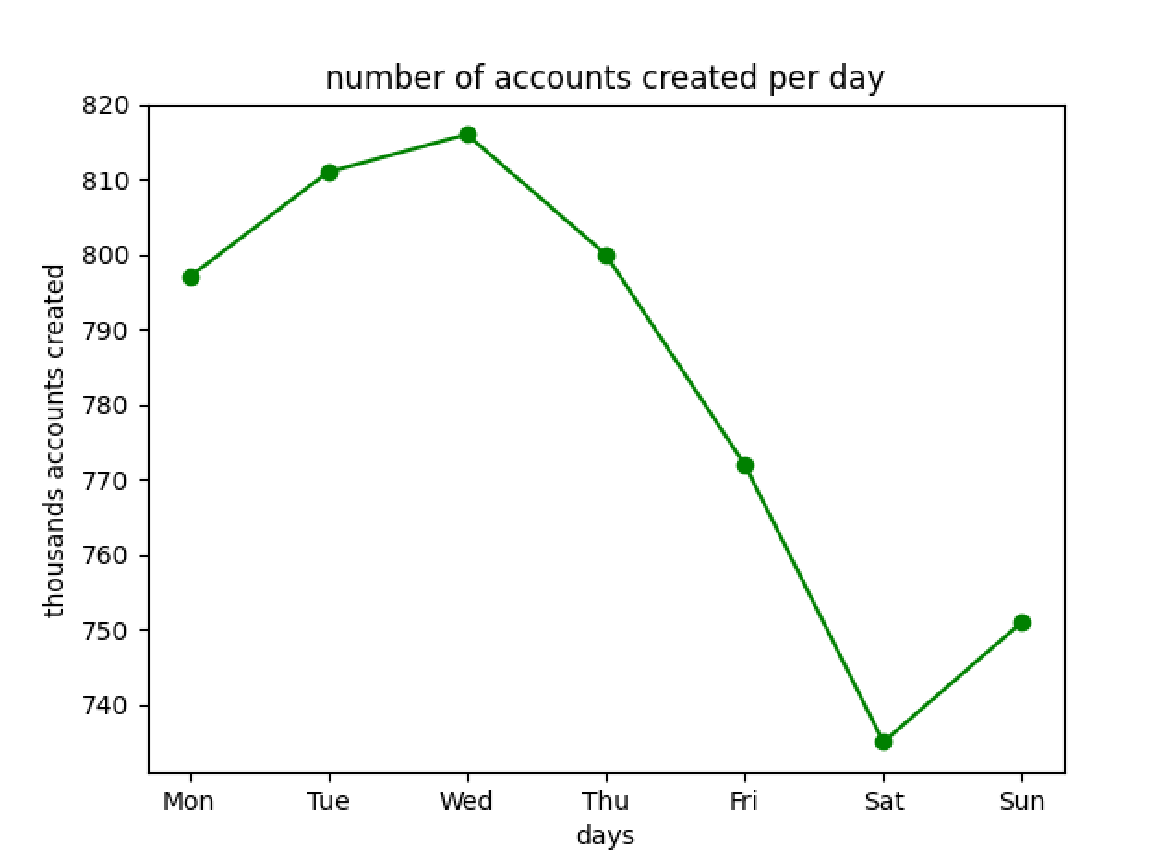
**Приложение 1**

**График регистраций по годам:**



**Приложение 2**

**График регистраций по дням недели:**



**Приложение 3**

**График регистраций по часам:**

