Государственное бюджетное профессиональное
образовательное учреждение Московской области
«Физико-технический колледж»

Отчёт по кейсу «Самолёт»:

Работу выполнил:
Студент группы № ИСП-22
Серый Александр

Введение:

В данном отчете рассматриваются выводы по первому интенсиву по сбору и анализу данных о продаваемых квартирах в Москве и Московской области. Для выполнения работы использовался язык програмирования Python.

Цель:

Собрать данные о продаваемых квартирах в Москве и Московской области, провести работу над ними, проанализировать и заполнить пропуски для дальнейших задач.

Задачи:

1. Собрать данные используя открытые источники, например такие как Циан или ДомКлик
2. Обработка собранной информации, очистка от ненужной или неверной информации
    a. Визуализация в ходе выполнения обработки информации.

Ход выполнения работы.

Первый этап — это сбор данных, который был выполнен с помощью языка программирования Python и библиотеки CianParser. В процессе работы пришлось изменить некоторые части кода библиотеки для более корректного вывода и уменьшения количества пропусков при сборе информации. Сбор данных производился по городам Москвы и Московской области с указанием города, типа продажи и количества комнат.

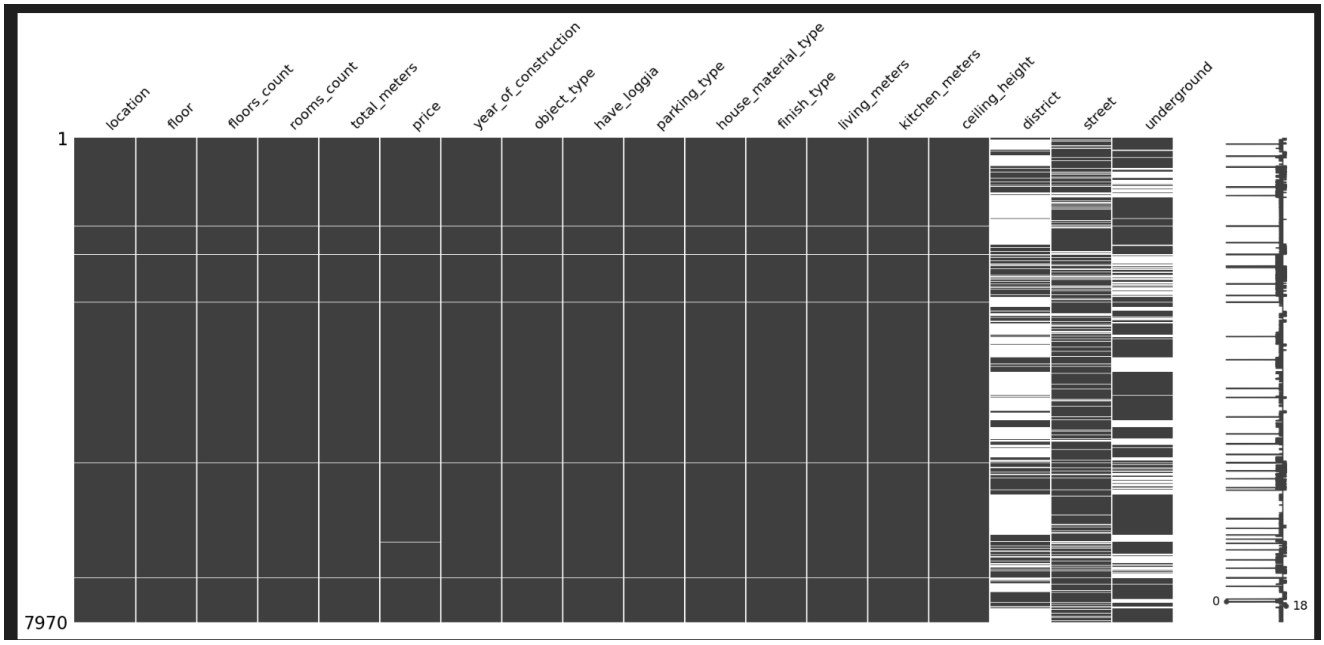
```
import cianparser
parser = cianparser.CianParser(location="Одинцово")# здесь указывается город для сбора информации
# переменная sale не менялась нам нужен только такой тип
data = parser.get_flats(deal_type="sale", rooms=(1), additional_settings={"start_page":1, "end_page": 54}, with_extra_data=True, with_saving_csv=True)
#rooms=(1) вместо 1 можно указать количество комнат для сбора
```

Второй этап — это обработка и анализ собранных данных. Вначале мы смотрим, все ли мы верно сделали и данные отображаются.

	author	author_type	url	location	deal_type	accommodation_type	floor	floors_count	rooms_count	total_meters	...	finish_type	living_meters	kitchen_meters	phone	ceiling_height	district	street	house_number	underground	residential_complex
0	ANT Development	developer	https://www.cian.ru/sale/flat/303519398/	Москва	sale	flat	11	13	5	265.6	...	Чистовая	-1	-1	74951346248	3 м	Дорогомилово	NaN	3к1	Парк Победы	Виктори Парк Резиденсез ЖК
1	ID 18174647	realtor	https://www.cian.ru/sale/flat/301450189/	Москва	sale	flat	26	31	5	246.7	...	-1	140,9 м²	28 м²	79166462390	3,1 м	Очаково-Матвеевское	Нежинская	1к1	Давыдково	Кутузовская Ривьера
2	Monumental Group	real_estate_agent	https://www.cian.ru/sale/flat/306835123/	Москва	sale	flat	4	8	5	117.0	...	-1	-1	10 м²	79672086536	3,2 м	Пресненский	Большая Садовая	3С1	Маяковская	NaN
3	Главстрой	developer	https://www.cian.ru/sale/flat/296921235/	Москва	sale	flat	24	24	5	172.5	...	Без отделки	104,6 м²	10,5 м²	74951379308	3,2 м	Филевский парк	Береговой проезд	2	Фили	Береговой-2
4	Penta	real_estate_agent	https://www.cian.ru/sale/flat/307983948/	Москва	sale	flat	3	9	5	234.8	...	-1	140 м²	23 м²	79663231016	-1	Тверской	Краснопролетарская	7	Новослободская	Ласточкино гнездо

5 rows × 27 columns

Далее мы убираем часть лишней информации из наших данных. Затем выводим количество отсутствующей информации визуально



После этого убираем полностью пустые строки и повторяющуюся информацию, а затем выводим количество пустот в столбцах
После этого удаляем столбцы, в которых более 70% значений являются пропусками.

	Column	Missing Percentage
0	location	0.000000
1	floor	0.000000
2	floors_count	0.000000
3	rooms_count	0.000000
4	total_meters	0.000000
5	price	0.000000
6	year_of_construction	17.166596
7	object_type	0.000000
8	have_loggia	0.000000
9	parking_type	0.000000
10	house_material_type	82.030700
11	finish_type	74.820448
12	living_meters	22.236305
13	kitchen_meters	16.406140
14	ceiling_height	35.093649
15	district	60.132376
16	street	20.729475
17	underground	30.770314

Далее убираем еще в некоторых столбцах пропуски заполнением средним значение где это возможно

Выводит статистическую сводку числовых данных

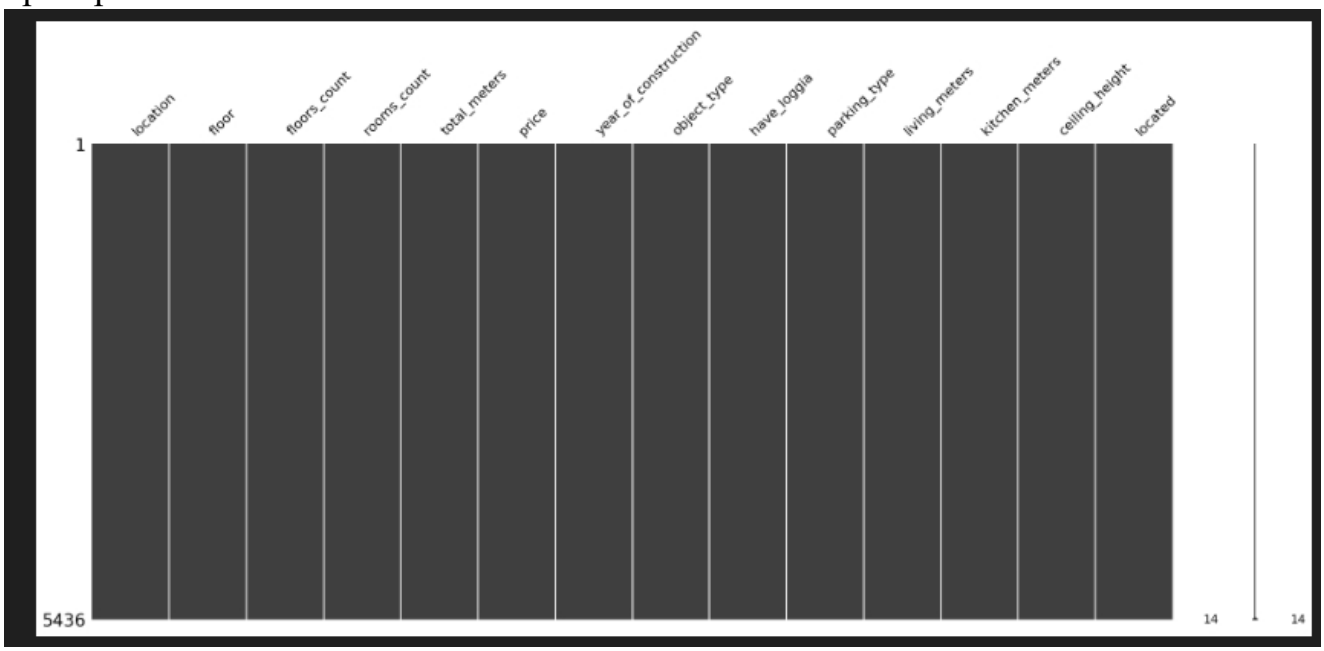
```
             floor  floors_count  rooms_count  total_meters         price  \
count  7100.000000   7100.000000  7100.000000   7100.000000  7.100000e+03
mean      7.410704     13.929577     1.731972     51.669144  1.500228e+07
std       6.682045      9.152099     0.847005     30.561712  4.766439e+07
min       1.000000      1.000000     1.000000     13.000000  8.300000e+05
25%       3.000000      7.000000     1.000000     35.000000  5.400000e+06
50%       5.000000     13.000000     2.000000     44.500000  7.757476e+06
75%      10.000000     18.000000     2.000000     60.000000  1.100000e+07
max      82.000000     97.000000     5.000000    590.300000  2.361200e+09

       ceiling_height
count     7100.000000
mean         2.828529
std          0.684942
min          0.000000
25%          2.700000
50%          2.828529
75%          2.828529
max         52.000000
```

После чего ставим ограничения, чтобы не было недостоверной информации

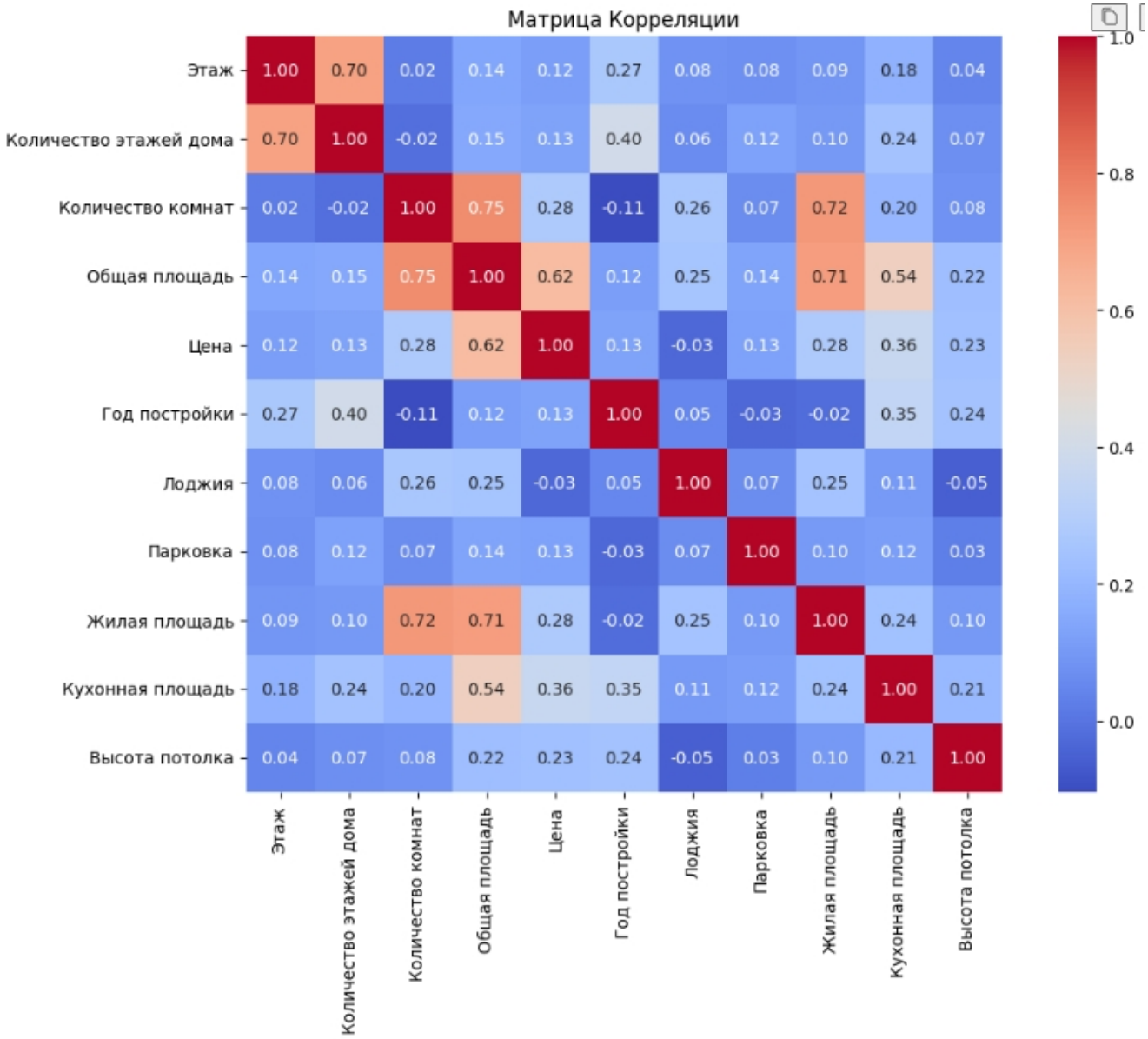
```
    'floor': (1, 97),
    'floors_count': (1, 95),
    'rooms_count': (1, 6),
    'total_meters': (8, 260),
    'price': (2000000, 800000000),
    'ceiling_height': (2.0, 6.0),
```

Далее удаляются все возможные пропуски, и выводится визуально для проверки.



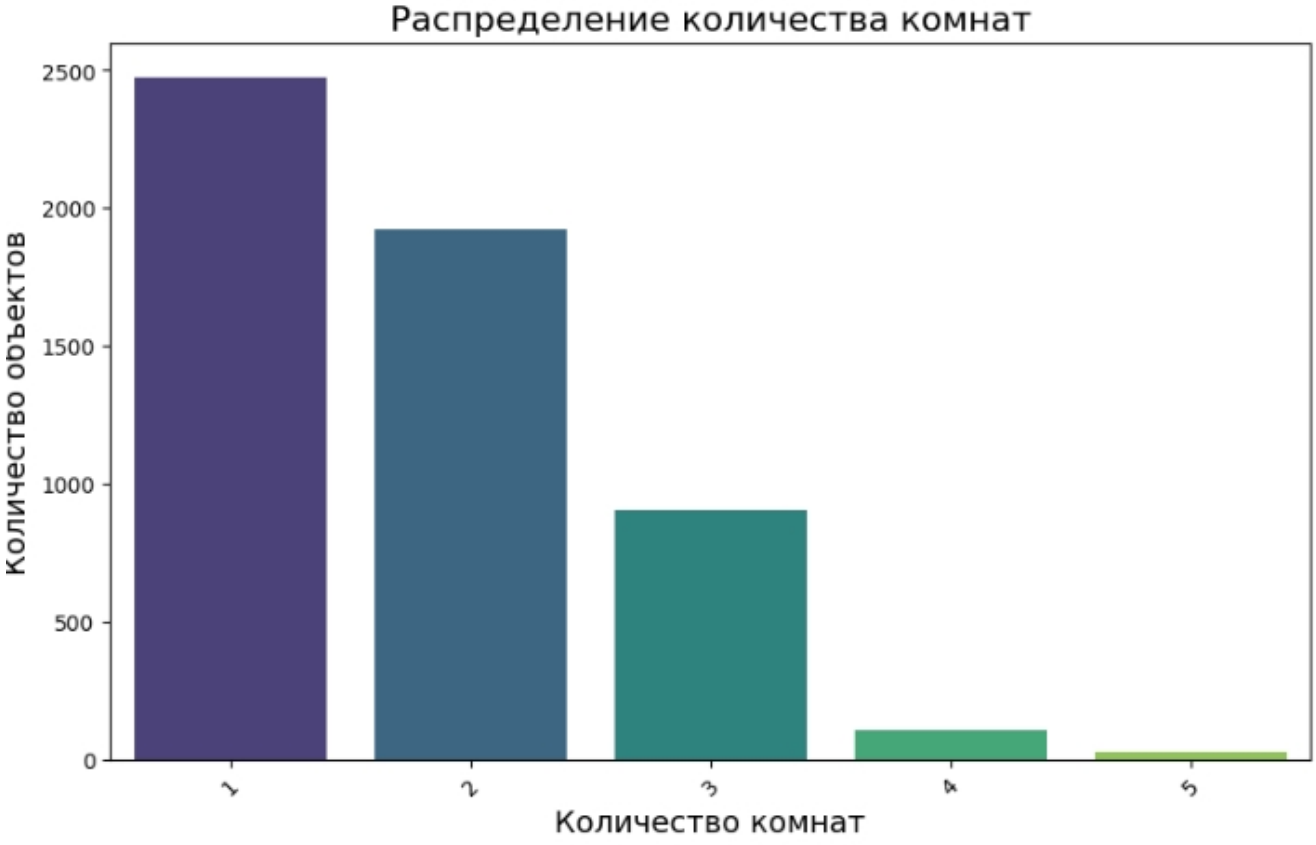
Окончательная часть включает просмотр зависимостей и исправление оставшейся некорректной информации.

Вывод таблицы зависимостей

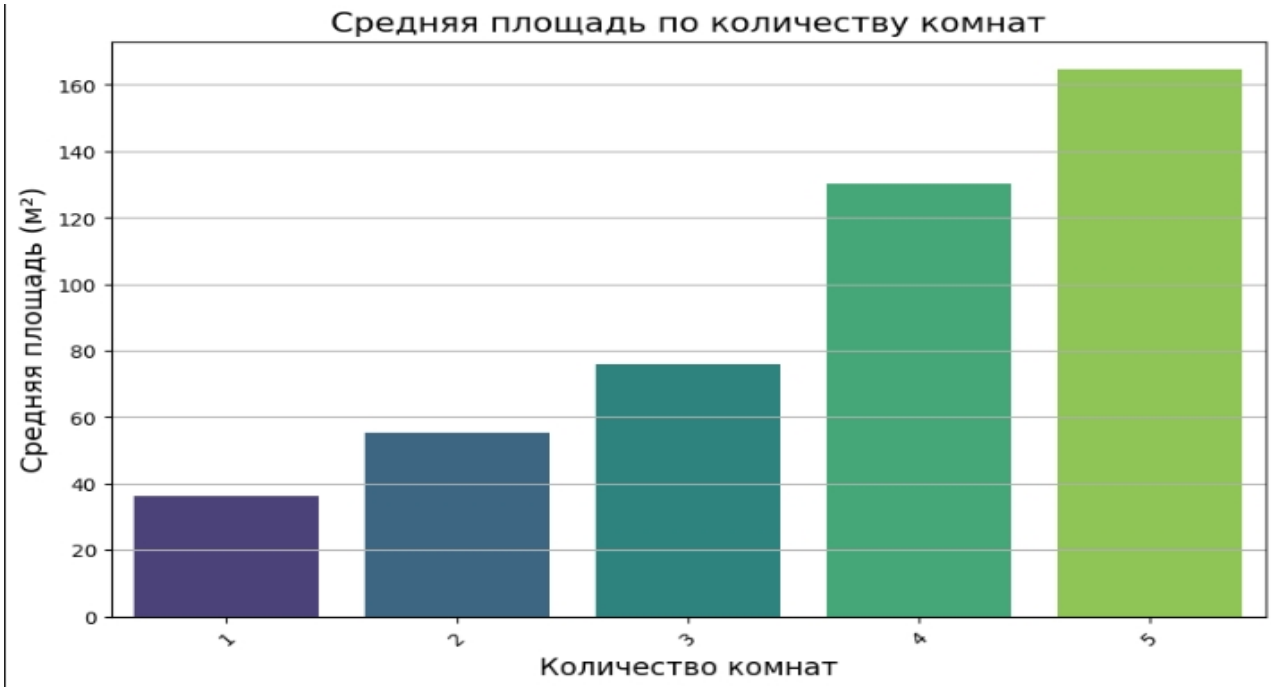


Отображение зависимости цены от года

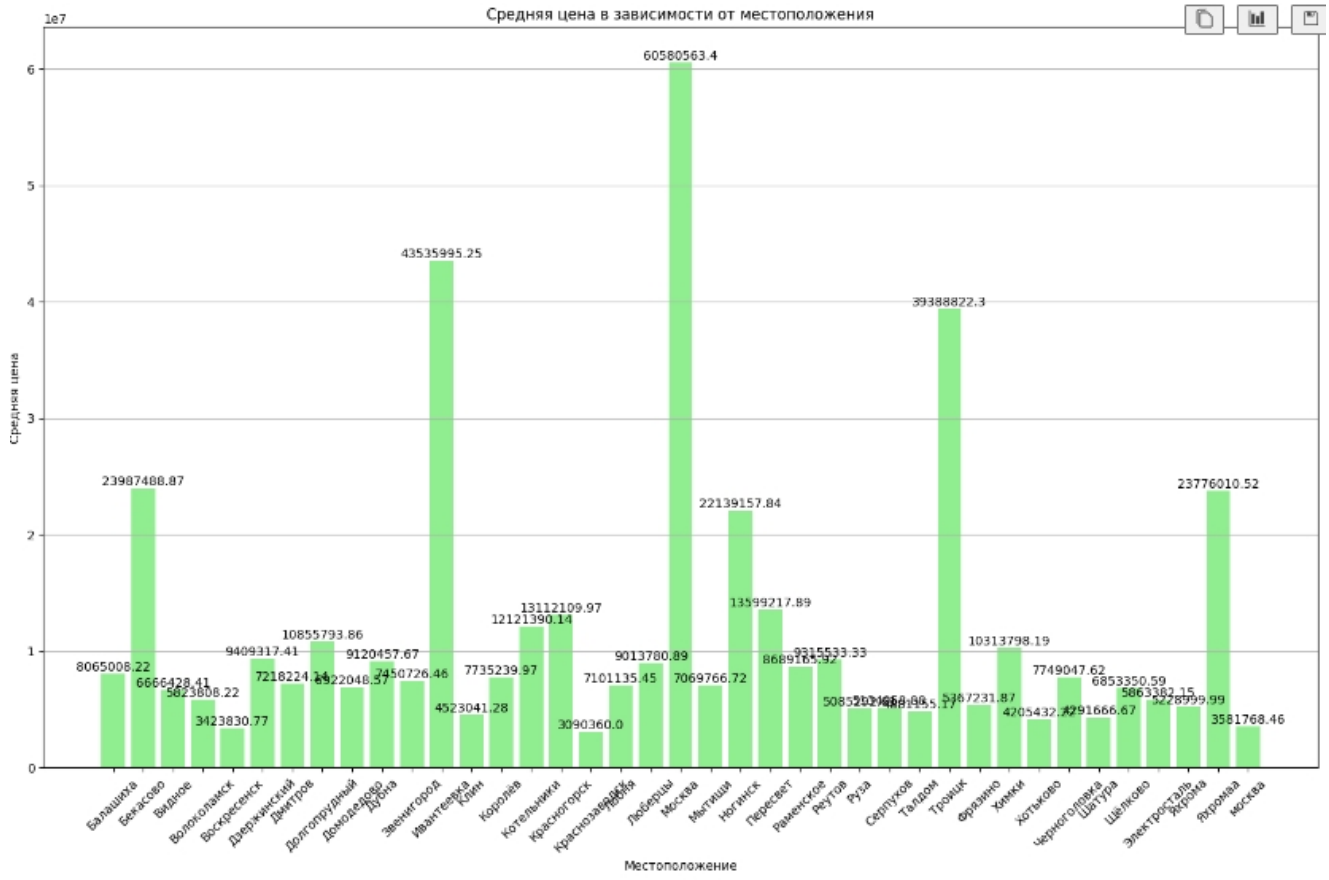
Количество информации после обработки по количеству комнат

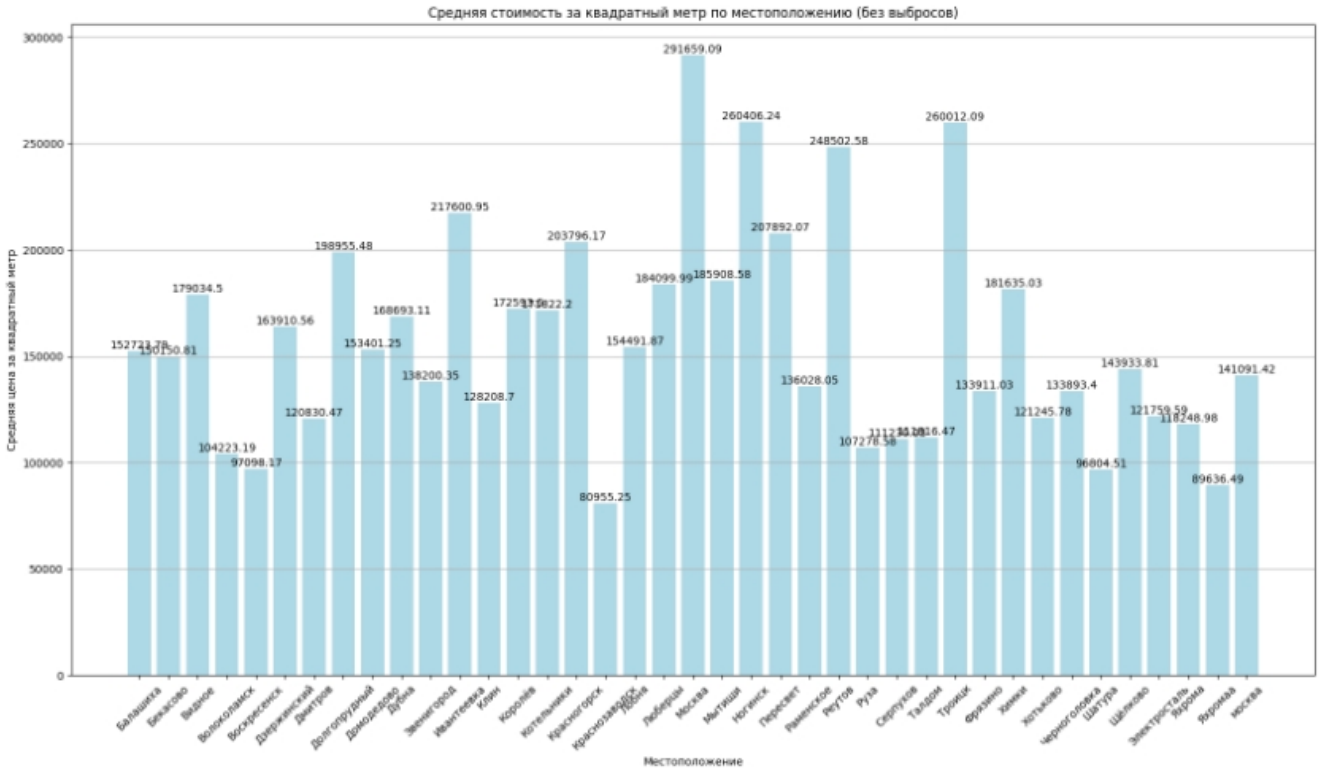


Зависимость площади от количества комнат.

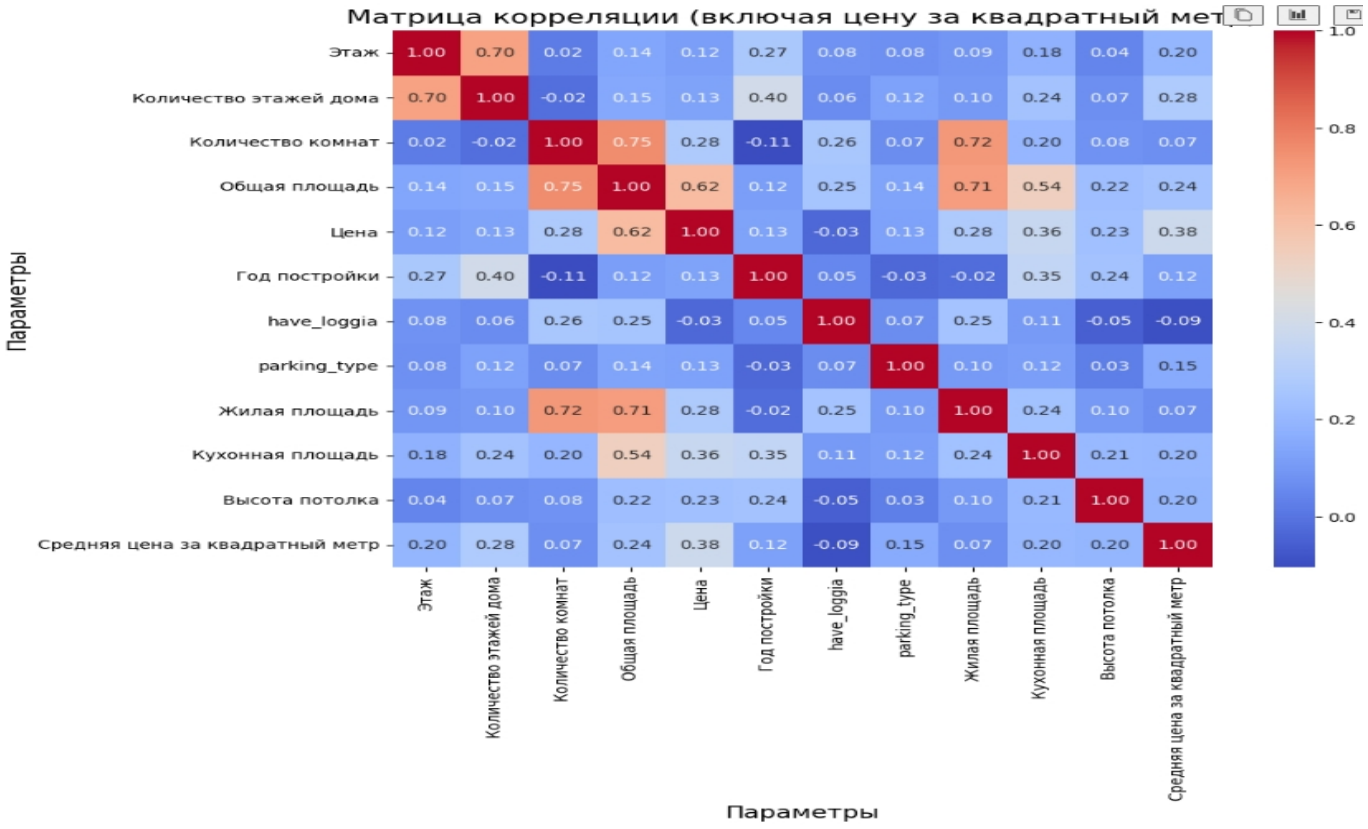


Средняя цена в (рублях) зависамости от города





Средняя цена на мтрах квадратный в разных городах
Также отображение зависимостей с стоимостью за квадратный метр.



Ссылки на источники сбора информации для проекта.
https://www.cian.ru
Ссылки на сам проект.
Выполнявший работу https://github.com/Sr123Saha
Ссылка на выполненную работу https://github.com/Sr123Saha/1intensive1
ссылка на работу в googl colab
https://colab.research.google.com/drive/1z1g0W5xrtg4gZ5MXBhbfsrJbOrVXsrB6