

Департамент образования и науки города Москвы
Государственное автономное образовательное учреждение высшего
образования города Москвы
«Московский городской педагогический университет»
Институт цифрового образования
Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

Инструменты для обработки и хранения больших данных

Отчет

Тема:

Визуализация данных из CSV файла в YandexDataLens

Выполнила: Олифир А. А., группа: АДЭУ-201

Преподаватель: Босенко Т. М.

Москва

2023

ColabResearch:

https://colab.research.google.com/drive/1aaTrpEIDMdg2qqyEcevkuyWfChSOp_BF?usp=sharing

YandexDataLens: <https://datalens.yandex.ru/wxn64jr4r5r8m-kvartiry-moskvy-olifir>

Для визуализации данных я использовала данные по ценам квартир в Москве (2 вариант).

"price", "totsp", "livesp", "kitsp", "dist", "metrdist", "walk", "brick", "floor", "code"									
1,81,58,40,6,12.5,7,1,1,1,3									
2,75,44,28,6,13.5,7,1,0,1,6									
3,128,70,42,6,14.5,3,1,1,1,3									
4,95,61,37,6,13.5,7,1,0,1,1									
5,330,104,60,11,10.5,7,0,1,1,3									
6,137,76,50,9,11,7,1,1,1,8									
7,98,59,39,6,7.5,10,0,0,1,8									
8,88,55,36,6,9.5,1,1,0,4									
9,225,80,56,9,9.5,1,1,1,3									
10,140,86,51,10,12.7,10,1,0,1,5									
11,132,96,56,12,10.5,7,0,0,1,3									
12,115,77,50,9,3.5,7,1,0,0,7									
13,142,70,47,8,7.5,5,1,0,1,3									
14,84,63,44,7,14.5,10,0,0,1,3									
15,100,66,43,7,12,7,0,0,1,7									
16,110,75,50,10,13.5,10,0,0,1,6									
17,170,87,58,8,7.5,7,1,1,1,3									
18,175,79,50,9,4.5,1,1,1,0,8									
19,96,60,43,6,12,15,1,0,1,7									

Нужно было немного преобразовать данные, для этого я воспользовалась ColabResearch.

Импортируем нужные библиотеки, выгружаем файл. Также из Kaggle я взяла информацию о данных.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

[2] from google.colab import files

uploaded = files.upload()

Выбрать файлы flats_moscow (2).csv
• flats_moscow (2).csv(text/csv) - 66976 bytes, last modified: 12.02.2023 - 100% done
Saving flats_moscow (2).csv to flats_moscow (2).csv

"""
price: цена квартиры в $1000 / the price of a flat in 1000$;
totsp: общая площадь квартиры, кв.м. / total square of a flat, m2;
livesp: жилая площадь квартиры, кв.м. / living square of a flat, m2;
kitsp: площадь кухни, кв.м. / kitchen square, m2;
dist: расстояние от центра в км. / distance to the city center, km;
metrdist: расстояние до метро в минутах / distance to the nearest metro station, min;
walk: 1 - пешком от метро, 0 - на транспорте / 1 - walk to metro, 0 - using transport;
brick: 1 - кирпичный, монолит ж/б, 0 - другой / 1 - brick, monolithic house, 0 - anothers;
floor: 1 - этаж кроме первого и последнего, 0 - иначе / 0 - the first or the last floor, 1 - anothers;
code: число от 1 до 8, обозначающее район города / number from 1 to 8 of the city area:
1. Север, около Калужско-Рижской линии метро / North of the city, around Kaluzhsko-Rizhskaya metro line;
2. Север, около Серпуховско-Тимирязевской линии метро / North of the city, around Serpukhovsko-Timiryazevskaya metro line;
3. Северо-запад, около Замоскворецкой линии метро / North-West, around Zamoskvoretskaya metro line;
4. Северо-запад, около Таганско-Краснопресненской линии метро / North-West, around Tagansko-Krasnopresnenskaya metro line;
5. Юго-восток, около Люблинской линии метро / South-East, around Lyublinskaya metro line;
6. Юго-восток, около Таганско-Краснопресненской линии метро / South-East, around Tagansko-Krasnopresnenskaya metro line;
7. Восток, около Калининской линии метро / East, around Kalininskaya metro line;
8. Восток, около Арбатско-Покровской линии метро / East, around Arbatko-Pokrovskaya metro line.
```

Датафрейм:

	price	totsp	livesp	kitsp	dist	metrdist	walk	brick	floor	code
0	81	58	40	6.0	12.5	7	1	1	1	3
1	75	44	28	6.0	13.5	7	1	0	1	6
2	128	70	42	6.0	14.5	3	1	1	1	3
3	95	61	37	6.0	13.5	7	1	0	1	1
4	330	104	60	11.0	10.5	7	0	1	1	3
...
2035	110	77	45	10.0	12.0	5	0	0	1	5
2036	95	60	43	6.0	9.0	5	0	0	1	4
2037	95	60	46	5.0	10.5	5	1	0	1	7
2038	129	76	48	10.0	12.5	5	0	0	1	3
2039	103	64	45	7.0	15.5	5	1	0	1	1

2040 rows × 10 columns

```
#Сколько процентов данных с пустыми значениями
fl.isna().mean()
```

```
price      0.0
totsp      0.0
livesp     0.0
kitsp      0.0
dist       0.0
metrdist   0.0
walk       0.0
brick      0.0
floor      0.0
code       0.0
dtype: float64
```

Таким образом видно, что пустые значения в фрейме отсутствуют.

```
[10] #Просматриваем типы данных
fl.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2040 entries, 0 to 2039
Data columns (total 10 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   price       2040 non-null   int64
1   totsp       2040 non-null   int64
2   livesp      2040 non-null   int64
3   kitsp       2040 non-null   float64
4   dist        2040 non-null   float64
5   metrdist    2040 non-null   int64
6   walk        2040 non-null   int64
7   brick       2040 non-null   int64
8   floor       2040 non-null   int64
9   code        2040 non-null   int64
dtypes: float64(2), int64(8)
memory usage: 175.3 KB
```

У площади кухни и расстояния от центра значения с плавающей запятой.

```
#Изменение типов данных
fl['kitsp'] = fl['kitsp'].astype(int)
fl['dist'] = fl['dist'].astype(int)
```

Остальные манипуляции проводились с заменой данных в некоторых столбцах для более удобного использования в дальнейшем. Вот пример на основе данных столбца “walk”:

Теперь заменяем значения, где это требуется.

```
[21] f1['brick'].unique()
array([1, 0])
```

```
[22] def brick(x):
      if x == 1:
          return 'Кирпичный, монолит ж/б'
      if x == 0:
          return 'Другой'
```

```
[24] f1['kirpich'] = f1['brick'].apply(brick)
```

floor	code	walking	kirpich
1	3	Пешком от метро	Кирпичный, монолит ж/б
1	6	Пешком от метро	Другой
1	3	Пешком от метро	Кирпичный, монолит ж/б
1	1	Пешком от метро	Другой
1	3	На транспорте	Кирпичный, монолит ж/б
...
1	5	На транспорте	Другой
1	4	На транспорте	Другой
1	7	Пешком от метро	Другой
1	3	На транспорте	Другой
1	1	Пешком от метро	Другой

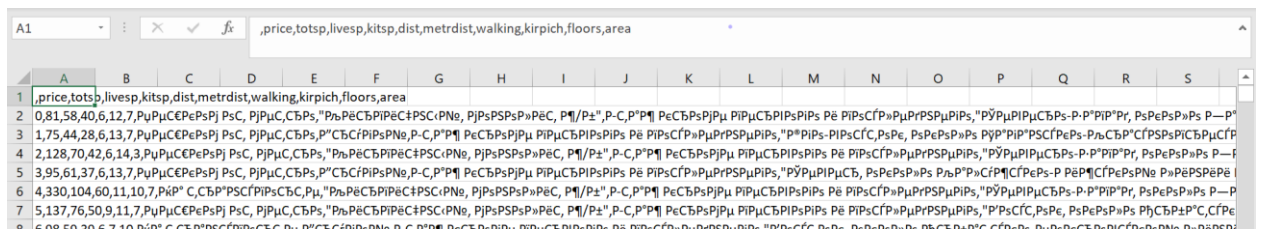
Итоговый датафрейм и его выгрузка в формате csv:

```
[44] f1=f1.drop('code', axis=1)
      f1.head()
```

	price	totsp	livesp	kitsp	dist	metrdist	walking	kirpich	floors	area
0	81	58	40	6	12	7	Пешком от метро	Кирпичный, монолит ж/б	Этаж кроме первого и последнего	Северо-запад, около Замоскворецкой линии метро
1	75	44	28	6	13	7	Пешком от метро	Другой	Этаж кроме первого и последнего	Юго-восток, около Таганско-Краснопресненской л...
2	128	70	42	6	14	3	Пешком от метро	Кирпичный, монолит ж/б	Этаж кроме первого и последнего	Северо-запад, около Замоскворецкой линии метро
3	95	61	37	6	13	7	Пешком от метро	Другой	Этаж кроме первого и последнего	Север, около Калужско-Рижской линии метро
4	330	104	60	11	10	7	На транспорте	Кирпичный, монолит ж/б	Этаж кроме первого и последнего	Северо-запад, около Замоскворецкой линии метро

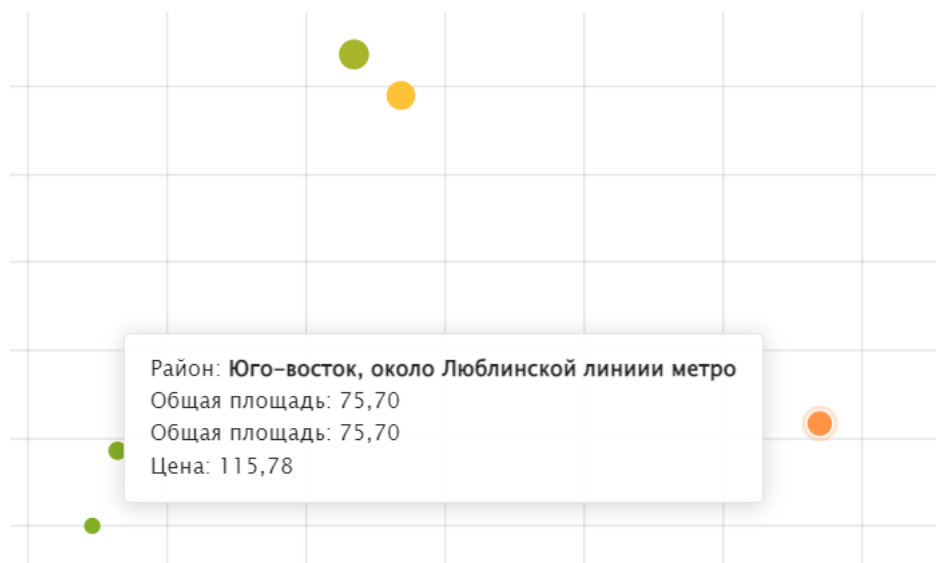
```
[48] from google.colab import files
      f1.to_csv('flat.csv')
      files.download('flat.csv')
```

Как он выглядит в Excel:

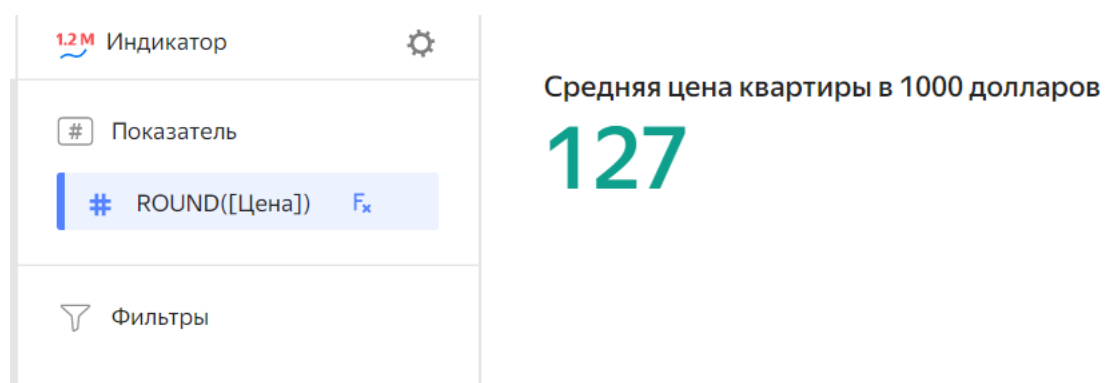


Один из чартов показывает зависимость цены и площади в восьми районах Москвы.

1. Точечная диаграмма с осями X (общая площадь квартиры) и Y(цена квартиры), точками-районами.
2. Цвета на диаграмме показывают соотношение площади: от зелёного до красного.
3. Размер точек показывает цену квартиры: от меньшей к большей.
4. Чтобы проанализировать какой-либо район, нужно навести курсор на точку. Например, самыми выгодными будут квартиры в районе Юго-востока Люблинской линии метро. Там средние площади и невысокие цены.



Еще одним видом чартов является индикатор. Например, средняя цена квартиры в 1000 долларов:



Здесь можно менять цвета и размер или написать формулу, соответствующую той задаче, которую нужно проанализировать:

Настройка поля

Поле

A

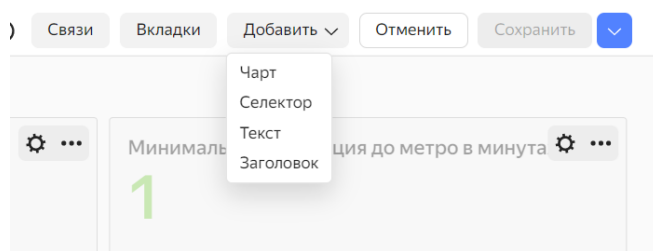
ROUND([price])

Fx

1 ROUND([Цена])

Таким образом я сделала несколько чартов, которые теперь можно перенести в дашборд.

В дашборд можно добавлять чарты, селекторы, текст:



Селекторы:

Способ добраться до квартиры

Район

Поиск

Выбрать все

Район	Цена
Северо-запад, около Таганско-Краснопресненской линии метро	14
Северо-запад, около Замоскворецкой линии метро	14
Восток, около Арбатско-Покровской линии метро	13
Север, около Калужско-Рижской линии метро	13
Юго-восток, около Люблинской линии метро	11

Текст:



Довольно благоприятным для покупки квартиры районом является Юго-Восток, около Люблинской линии метро, так как площади квартир больше, чем во многих других районах, а цены минимальные. Однако при рассмотрении квартир следует учитывать и другие факторы: местоположение, безопасность, инфраструктуру и т.д.

И чарты:

