

# Тема - Исследования рынка общепита в Москве для принятия решения об открытии нового заведения

## Содержание

- 1 Описание проекта
- 2 Шаг 1. Загрузим данные и изучим общую информацию
  - 2.1 Подключим необходимые библиотеки
  - 2.2 Загрузим наш датасет
  - 2.3 Изучим общую информацию о датасете
  - 2.4 Посмотри на уникальные значения в столбцах `category` и `district`
  - 2.5 Вывод по шагу №1
- 3 Шаг 2. Выполним предобработку данных
  - 3.1 Проверим датасет на дубликаты
  - 3.2 Проверим количество пропусков
  - 3.3 Добавим столбец `street` с названиями улиц из столбца с адресом
  - 3.4 Добавим столбец `is_24/7` с обозначением, что заведение работает ежедневно и круглосуточно (24/7):
  - 3.5 Вывод по шагу №2 - Предобработка данных
- 4 Шаг 3. Анализ данных
  - 4.1 Какие категории заведений представлены в данных?
  - 4.2 Исследуйте количество посадочных мест в местах по категориям: рестораны, кофейни, пиццерии, бары и так далее.
  - 4.3 Рассмотрите и изобразите соотношение сетевых и несетевых заведений в датасете. Каких заведений больше?
  - 4.4 Какие категории заведений чаще являются сетевыми?
  - 4.5 Сгруппируйте данные по названиям заведений и найдите топ-15 популярных сетей в Москве.
  - 4.6 Какие административные районы Москвы присутствуют в датасете?
  - 4.7 Визуализируйте распределение средних рейтингов по категориям заведений.
  - 4.8 Постройте фоновую картограмму (хороплет) со средним рейтингом заведений каждого района.
  - 4.9 Отобразите все заведения датасета на карте с помощью кластеров средствами библиотеки `folium`.
  - 4.10 Найдите топ-15 улиц по количеству заведений.
  - 4.11 Найдите улицы, на которых находится только один объект общепита.
  - 4.12 Значения средних чеков заведений хранятся в столбце `middle_avg_bill`. Эти числа показывают примерную стоимость заказа в рублях, которая чаще всего выражена диапазоном. Посчитайте медиану этого столбца для каждого района.
  - 4.13 Посмотрим как распределены круглосуточные заведения на карте Москвы
  - 4.14 Посмотрим к каким категориям относятся круглосуточные заведения в Москве
  - 4.15 Вывод по Шагу №3. Анализ данных
- 5 Шаг 4. Детализируем исследование: открытие кофейни
  - 5.1 Посчитаем количество кофеен в датасете

- 5.2 Посмотрим на топ-5 популярных сетевых кофеен
- 5.3 Узнаем, в каких районах больше всего кофеен
- 5.4 Узнаем долю круглосуточных кофеен
- 5.5 Какие у кофеен рейтинги? Как они распределяются по районам?
- 5.6 На какую стоимость чашки капучино стоит ориентироваться при открытии и почему?
- 5.7 Посчитаем медианное число посадочных мест для несетевых кофеен
- 5.8 Предложение инвесторам фонда по открытию кофейни (вывод по исследованию кофеен)
- 6 Вывод по исследованию - Шаг №5
- 7 Презентация

## Описание проекта

Инвесторы из фонда «Shut Up and Take My Money» решили попробовать себя в новой области и открыть заведение общественного питания в Москве. Заказчики ещё не знают, что это будет за место: кафе, ресторан, пиццерия, паб или бар, — и какими будут расположение, меню и цены.

Для начала они просят нас — как аналитика — подготовить исследование рынка Москвы, найти интересные особенности и презентовать полученные результаты, которые в будущем помогут в выборе подходящего инвесторам места.

Нам доступен датасет с заведениями общественного питания Москвы, составленный на основе данных сервисов Яндекс Карты и Яндекс Бизнес на лето 2022 года. Информация, размещённая в сервисе Яндекс Бизнес, могла быть добавлена пользователями или найдена в общедоступных источниках.

## Шаг 1. Загрузим данные и изучим общую информацию

### Подключим необходимые библиотеки

In [1]:

```
# Запустить если не установлена библиотека folium
!pip install folium
```

```
Requirement already satisfied: folium in c:\users\rulkevichvv\anaconda3\lib\site-packages
(0.14.0)
Requirement already satisfied: requests in c:\users\rulkevichvv\anaconda3\lib\site-packages (from folium) (2.26.0)
Requirement already satisfied: jinja2>=2.9 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from folium) (2.11.3)
Requirement already satisfied: branca>=0.6.0 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from folium) (0.6.0)
Requirement already satisfied: numpy in c:\users\rulkevichvv\anaconda3\lib\site-packages (from folium) (1.20.3)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (1.1.1)
Requirement already satisfied: idna<4,>=2.5 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from requests->folium) (3.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from requests->folium) (2021.10.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from requests->folium) (1.26.7)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\rulkevichvv\anaconda3\lib\site-packages (from requests->folium) (2.0.4)
```

In [2]:

```
import pandas as pd
import seaborn as sns
from datetime import datetime as dt
```

```

import matplotlib.pyplot as plt
import numpy as np
import plotly.graph_objects as go
import plotly.express as px
import warnings

# импортируем карту, маркер и хороплет
from folium import Map, Choropleth, Marker
# импортируем кластер
from folium.plugins import MarkerCluster
# подключаем модуль для работы с JSON-форматом
import json

```

## Загрузим наш датасет

In [3]:

```

try:
    df = pd.read_csv('/datasets/moscow_places.csv', sep=',')
except:
    df = pd.read_csv('moscow_places.csv', sep=',')

```

In [4]:

```
df.head()
```

Out[4]:

	<b>name</b>	<b>category</b>	<b>address</b>	<b>district</b>	<b>hours</b>	<b>lat</b>	<b>lng</b>	<b>rating</b>	<b>price</b>
0	WoWфли	кафе	Москва, улица Дыбенко, 7/1	Северный административный округ	ежедневно, 10:00–22:00	55.878494	37.478860	5.0	NaN
1	Четыре комнаты	ресторан	Москва, улица Дыбенко, 36, корп. 1	Северный административный округ	ежедневно, 10:00–22:00	55.875801	37.484479	4.5	выше среднего
2	Хазри	кафе	Москва, Клязьминская улица, 15	Северный административный округ	ПН-ЧТ 11:00–02:00; ПТ,СБ 11:00–05:00; ВС 11:00...	55.889146	37.525901	4.6	средние
3	Dormouse Coffee Shop	кофейня	Москва, улица Маршала Федоренко, 12	Северный административный округ	ежедневно, 09:00–22:00	55.881608	37.488860	5.0	NaN
4	Илья Марко	пиццерия	Москва, Правобережная улица, 1Б	Северный административный округ	ежедневно, 10:00–22:00	55.881166	37.449357	5.0	средние

## Изучим общую информацию о датасете

In [5]:

```

df.info()
df.shape

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8406 entries, 0 to 8405
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   name            8406 non-null   object  
 1   category        8406 non-null   object  
 2   address         8406 non-null   object  
 3   district        8406 non-null   object  
 4   hours           8406 non-null   object  
 5   lat             8406 non-null   float64
 6   lng             8406 non-null   float64
 7   rating          8406 non-null   float64
 8   price           8406 non-null   float64
 9   geom_type       8406 non-null   object  
 10  geometry        8406 non-null   geometry
 11  geometry_type  8406 non-null   object  
 12  place_id        8406 non-null   int64  
 13  place_name      8406 non-null   object  

```

```
4    hours           7870 non-null   object
5    lat              8406 non-null   float64
6    lng              8406 non-null   float64
7    rating           8406 non-null   float64
8    price             3315 non-null   object
9    avg_bill          3816 non-null   object
10   middle_avg_bill   3149 non-null   float64
11   middle_coffee_cup  535 non-null   float64
12   chain             8406 non-null   int64
13   seats              4795 non-null   float64
dtypes: float64(6), int64(1), object(7)
memory usage: 919.5+ KB
(8406, 14)
```

Out[5]:

Как мы видим, в нашем датасете - информация о **8 406 заведениях города Москвы.**

В нашей таблице - 14 столбцов с данными.

В следующих полях есть пропуски -

- `hours` - часы работы
- `price` - Категория цен в заведении (средние, ниже среднего, выше среднего и тд)
- `avg_bill` - Средняя стоимость заказа в виде диапазона
- `middle_avg_bill` - число с оценкой среднего чека, которое указано только для значений из столбца `avg_bill`, начинающихся с подстроки «Средний счёт»
- `middle_coffee_cup` - число с оценкой одной чашки капучино, которое указано только для значений из столбца `avg_bill`, начинающихся с подстроки «Цена одной чашки капучино»
- `seats` - количество посадочных мест

Для столбца `chain` (является ли заведение сетевым) - указан некорректный тип данных. Необходимо перевести `int64` --> `boolean`

```
In [6]: df['chain'] = df['chain'].astype('bool')
```

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8406 entries, 0 to 8405
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   name              8406 non-null   object 
 1   category          8406 non-null   object 
 2   address            8406 non-null   object 
 3   district           8406 non-null   object 
 4   hours              7870 non-null   object 
 5   lat                 8406 non-null   float64
 6   lng                 8406 non-null   float64
 7   rating              8406 non-null   float64
 8   price                3315 non-null   object 
 9   avg_bill             3816 non-null   object 
 10  middle_avg_bill     3149 non-null   float64
 11  middle_coffee_cup    535 non-null   float64
 12  chain                  8406 non-null   bool    
 13  seats                  4795 non-null   float64
dtypes: bool(1), float64(6), object(7)
memory usage: 862.1+ KB
```

# Посмотри на уникальные значения в столбцах category и district

In [8]:

```
print(df['category'].unique())
print()
print(df['district'].unique())

['кафе' 'ресторан' 'кофейня' 'пиццерия' 'бар, паб' 'быстрое питание'
 'булочная' 'столовая']

['Северный административный округ'
 'Северо-Восточный административный округ'
 'Северо-Западный административный округ'
 'Западный административный округ' 'Центральный административный округ'
 'Восточный административный округ' 'Юго-Восточный административный округ'
 'Южный административный округ' 'Юго-Западный административный округ']
```

В нашем датасете присутствует информация о **8 типах заведениях Общепита** и о **9 административных округах Москвы**

## Вывод по шагу №1

Мы имеем датафрейм с информацией о **8 406 заведениях Общепита** города Москвы

В нашей таблице - 14 столбцов с данными

- Название заведения
- Категория заведения
- Адрес
- Район Москвы
- Часы работы
- Долгота и широта (координаты)
- Рейтинг заведения
- Средний чек
- Средняя цена кружки кофе
- Является ли заведение сетевым
- Количество посадочных мест

В нашем датасете присутствует информация о **8 типах заведениях Общепита** и о **9 административных округах Москвы**

В следующих полях есть пропуски -

- hours - часы работы
- price - Категория цен в заведении (средние, ниже среднего, выше среднего и тд)
- avg\_bill - Средняя стоимость заказа в виде диапазона
- middle\_avg\_bill - число с оценкой среднего чека, которое указано только для значений из столбца avg\_bill, начинающихся с подстроки «Средний счёт»
- middle\_coffee\_cup - число с оценкой одной чашки капучино, которое указано только для значений из столбца avg\_bill, начинающихся с подстроки «Цена одной чашки капучино»
- seats - количество посадочных мест

## Шаг 2. Выполним предобработку данных

Проверим датасет на дубликаты

In [9]:

```
print(f"Количество дубликатов в датафрейме - {df.duplicated().sum()}"")
```

Количество дубликатов в датафрейме - 0

Дубликатов в нашем датасете - нет.

Проведем дополнительную проверку на неявные дубликаты по столбцам, где не было пропусков -

- Имя
- Категория
- Адрес
- Район

In [10]:

```
duplicates = df.duplicated(subset=['name', 'category', 'address', 'district']).sum()
duplicates
```

Out[10]:

0

Мы еще раз убедились, что дубликатов в нашем датасете - нет.

In [11]:

```
# код ревьюера
df['name'] = df['name'].str.lower()
df['address'] = df['address'].str.lower()

df[df.duplicated(subset=['name', 'address'])]
```

Out[11]:

		name	category	address	district	hours	lat	lng	rating	price	avg_bill
215		кафе	кафе	москва, парк аангарские пруды	Северный административный округ	ежедневно, 10:00–22:00	55.881438	37.531848	3.2	NaN	
1511	more poke	ресторан		москва, воловоламское шоссе, 11, стр. 2	Северный административный округ	пн-чт 09:00– 18:00; пт,сб 09:00– 21:00; вс 09:00...	55.806307	37.497566	4.2	NaN	
2420	раковарня клешни и хвосты	бар,паб		москва, проспект мира, 118	Северо-Восточный административный округ	пн-чт 12:00– 00:00; пт,сб 12:00– 01:00; вс 12:00...	55.810677	37.638379	4.4	NaN	
3109	хлеб да выпечка		кафе	москва, ярцевская улица, 19	Западный административный округ	NaN	55.738449	37.410937	4.1	NaN	

## Проверим количество пропусков

In [12]:

```
pd.DataFrame(round(df.isna().mean()*100, 1)).sort_values(0, ascending=False).style.background
```

Out[12]:

0

middle_coffee_cup	93.600000
-------------------	-----------

middle_avg_bill	62.500000
-----------------	-----------

	0
price	60.600000
avg_bill	54.600000
seats	43.000000
hours	6.400000
name	0.000000
category	0.000000
address	0.000000
district	0.000000
lat	0.000000
lng	0.000000
rating	0.000000
chain	0.000000

- Как мы видим, самое большое количество пропусков 93% в столбце `middle_coffee_cup` - Средняя цена чашки кофе.
- Так же большое количество пропусков 62% - в поле `midle_avg_bill` - число с оценкой среднего чека, которое указано только для значений из столбца `avg_bill`
- Как мы видим, в поле `avg_bill` также есть пропуски, которые мы ничем не можем заменить, т.к. их довольно много.
- Еще есть пропуски в столбце `seats` - количество посадочных мест -43%
- часы работы заведения `hours` - 6%

Чтобы не искажить данные - не будем вносить изменения в пропуски на данном этапе

## Добавим столбец `street` с названиями улиц из столбца с адресом

Посмотрим на уникальные значения в поле `address`

```
In [13]: df['address'].unique()
```

```
Out[13]: array(['москва, улица дыбенко, 7/1', 'москва, улица дыбенко, 36, корп. 1',
   'москва, клязьминская улица, 15', ...,
   'москва, улица лобачевского, 52, корп. 1',
   'москва, болотниковская улица, 52, корп. 2',
   'москва, чонгарский бульвар, 26а, корп. 1'], dtype=object)
```

Как мы видим, улица идет сразу после названия города и разделена запятыми -

Вычленим название улицы -

```
In [14]: df['street'] = df['address'].str.split(pat=', ', expand=True)[1]
```

```
In [15]: df.head()
```

	name	category	address	district	hours	lat	lng	rating	price
--	------	----------	---------	----------	-------	-----	-----	--------	-------

	<b>name</b>	<b>category</b>	<b>address</b>	<b>district</b>	<b>hours</b>	<b>lat</b>	<b>lng</b>	<b>rating</b>	<b>price</b>
0	wowфли	кафе	москва, улица дыбенко, 7/1	Северный административный округ	ежедневно, 10:00–22:00	55.878494	37.478860	5.0	NaN
1	четыре комнаты	ресторан	москва, улица дыбенко, 36, корп. 1	Северный административный округ	ежедневно, 10:00–22:00	55.875801	37.484479	4.5	выше среднего
2	хазри	кафе	москва, клязьминская улица, 15	Северный административный округ	пн-чт 11:00–02:00; пт,сб 11:00–05:00; вс 11:00...	55.889146	37.525901	4.6	средние
3	dormouse coffee shop	кофейня	москва, улица маршала федоренко, 12	Северный административный округ	ежедневно, 09:00–22:00	55.881608	37.488860	5.0	NaN ка
4	иль марко	пиццерия	москва, правобережная улица, 16	Северный административный округ	ежедневно, 10:00–22:00	55.881166	37.449357	5.0	средние сч

Добавим столбец `is_24/7` с обозначением, что заведение работает ежедневно и круглосуточно (24/7):

Посмотрим на уникальные значения столбца - `hours`

In [16]:

```
print(df['hours'].nunique())
```

1307

Мы имеем - 1307 уникальных значений в данном поле

In [17]:

```
df['hours'].str.split(pat= ',', ',', expand=True)
```

Out[17]:

	0	1	2
0	ежедневно	10:00–22:00	None
1	ежедневно	10:00–22:00	None
2	пн-чт 11:00–02:00; пт,сб 11:00–05:00; вс 11:00...	None	None
3	ежедневно	09:00–22:00	None
4	ежедневно	10:00–22:00	None
...	...	...	...
8401	ежедневно	09:00–02:00	None
8402	ежедневно	08:00–22:00	None
8403	ежедневно	круглосуточно	None
8404	ежедневно	круглосуточно	None
8405	ежедневно	круглосуточно	None

8406 rows × 3 columns

Создадим столбец согласно условию, если заведение работает ежедневно и круглосуточно, то в поле `is_24/7` будет `True`

In [18]:

```
df['is_24/7']=np.where((df['hours'].str.split(pat= ',', expand=True)[0] == 'ежедневно') &
                      (df['hours'].str.split(pat= ',', expand=True)[1].str.contains('круглосуточно')) , True, False)
```

In [19]:

```
# код ревьюера
df['is_24/7'] = df['hours'].str.contains('ежедневно, круглосуточно')
df['is_24/7'].value_counts()
```

Out[19]:

```
False    7140
True     730
Name: is_24/7, dtype: int64
```

In [20]:

```
df.tail()
```

Out[20]:

		name	category	address	district	hours	lat	lng	rating	price
8401	суши мания	кафе		москва, профсоюзная улица, 56	Юго-Западный административный округ	ежедневно, 09:00–02:00	55.670021	37.552480	4.4	NaN
8402	миславнес	кафе		москва, пролетарский проспект, 19, корп. 1	Южный административный округ	ежедневно, 08:00–22:00	55.640875	37.656553	4.8	NaN
8403	самовар	кафе		москва, люблинская улица, 112а, стр. 1	Юго-Восточный административный округ	ежедневно, круглосуточно	55.648859	37.743219	3.9	NaN
8404	чайхана sabr	кафе		москва, люблинская улица, 112а, стр. 1	Юго-Восточный административный округ	ежедневно, круглосуточно	55.648849	37.743222	4.2	NaN
8405	kebab time	кафе		москва, россошанский проезд, 6	Южный административный округ	ежедневно, круглосуточно	55.598229	37.604702	3.9	NaN

In [21]:

```
df['is_24/7'].unique()
```

Out[21]:

```
array([False, True, nan], dtype=object)
```

## Выход по шагу №2 - Предобработка данных

1. Явных дубликатов в нашем датасете - не выявлено

2. В 6 столбцах с данными есть пропуски - было решено оставить их, чтобы не искажить данные

middle_coffee_cup	93.600000
middle_avg_bill	62.500000
price	60.600000
avg_bill	54.600000
seats	43.000000
hours	6.400000

исследования

3. Самое большое количество пропусков в поле `middle_coffee_cup` - Средняя цена чашки кофе - 93% пропусков
4. Добавили столбец `street` - для группировки данных по улице - понадобится в дальнейшем анализе
5. Добавили столбец `is_24/7` с указанием - работает ли заведение ежедневно и круглосуточно

## Шаг 3. Анализ данных

### Какие категории заведений представлены в данных?

Исследуйте количество объектов общественного питания по категориям: рестораны, кофейни, пиццерии, бары и так далее.

Постройте визуализации.

In [22]:

```
category = df.groupby('category', as_index=False) ['name'] \
    .count() \
    .sort_values(by='name', ascending=False) \
    .reset_index(drop=True)
category.columns = ('category', 'count')
category
```

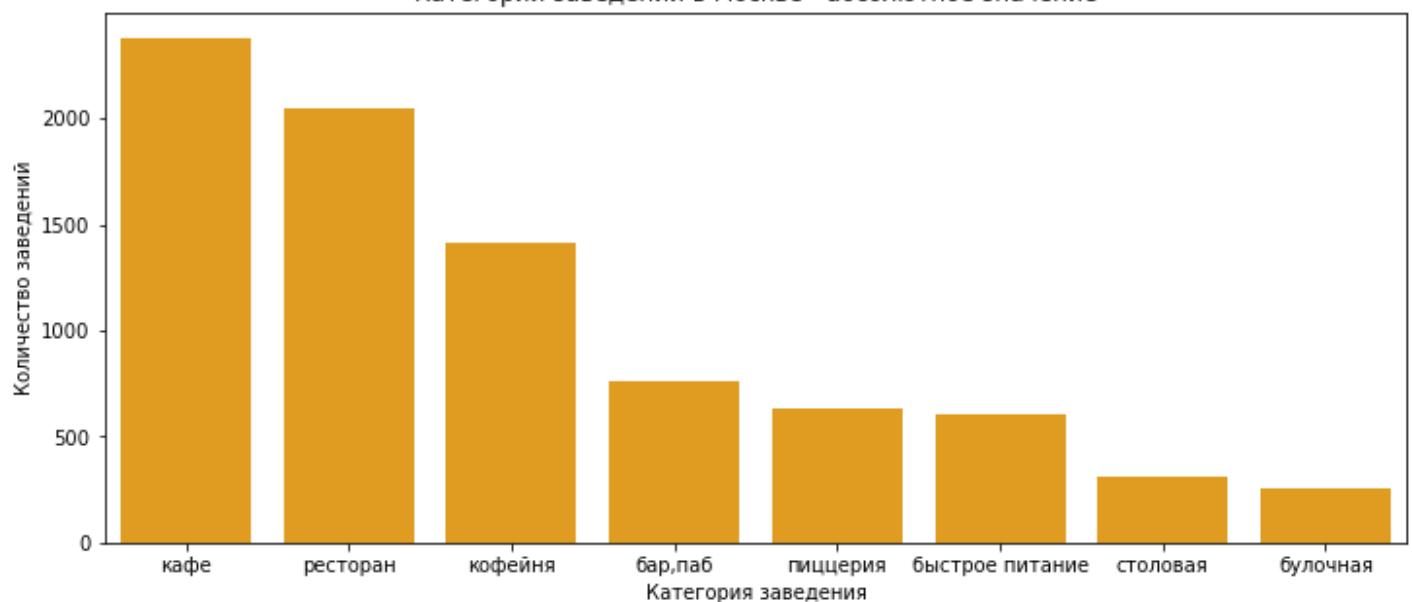
Out[22]:

	category	count
0	кафе	2378
1	ресторан	2043
2	кофейня	1413
3	бар,паб	765
4	пиццерия	633
5	быстрое питание	603
6	столовая	315
7	булочная	256

In [23]:

```
plt.figure(figsize=(12, 5))
plt.title('Категории заведений в Москве - абсолютное значение')
ax = sns.barplot(x='category', y='count', data=category, color='orange')
ax.set(xlabel='Категория заведения', ylabel='Количество заведений')
plt.show()
```

## Категории заведений в Москве - абсолютное значение



Как мы видим, больше всего в Москве - кафе и ресторанов (2378 и 2043 соответственно)

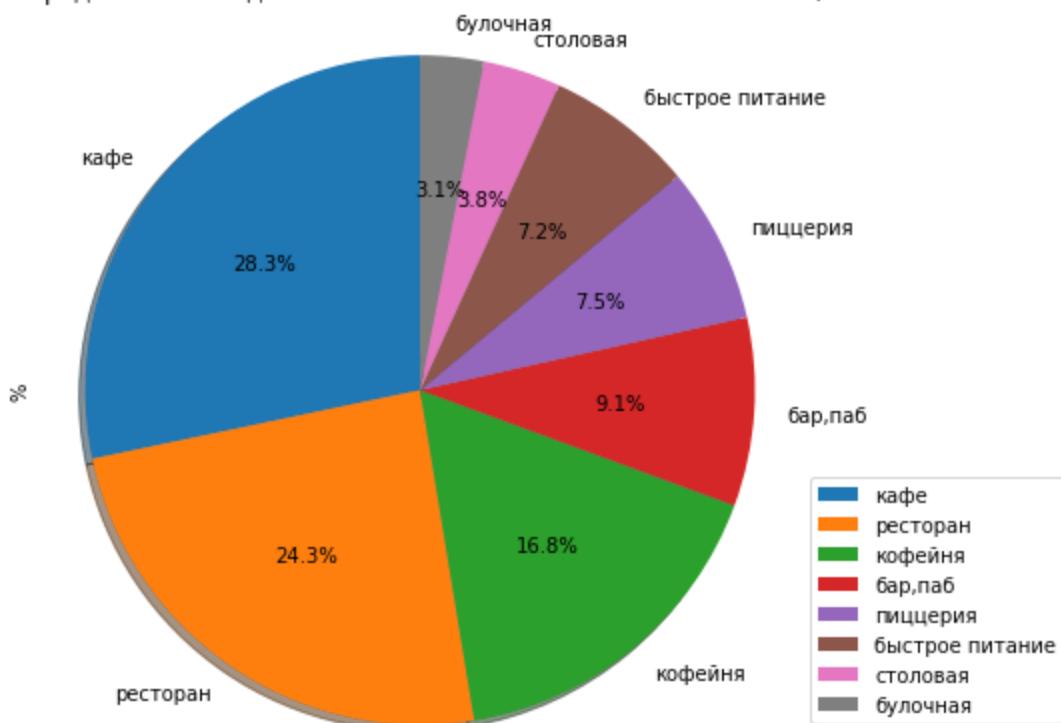
Посмотрим на относительное количество -

```
In [24]: category['%'] = round(category['count']*100/category['count'].sum(),2)
category
category.set_index('category', inplace=True)
```

```
In [25]: ax = category.plot(
    kind='pie',
    y='%',
    shadow = True,
    startangle=90,
    figsize = (12,7),
    autopct='%1.1f%%',
    radius = 1.1,
    title='Распределение заведений в Москве - относительное количество, %'

)
ax.legend(bbox_to_anchor=(1, 0.4), loc='upper left')
plt.show()
```

## Распределение заведений в Москве - относительное количество, %



In [26]:

```
# код ревьюера
category = category.reset_index()
category['%'] = category['%'].astype(str) + '%'
category
```

Out[26]:

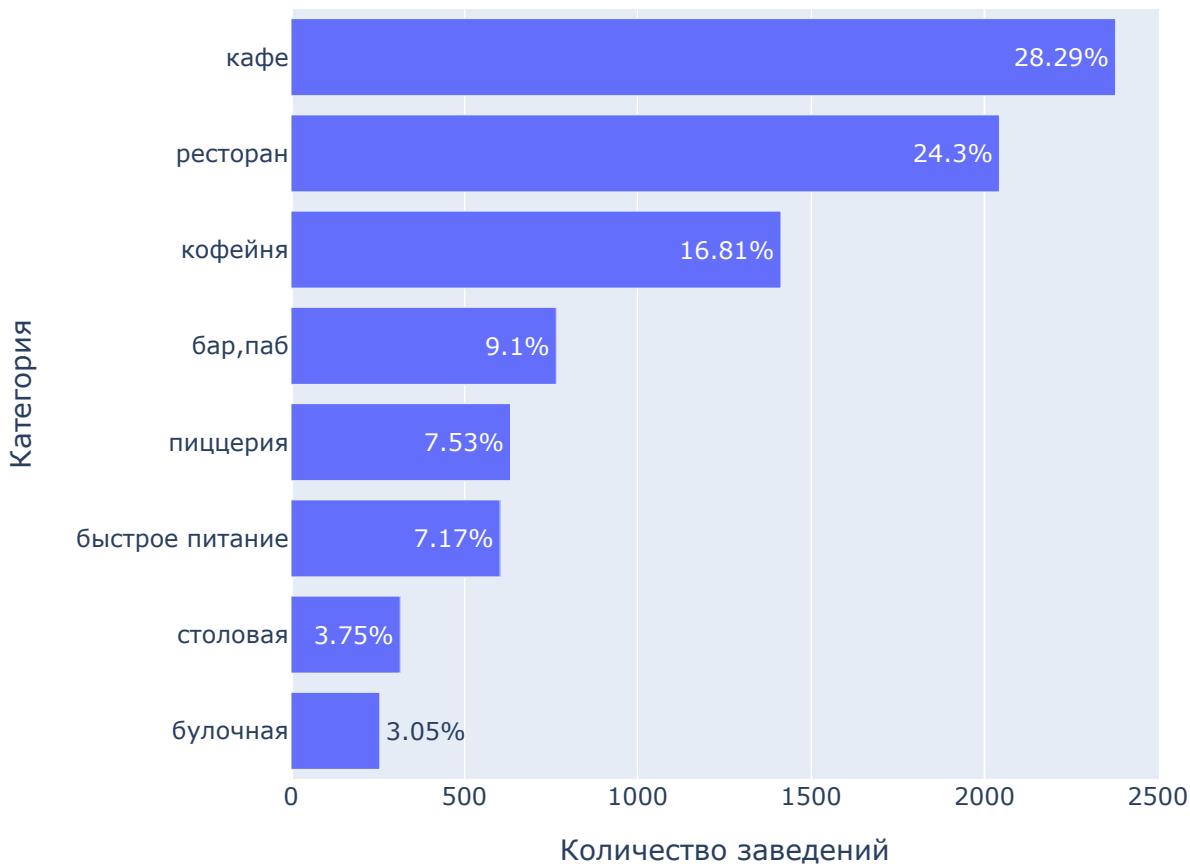
	category	count	%
0	кафе	2378	28.29%
1	ресторан	2043	24.3%
2	кофейня	1413	16.81%
3	бар,паб	765	9.1%
4	пиццерия	633	7.53%
5	быстрое питание	603	7.17%
6	столовая	315	3.75%
7	булочная	256	3.05%

In [27]:

```
fig = px.bar(category,
              y='category',
              x='count',
              text='%'
            )

fig.update_layout(title='Распределение заведений в Москве',
                  xaxis_title='Количество заведений',
                  yaxis_title='Категория',
                  yaxis={'categoryorder':'total ascending'}
                )
fig.show()
```

## Распределение заведений в Москве



**Исследуйте количество посадочных мест в местах по категориям: рестораны, кофейни, пиццерии, бары и так далее.**

Постройте визуализации. Проанализируйте результаты и сделайте выводы.

Для визуализации возьмем медианное число посадочных мест в каждой из категорий заведения (это будет более корректно, чем брать среднее).

In [28]:

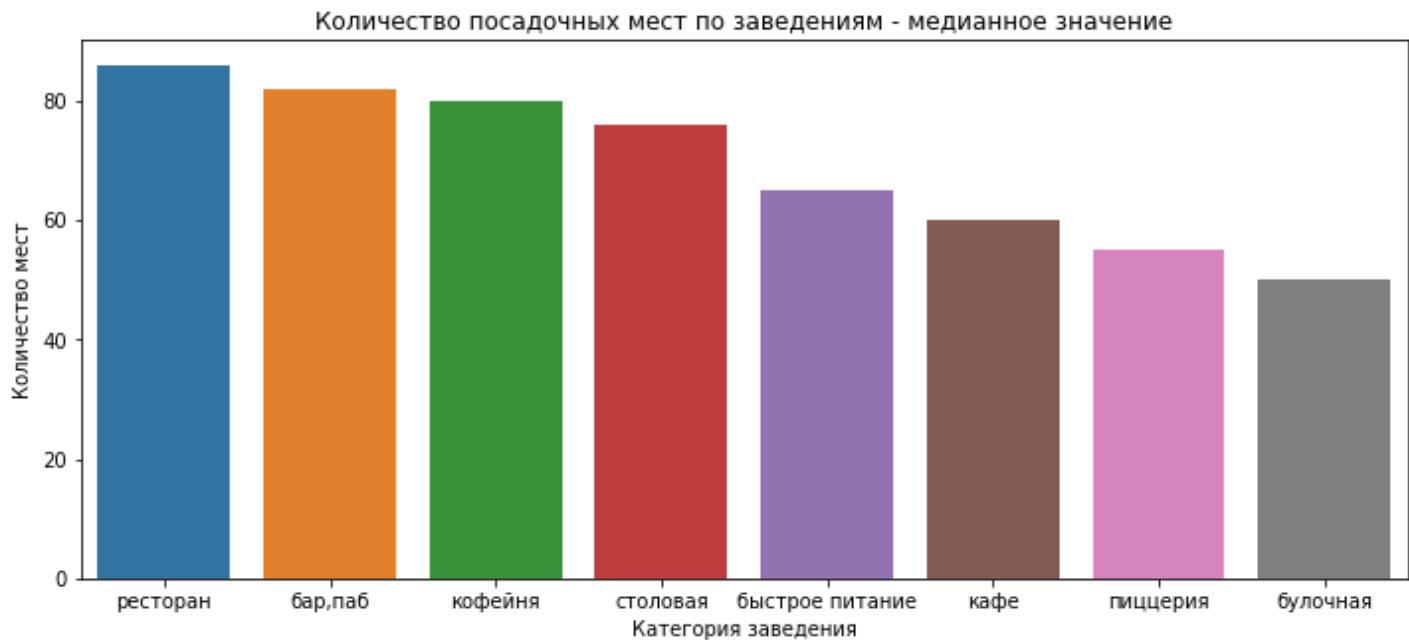
```
seats_category = pd.pivot_table(df, index='category', values='seats', aggfunc='median') \
    .sort_values(by='seats', ascending=False) \
    .reset_index()
seats_category['seats'] = round(seats_category['seats'])
seats_category
```

Out[28]:

	category	seats
0	ресторан	86.0
1	бар, паб	82.0
2	кофейня	80.0
3	столовая	76.0
4	быстрое питание	65.0
5	кафе	60.0
6	пиццерия	55.0
7	булочная	50.0

In [29]:

```
plt.figure(figsize=(12, 5))
plt.title('Количество посадочных мест по заведениям - медианное значение')
ax = sns.barplot(x='category',y='seats',data=seats_category)
ax.set(xlabel='Категория заведения', ylabel='Количество мест')
plt.show()
```



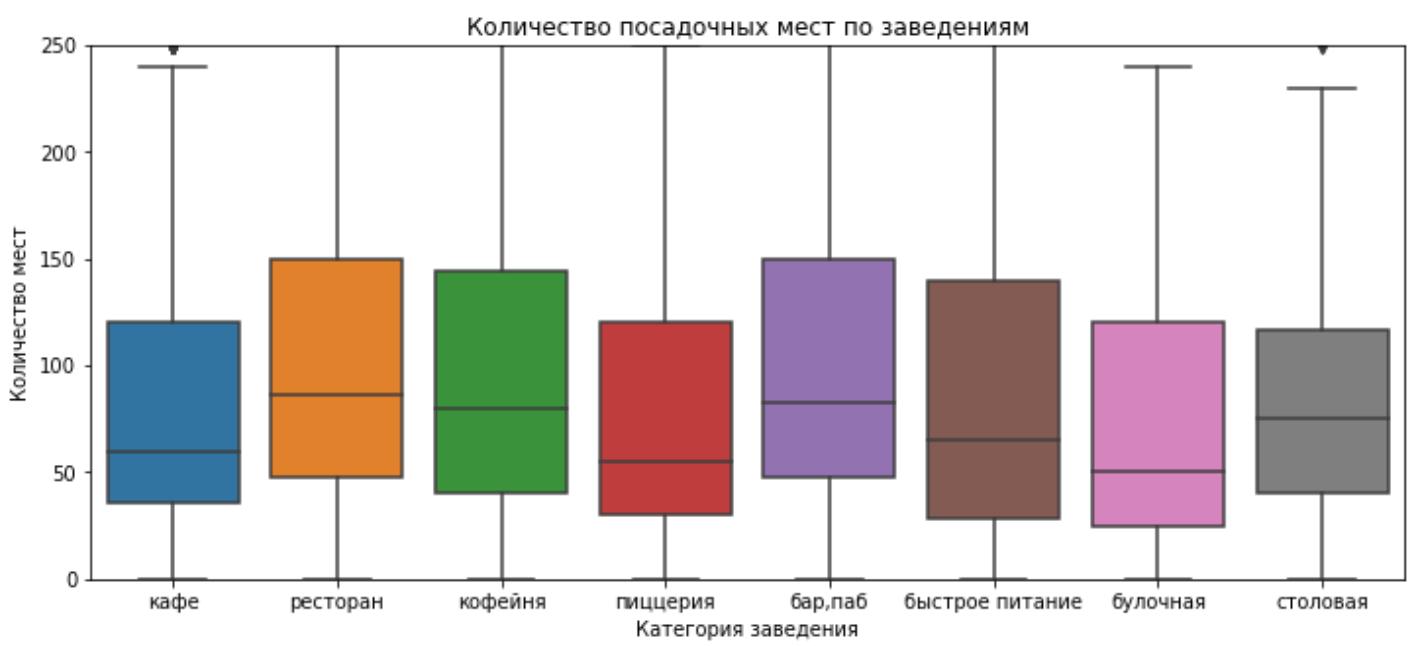
Как мы видим, **медианное значение посадочных мест в больших заведениях (ресторан, бар/паб, кофейня, столовая)** - составляет от 76 до 86 посадочных мест

**В маленьких заведениях (Булочная, пицерия, кафе, быстрое питание)** - количество мест составляет - 55-65

Для наглядности - нарисуем график - **boxplot** - взяв за верхнее значение количество посадочных мест менее 200

In [30]:

```
plt.figure(figsize=(12, 5))
plt.title('Количество посадочных мест по заведениям')
plt.ylim([0,250])
ax = sns.boxplot(data=df,x='category',y='seats')
ax.set(xlabel='Категория заведения', ylabel='Количество мест')
plt.show()
```



**Рассмотрите и изобразите соотношение сетевых и несетевых заведений в датасете. Каких заведений больше?**

In [31]:

```
chain = df.groupby('chain', as_index=False) ['name'].count()
chain.columns = ([ 'сетевое заведение', 'количество'])
chain['%'] = round(chain['количество']*100/ chain['количество'].sum(),2)
chain.loc[chain['сетевое заведение']==True, 'сетевое заведение'] = 'Сетевое заведение'
chain.loc[chain['сетевое заведение']==False, 'сетевое заведение'] = 'Несетевое заведение'
chain
```

Out[31]:

	сетевое заведение	количество	%
0	Несетевое заведение	5201	61.87
1	Сетевое заведение	3205	38.13

Как мы видим, в нашем датасете около 62% (~ 5 200 шт) несетевых заведений.

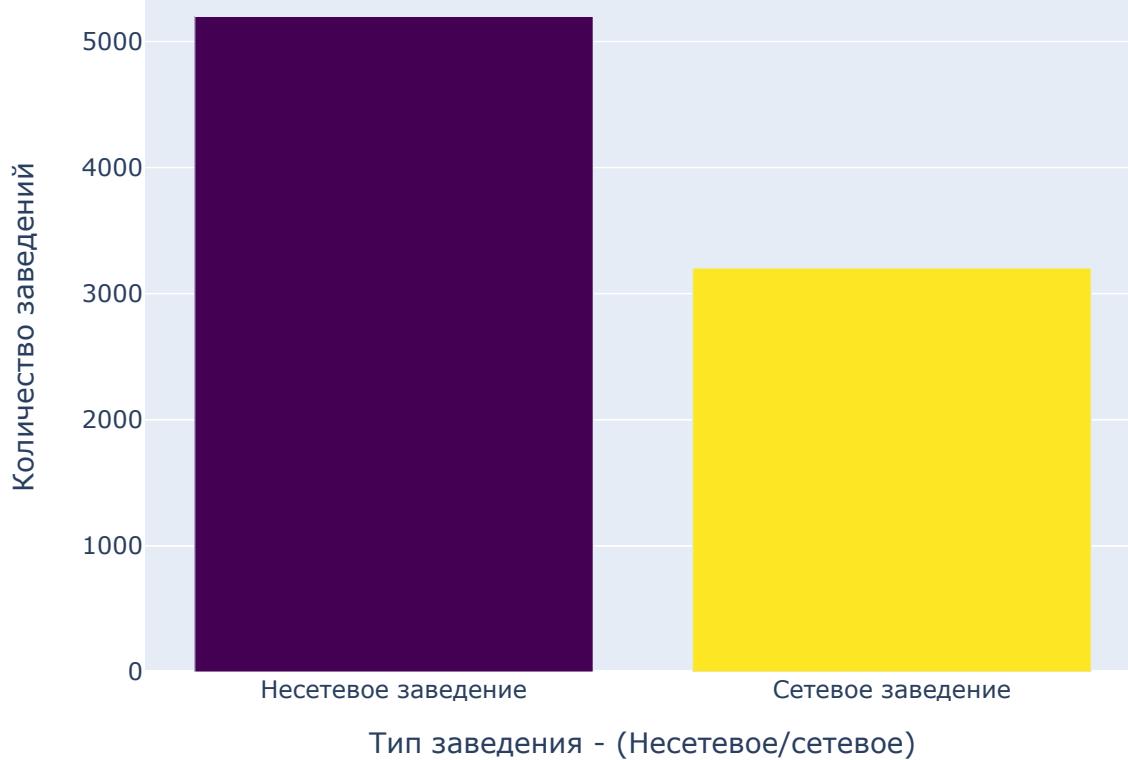
Остальную часть занимают сетевые заведения - 38% (~ 3200 шт)

Визуализируем эти данные

In [32]:

```
fig = go.Figure(go.Bar(
    x=chain['сетевое заведение'],
    y=chain['количество'],
    orientation='v',
    marker=dict(color = [1,2],
                colorscale='viridis')))\n.update_layout(\n    xaxis_title="Тип заведения - (Несетевое/сетевое)", yaxis_title="Количество заведений",
    title="Соотношение несетевых и сетевых заведений (абсолютное)", title_x=0.5\n)
fig.show()
```

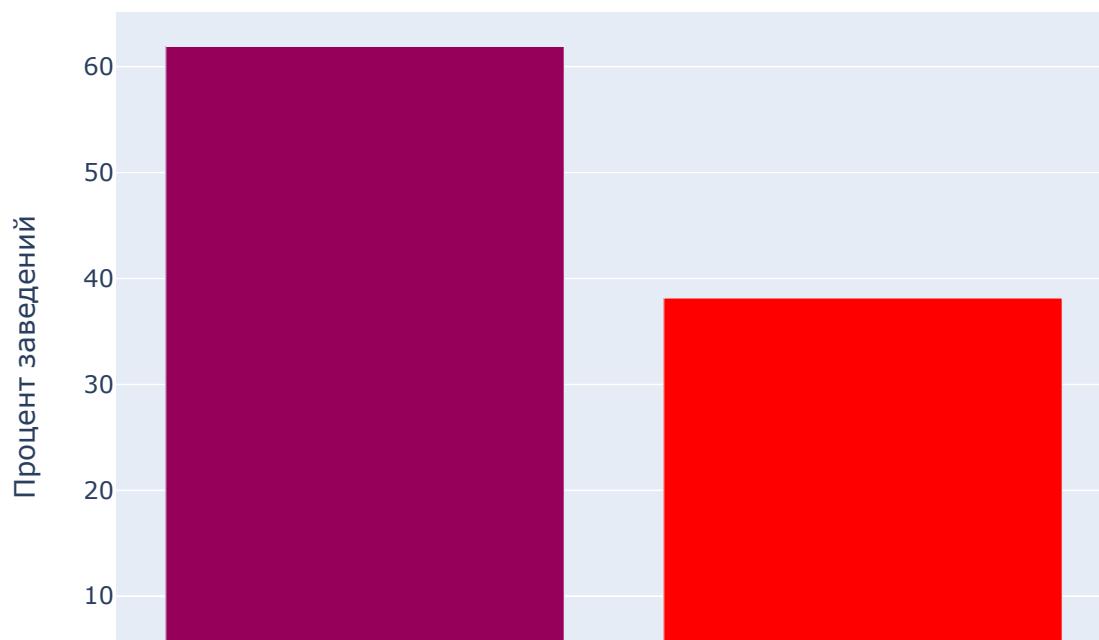
Соотношение несетевых и сетевых заведений (абсолютное)



In [33]:

```
fig = go.Figure(go.Bar(
    x=chain['сетевое заведение'],
    y=chain['%'],
    orientation='v',
    marker=dict(color = [1,1.2],
                colorscale='rainbow')))\n.update_layout(\n    xaxis_title="Тип заведения - (Сетевое/несетевое)", yaxis_title="Процент заведений",
    title="Соотношение сетевых и несетевых заведений (относительное, %)",title_x=0.5\n)
fig.show()
```

Соотношение сетевых и несетевых заведений (относительное, %)



## Тип заведения - (Сетевое/несетевое)

**Какие категории заведений чаще являются сетевыми?**

Исследуйте данные и ответьте на вопрос графиком

Посмотрим на абсолютное количество сетевых заведений -

In [34]:

```
chain_category = df[df['chain']==True].groupby('category', as_index=False) ['name'] \
    .count() \
    .sort_values('name', ascending=False) \
    .reset_index(drop=True)
chain_category
```

Out[34]:

	category	name
0	кафе	779
1	ресторан	730
2	кофейня	720
3	пиццерия	330
4	быстрое питание	232
5	бар,паб	169
6	булочная	157
7	столовая	88

Посчитаем какой процент от общего количества заведений в каждой категории занимают сети -

In [35]:

```
# Посчитаем общее количество
all_category = df.groupby('category', as_index=False) ['name'] \
    .count() \
    .sort_values('name', ascending=False) \
    .reset_index(drop=True)
all_category.columns =(['category', 'total'])

#Склеим таблицы по полю категория
chain_category = chain_category.merge(all_category, on='category', how='left')

chain_category
```

Out[35]:

	category	name	total
0	кафе	779	2378
1	ресторан	730	2043
2	кофейня	720	1413
3	пиццерия	330	633
4	быстрое питание	232	603

	category	name	total
--	----------	------	-------

5	бар,паб	169	765
6	булочная	157	256
7	столовая	88	315

In [36]:

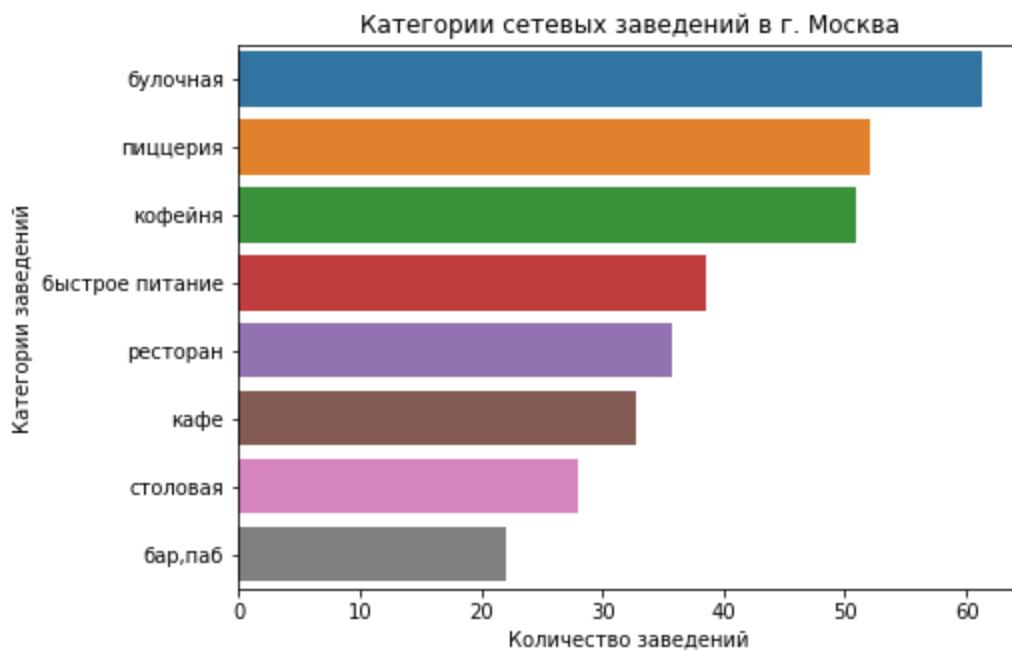
```
#Посчитаем процент
chain_category['%']=round(chain_category['name']*100/chain_category['total'],2)
chain_category = chain_category.sort_values('%',ascending=False).reset_index(drop=True)
chain_category
```

Out[36]:

	category	name	total	%
0	булочная	157	256	61.33
1	пиццерия	330	633	52.13
2	кофейня	720	1413	50.96
3	быстрое питание	232	603	38.47
4	ресторан	730	2043	35.73
5	кафе	779	2378	32.76
6	столовая	88	315	27.94
7	бар,паб	169	765	22.09

In [37]:

```
plt.figure(figsize=(7, 5))
plt.title('Категории сетевых заведений в г. Москва')
ax = sns.barplot(x='%', y='category', data=chain_category)
ax.set(xlabel='Количество заведений', ylabel='Категории заведений')
plt.show()
```



Чаще всего сетевыми заведениями являются -

- Булочные
- Пиццерии

- Кофейни

## Сгруппируйте данные по названиям заведений и найдите топ-15 популярных сетей в Москве.

Под популярностью понимается количество заведений этой сети в регионе.

Постройте подходящую для такой информации визуализацию. Есть ли какой-то признак, который объединяет эти сети? К какой категории заведений они относятся?

In [38]:

```
top15_chain = df[df['chain']==True].groupby(['name', 'category'], as_index=False) ['address'
    .count()).sort_values('address', ascending=False).head(15).reset_index(drop=True)

top15_chain = top15_chain.rename(columns={'address': 'count'})
top15_chain
```

Out[38]:

		name	category	count
0		шоколадница	кофейня	119
1		домино'с пицца	пиццерия	76
2		додо пицца	пиццерия	74
3		one price coffee	кофейня	71
4		яндекс лавка	ресторан	69
5		cofix	кофейня	65
6		prime	ресторан	49
7		кофепорт	кофейня	42
8	кулинарная лавка братьев караваевых		кафе	39
9		теремок	ресторан	36
10		cofefest	кофейня	31
11		чайхана	кафе	26
12		буханка	булочная	25
13		drive café	кафе	24
14		кофемания	кофейня	22

In [39]:

```
fig = go.Figure(data=[go.Pie(labels=top15_chain['name'], values=top15_chain['count'])])
fig.update_layout(
    title="Топ 15 сетевых заведений в Москве", title_x=0.4)
fig.show()
```

Топ 15 сетевых заведений в Москве





Для наглядности добавим столбчатую диаграмму -

```
In [40]: fig = px.bar(top15_chain,
                  y='name',
                  x='count',
                  text = "count",
                  barmode='stack'
                 )

fig.update_layout(title='Топ 15 сетевых заведений в Москве',
                  xaxis_title='Количество заведений',
                  yaxis_title='Название заведения',
                  title_x=0.4,
                  yaxis={'categoryorder':'total ascending'}
                 )
fig.show()
```

Топ 15 сетевых заведений в Москве



Самые популярные сети относятся к категориям -

- кофейня
- пицерия

В списке есть также категория "рестораны" (Яндекс.Лавка, Теремок, Prime) - но это скорее кафе со средним уровнем цен.

**Общее у всех заведений - ценовая политика - сегмент - средний и ниже среднего.**

## Какие административные районы Москвы присутствуют в датасете?

Отобразите общее количество заведений и количество заведений каждой категории по районам.

In [41]:

```
district_total = df.groupby('district', as_index=False) ['name'].count().sort_values('name', ascending=False)
```

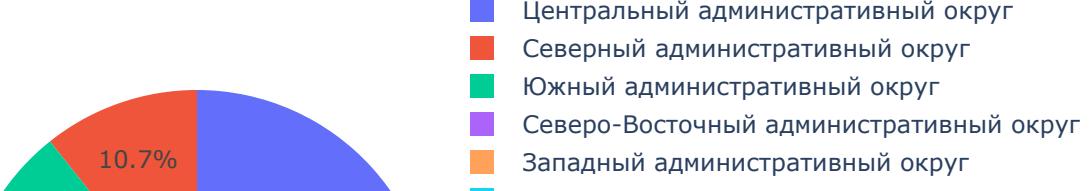
Out[41]:

	district	name
5	Центральный административный округ	2242
2	Северный административный округ	900
8	Южный административный округ	892
3	Северо-Восточный административный округ	891
1	Западный административный округ	851
0	Восточный административный округ	798
6	Юго-Восточный административный округ	714
7	Юго-Западный административный округ	709
4	Северо-Западный административный округ	409

In [42]:

```
fig = go.Figure(data=[go.Pie(labels=district_total['district'], values=district_total['name'],
fig.update_layout(
    title="Разделение заведений по районам Москвы", title_x=0.4)
fig.show()
```

## Разделение заведений по районам Москвы





**Большая часть заведений находится в центральном административном округе (далее - ЦАО)** - оно и понятно, т.к. в ЦАО - основное скопление туристов и платежеспособных людей.

In [43]:

```
district_type = pd.pivot_table(df, index=['district', 'category'], values='name', aggfunc='count').unstack().sort_values('district')

#district_type.columns = ('бар,паб', 'булочная', 'быстрое питание', 'кафе', 'кофейня', 'пиццерия', 'ресторан', 'столовая')
district_type
```

Out[43]:

	category	бар,паб	булочная	быстрое питание	кафе	кофейня	пиццерия	ресторан	столовая	name
district										
<b>Восточный административный округ</b>		53	25	71	272	105	72	160	40	
<b>Западный административный округ</b>		50	37	62	239	150	71	218	24	
<b>Северный административный округ</b>		68	39	58	235	193	77	189	41	
<b>Северо-Восточный административный округ</b>		63	28	82	269	159	68	182	40	
<b>Северо-Западный административный округ</b>		23	12	30	115	62	40	109	18	
<b>Центральный административный округ</b>		364	50	87	464	428	113	670	66	
<b>Юго-Восточный административный округ</b>		38	13	67	282	89	55	145	25	
<b>Юго-Западный административный округ</b>		38	27	61	238	96	64	168	17	
<b>Южный административный округ</b>		68	25	85	264	131	73	202	44	

In [44]:

```
# Взял за основу код из этой статьи - чтобы уместить все графики на одном
```

```
#https://sharkcoder.com/data-visualization/mpl-pie-charts

font_color = '#525252'
colors = ['#f7ecb0', '#ffb3e6', '#99ff99', '#66b3ff', '#c7b3fb', '#ff6666', '#f9c3b7']

fig, axes = plt.subplots(3, 3, figsize=(10, 10), facecolor='#e8f4f0')

for i, (idx, row) in enumerate(district_type.iterrows()):
    ax = axes[i // 3, i % 3]
    row = row[row.gt(row.sum() * .01)]
    ax.pie(row,
            labels=row.values,
            startangle=30,
            wedgeprops=dict(width=.5),
            colors=colors,
            textprops={'color':font_color})
    ax.set_title(idx, fontsize=8, color=font_color)

legend = plt.legend([x[1] for x in row.index],
                    bbox_to_anchor=(1.1, 2.5), # Legend position
                    loc='lower left',
                    ncol=1,
                    fancybox=True)
for text in legend.get_texts():
    plt.setp(text, color=font_color) # Legend font color

fig.subplots_adjust(wspace=.2) # Space between charts

title = fig.suptitle('Распределение категорий заведений по районам Москвы', y=.95, fontsize=14)
# To prevent the title from being cropped
plt.subplots_adjust(top=0.85, bottom=0.15)
```

# Распределение категорий заведений по районам Москвы



Судя по графикам - самые популярные заведения в каждом районе Москвы -

- Кафе
- Кофейни
- Рестораны

В ЦАО значительно больше ресторанов, в отличии от других административных округов.

Связано с тем, что в центре больше туристов и платежеспособного населения

Построим **дополнительный график** для более наглядной визуализации распределения -

In [45]:

```
category_district = (df.groupby(['district', 'category'], as_index=False)
                     .agg(count_object=('name', 'count')))

category_district['sum_object'] = category_district.groupby('district')['count_object'].transform('sum')
category_district['ratio'] = round(category_district['count_object'] / category_district['sum_object'], 3)
category_district.head(10)
```

Out[45]:

	district	category	count_object	sum_object	ratio
0	Восточный административный округ	бар,паб	53	798	0.066
1	Восточный административный округ	булочная	25	798	0.031
2	Восточный административный округ	быстрое питание	71	798	0.089

	district	category	count_object	sum_object	ratio
3	Восточный административный округ	кафе	272	798	0.341
4	Восточный административный округ	кофейня	105	798	0.132
5	Восточный административный округ	пиццерия	72	798	0.090
6	Восточный административный округ	ресторан	160	798	0.201
7	Восточный административный округ	столовая	40	798	0.050
8	Западный административный округ	бар,паб	50	851	0.059
9	Западный административный округ	булочная	37	851	0.043

In [46]:

```
fig = px.bar(category_district,
              x='ratio',
              y='district',
              color='category',
              title='Распределение заведений в разрезе категорий по округам',
              text=round(category_district['ratio']*100, 2))

fig.update_layout(
    yaxis_title="Район",
    xaxis_title="%",
    yaxis=dict(
        tickmode='array',
        tickvals=category_district['district'],
        ticktext=category_district['district']
    ),
    legend_title='Категории заведений',
    autosize=False,
    width=1000, height=500
)

fig.show()
```

## Распределение заведений в разрезе категорий по округам



## Визуализируйте распределение средних рейтингов по категориям заведений.

Сильно ли различаются усреднённые рейтинги в разных типах общепита?

In [47]:

```
category_mean = df.groupby('category', as_index=False) ['rating'] \
    .mean() \
    .sort_values('rating', ascending=False) \
    .reset_index(drop=True)

category_mean['rating'] = round(category_mean['rating'], 2)
category_mean
```

Out[47]:

	category	rating
0	бар,паб	4.39
1	пиццерия	4.30
2	ресторан	4.29
3	кофейня	4.28
4	булочная	4.27
5	столовая	4.21
6	кафе	4.12
7	быстрое питание	4.05

In [48]:

```
plt.figure(figsize=(7, 5))
ax = sns.barplot(data=category_mean, x='rating', y='category')
plt.title('Средний рейтинг по типу заведения')
ax.set(xlabel='Рейтинг', ylabel='Категории заведений')
plt.show()
```



Как мы видим, самый высокий средний рейтинг у баров(пабов), пиццерий, ресторанов, кофейн и булочных.

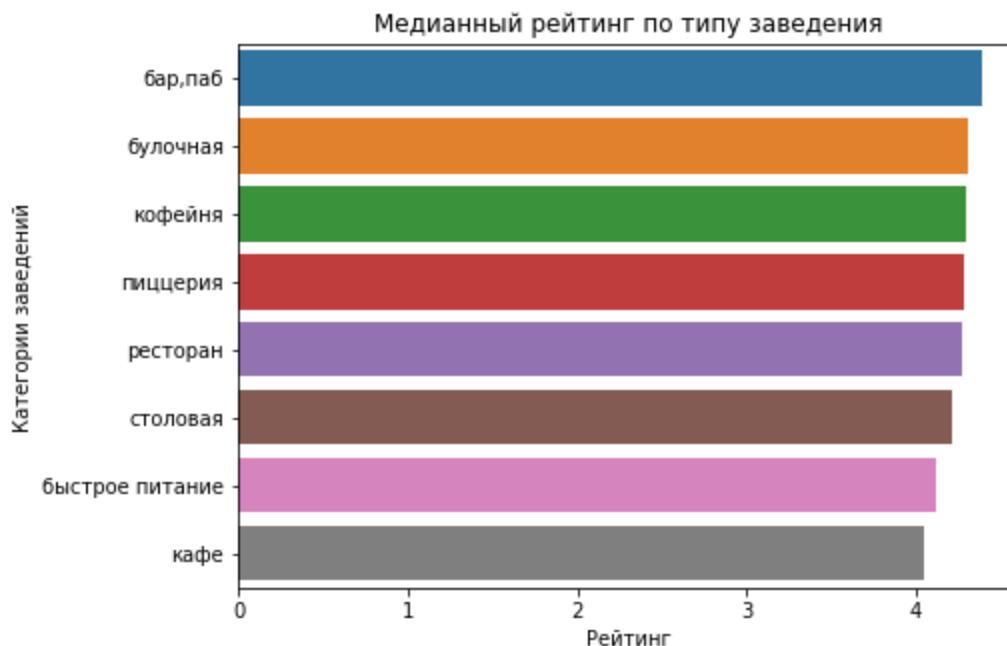
Проверим значение медианного рейтинга -

```
In [49]: category_median = df.groupby('category', as_index=False) ['rating']\n    .median() \\n    .sort_values('rating', ascending=False) \\n    .reset_index(drop=True)\n\n    category_median['rating'] = round(category_mean['rating'],2)\n    category_median
```

```
Out[49]:
```

	category	rating
0	бар,паб	4.39
1	булочная	4.30
2	кофейня	4.29
3	пиццерия	4.28
4	ресторан	4.27
5	столовая	4.21
6	быстрое питание	4.12
7	кафе	4.05

```
In [50]: plt.figure(figsize=(7, 5))\nax = sns.barplot(data=category_median,x='rating',y='category')\nplt.title('Медианный рейтинг по типу заведения')\nax.set(xlabel='Рейтинг', ylabel='Категории заведений')\nplt.show()
```



По медианному рейтингу - бары(пабы) - также заняли первое место. Места со 2 по 5 заняли такие же типы заведений, но в другом порядке.

Связано с тем, что при расчете среднего аномально большие/малые значения рейтинга могут повлиять на расчет.

## Постройте фоновую картограмму (хороплет) со средним рейтингом заведений каждого района.

In [51]:

```
rating_df = df.groupby('district', as_index=False) ['rating'].median()  
rating_df
```

Out[51]:

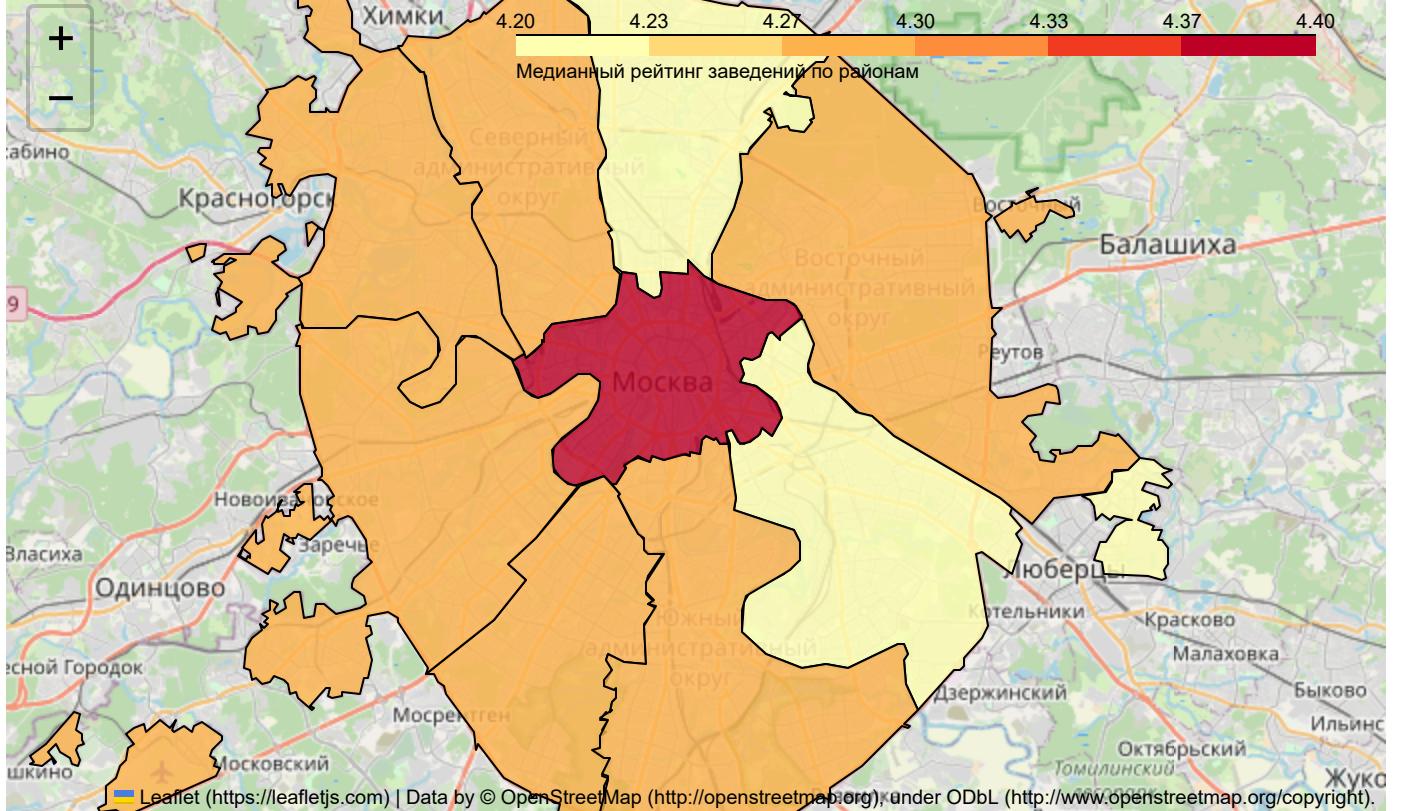
	district	rating
0	Восточный административный округ	4.3
1	Западный административный округ	4.3
2	Северный административный округ	4.3
3	Северо-Восточный административный округ	4.2
4	Северо-Западный административный округ	4.3
5	Центральный административный округ	4.4
6	Юго-Восточный административный округ	4.2
7	Юго-Западный административный округ	4.3
8	Южный административный округ	4.3

Подключим JSON - файл для отображения границ районов Москвы (добавим локальный и внешний путь к файлу) -

In [52]:

```
# импортируем карту и хороплет  
from folium import Map, Choropleth  
  
# загружаем JSON-файл с границами округов Москвы  
try:  
    with open('admin_level_geomap.geojson', 'r', encoding='utf-8') as f:  
        geo_json=json.load(f)  
except:  
    with open('/datasets/admin_level_geomap.geojson', 'r', encoding='utf-8') as f:  
        geo_json=json.load(f)  
  
# moscow_lat - широта центра Москвы, moscow_lng - долгота центра Москвы  
moscow_lat, moscow_lng = 55.751244, 37.618423  
  
# создаём карту Москвы  
m = Map(location=[moscow_lat, moscow_lng], zoom_start=10)  
  
# создаём хороплет с помощью конструктора Choropleth и добавляем его на карту  
Choropleth(  
    geo_data=geo_json,  
    data=rating_df,  
    columns=['district', 'rating'],  
    key_on='feature.name',  
    fill_color='YlOrRd',  
    fill_opacity=0.8,  
    legend_name='Медианный рейтинг заведений по районам',  
).add_to(m)  
  
# выводим карту  
m
```

Out[52]:



Как мы видим по тепловой карте - **лучшие по рейтингу заведения Москвы сосредоточены в центре города.**

Кафе с минимальным рейтингом - 4.2 - находятся Юго-Восточном и Северо-Восточном административных округах.

**Отобразите все заведения датасета на карте с помощью кластеров средствами библиотеки folium.**

In [53]:

```
# moscow_lat - широта центра Москвы, moscow_lng - долгота центра Москвы
moscow_lat, moscow_lng = 55.751244, 37.618423

# создаём карту Москвы
m = Map(location=[moscow_lat, moscow_lng], zoom_start=10)
# создаём пустой кластер, добавляем его на карту
marker_cluster = MarkerCluster().add_to(m)

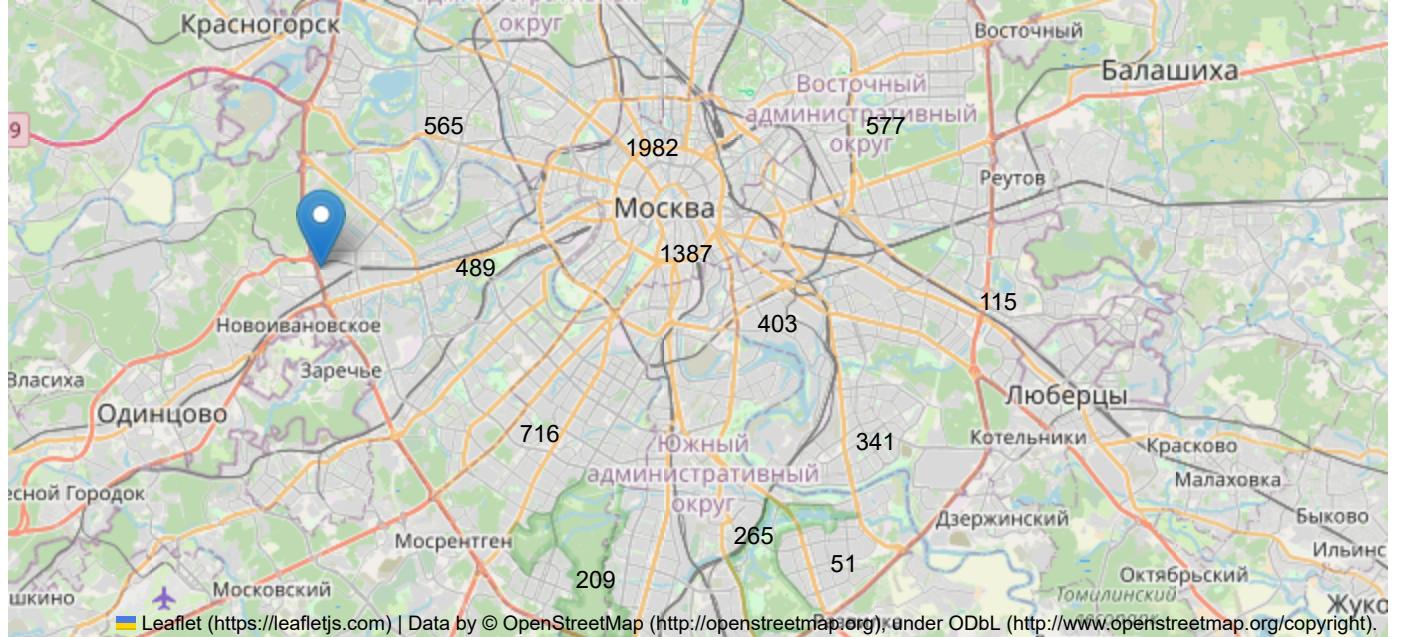
# пишем функцию, которая принимает строку датафрейма,
# создаёт маркер в текущей точке и добавляет его в кластер marker_cluster
def create_clusters(row):
    Marker(
        [row['lat'], row['lng']],
        popup=f'{row['name']} {row['rating']}',
    ).add_to(marker_cluster)

# применяем функцию create_clusters() к каждой строке датафрейма
df.apply(create_clusters, axis=1)

# выводим карту
m
```

Out[53]:





Как мы видим на карте - самое **большое количество заведений в центре сосредоточено в центре города.**

## Найдите топ-15 улиц по количеству заведений.

Постройте график распределения количества заведений и их категорий по этим улицам.

In [54]:

```
top15_street = df.groupby('street', as_index = False) ['name'].count() \
    .sort_values(by='name', ascending = False).reset_index(drop=True).head(15)
top15_street
```

Out[54]:

	street	name
0	проспект мира	184
1	профсоюзная улица	122
2	проспект вернадского	108
3	ленинский проспект	107
4	ленинградский проспект	95
5	дмитровское шоссе	88
6	каширское шоссе	77
7	варшавское шоссе	76
8	ленинградское шоссе	70
9	мкад	65
10	люблинская улица	60
11	улица вавилова	55
12	кутузовский проспект	54
13	улица миклухо-маклая	49
14	пятницкая улица	48

In [55]:

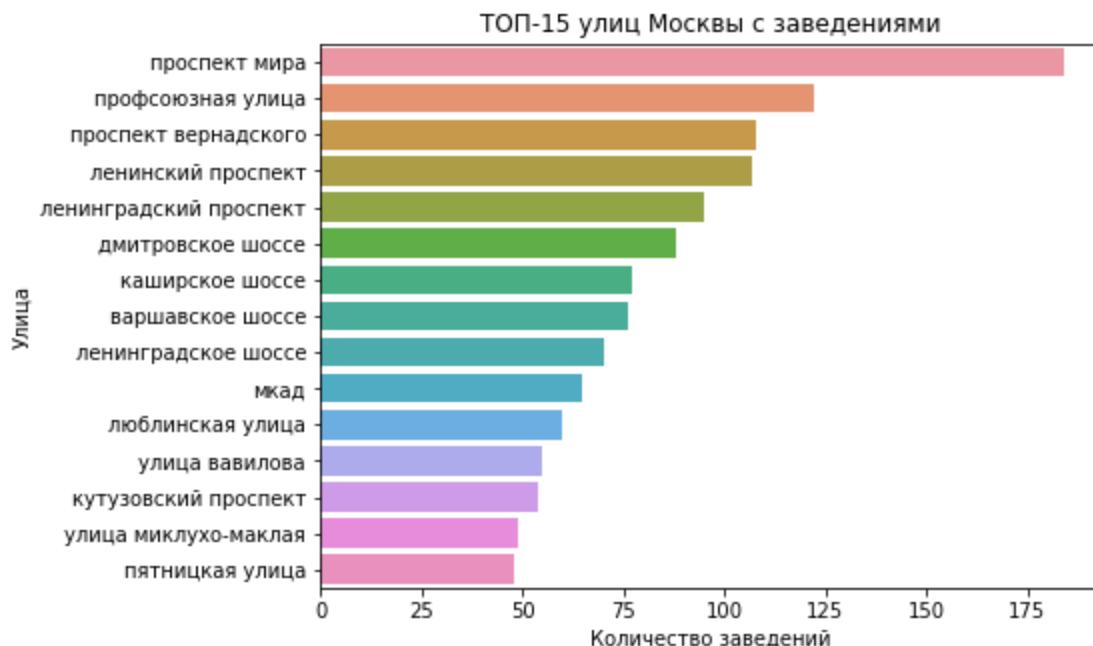
```
plt.figure(figsize=(7, 5))
```

```

ax = sns.barplot(data=top15_street,y='street',x='name')

plt.title('ТОП-15 улиц Москвы с заведениями')
ax.set(ylabel='Улица', xlabel='Количество заведений')
plt.show()

```



ТОП-5 улиц по количеству заведений -

- Проспект Мира
- Профсоюзная улица
- Проспект Вернадского
- Ленинский проспект
- Ленинградский проспект

Посмотрим на типы заведений на этих улицах

In [56]:

```

streets = top15_street['street'].tolist()
streets

df_top15 = df.query('street in @streets')
top15_with_category = df_top15.groupby(['street','category'], as_index=False) ['name'].count()
top15_with_category['sum_object'] = top15_with_category.groupby('street')['name'].transform('sum')
top15_with_category['ratio'] = round(top15_with_category['name'] / top15_with_category['sum_object'], 3)
top15_with_category

```

Out[56]:

	street	category	name	sum_object	ratio
0	варшавское шоссе	бар,лаб	6	76	0.079
1	варшавское шоссе	быстрое питание	7	76	0.092
2	варшавское шоссе	кафе	18	76	0.237
3	варшавское шоссе	кофейня	14	76	0.184
4	варшавское шоссе	пиццерия	4	76	0.053
...	...	...	...	...	...
106	улица миклухо-маклая	быстрое питание	4	49	0.082
107	улица миклухо-маклая	кафе	21	49	0.429

	street	category	name	sum_object	ratio
108	улица миклухо-маклая	кофейня	4	49	0.082
109	улица миклухо-маклая	пиццерия	2	49	0.041
110	улица миклухо-маклая	ресторан	15	49	0.306

111 rows × 5 columns

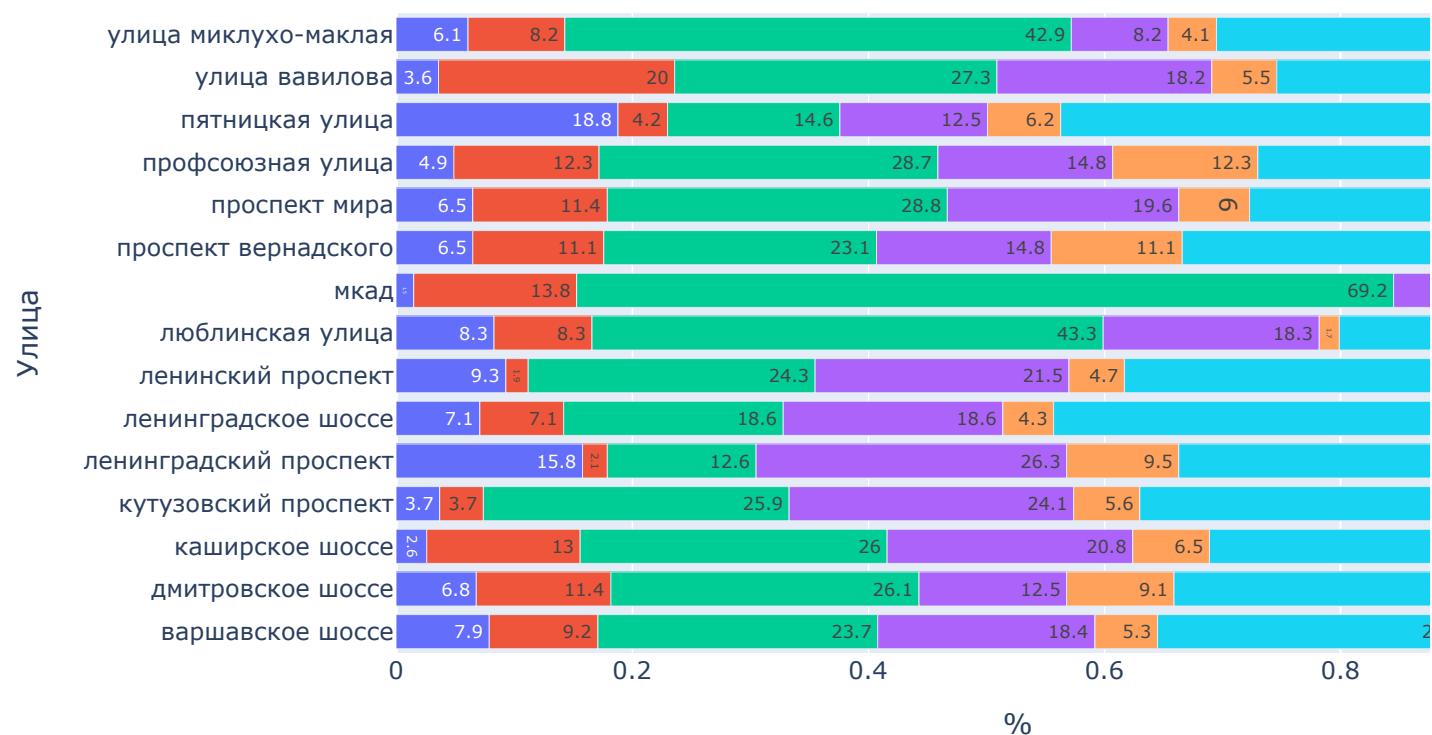
In [57]:

```
fig = px.bar(top15_with_category,
              x='ratio',
              y='street',
              color='category',
              title='Типы заведений на топ-15 улиц Москвы',
              text=round(top15_with_category['ratio']*100, 2))

fig.update_layout(
    yaxis_title="Улица",
    xaxis_title="%",
    yaxis=dict(
        tickmode='array',
        tickvals=top15_with_category['street'],
        ticktext=top15_with_category['street']
    ),
    legend_title='Категории заведений',
    autosize=False,
    width=1000, height=500
)

fig.show()
```

## Типы заведений на топ-15 улиц Москвы



Как мы видим, на топ-15 улиц Москвы - преимущественно стоят кафе, рестораны и кофейни

## Найдите улицы, на которых находится только один объект общепита.

Что можно сказать об этих заведениях?

In [58]:

```
streets_one_object = df.groupby('street', as_index = False) ['name'].count()
streets_one_object.columns = ('street', 'count')
streets_one_object = streets_one_object[streets_one_object['count']==1].reset_index(drop=True)
streets_one_object
```

Out[58]:

	street	count
0	1-й автозаводский проезд	1
1	1-й балтийский переулок	1
2	1-й варшавский проезд	1
3	1-й вешняковский проезд	1
4	1-й голутвинский переулок	1
...	...	...
452	электрический переулок	1
453	юго-западный административный округ	1
454	якиманский переулок	1
455	яковоапостольский переулок	1
456	№ 7	1

457 rows × 2 columns

Как мы видим, в списке 457 улиц Москвы, где находится только один объект общепита

Посмотрим на категории данных заведений -

In [59]:

```
streets = streets_one_object['street'].tolist()
streets

df_1_object = df.query('street in @streets')
df_1_object = df_1_object.groupby(['category'], as_index=False) ['name'].count() \
    .sort_values('name', ascending=False).reset_index(drop=True)
df_1_object
```

Out[59]:

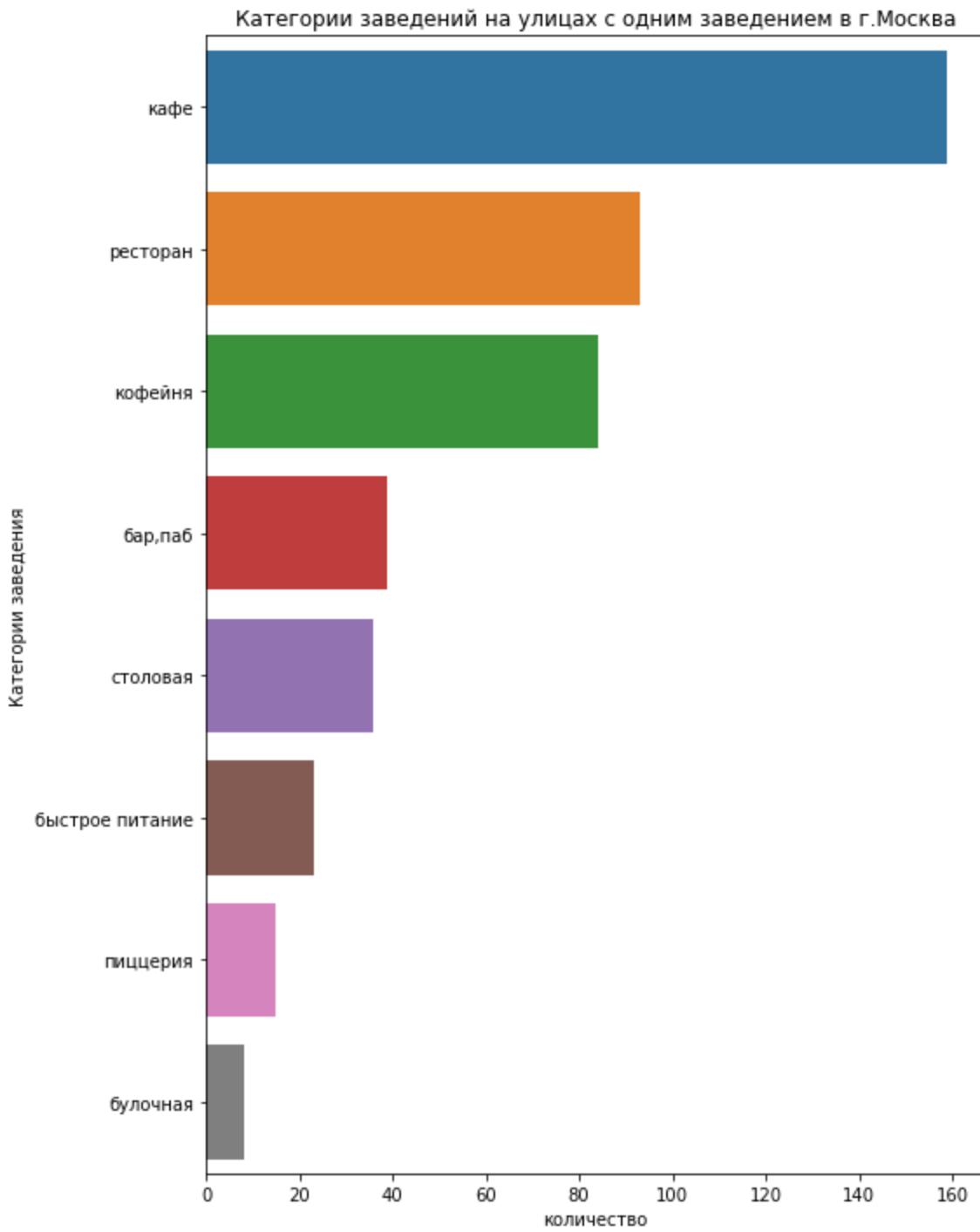
	category	name
0	кафе	159
1	ресторан	93
2	кофейня	84
3	бар,паб	39
4	столовая	36
5	быстрое питание	23
6	пиццерия	15

category name

7	булочная	8
---	----------	---

In [60]:

```
plt.figure(figsize=(8, 12))
ax = sns.barplot(y="category", x="name", data=df_1_object);
sns.color_palette('bright')
plt.title('Категории заведений на улицах с одним заведением в г.Москва')
ax.set(ylabel='Категории заведения', xlabel='количество')
plt.show()
```



Как мы видим, если на улице есть только одно заведение Общепита - чаще всего это кафе.

На втором и третьем месте идут кофейни и рестораны.

Посмотрим на средний рейтинг данных заведений -

```
In [61]: df_1_object = df.query('street in @streets')
df_1_object = df_1_object.groupby(['rating'], as_index=False) ['name'].count() \
    .sort_values('name', ascending=False).reset_index(drop=True).head(5)
df_1_object
```

```
Out[61]:   rating  name
0      4.3     84
1      4.4     56
2      4.2     54
3      4.1     49
4      4.7     28
```

Большая часть заведений на улицах, где представлено только одно заведение, имеет рейтинг - **4.1 - 4.4**

**Значения средних чеков заведений хранятся в столбце middle\_avg\_bill. Эти числа показывают примерную стоимость заказа в рублях, которая чаще всего выражена диапазоном. Посчитайте медиану этого столбца для каждого района.**

Используйте это значение в качестве ценового индикатора района. Постройте фоновую картограмму (хороплет) с полученными значениями для каждого района.

Проанализируйте цены в центральном административном округе и других.

Как удалённость от центра влияет на цены в заведениях?

```
In [62]: rating_df = df.groupby('district', as_index=False) ['middle_avg_bill'].median() \
    .sort_values('middle_avg_bill', ascending = False).reset_index(drop=True)
rating_df
```

```
Out[62]:          district  middle_avg_bill
0      Западный административный округ      1000.0
1      Центральный административный округ      1000.0
2  Северо-Западный административный округ      700.0
3      Северный административный округ      650.0
4  Юго-Западный административный округ      600.0
5      Восточный административный округ      575.0
6  Северо-Восточный административный округ      500.0
7      Южный административный округ      500.0
8  Юго-Восточный административный округ      450.0
```

```
In [63]: # moscow_lat - широта центра Москвы, moscow_lng - долгота центра Москвы
moscow_lat, moscow_lng = 55.751244, 37.618423

# создаём карту Москвы
m = Map(location=[moscow_lat, moscow_lng], zoom_start=10)

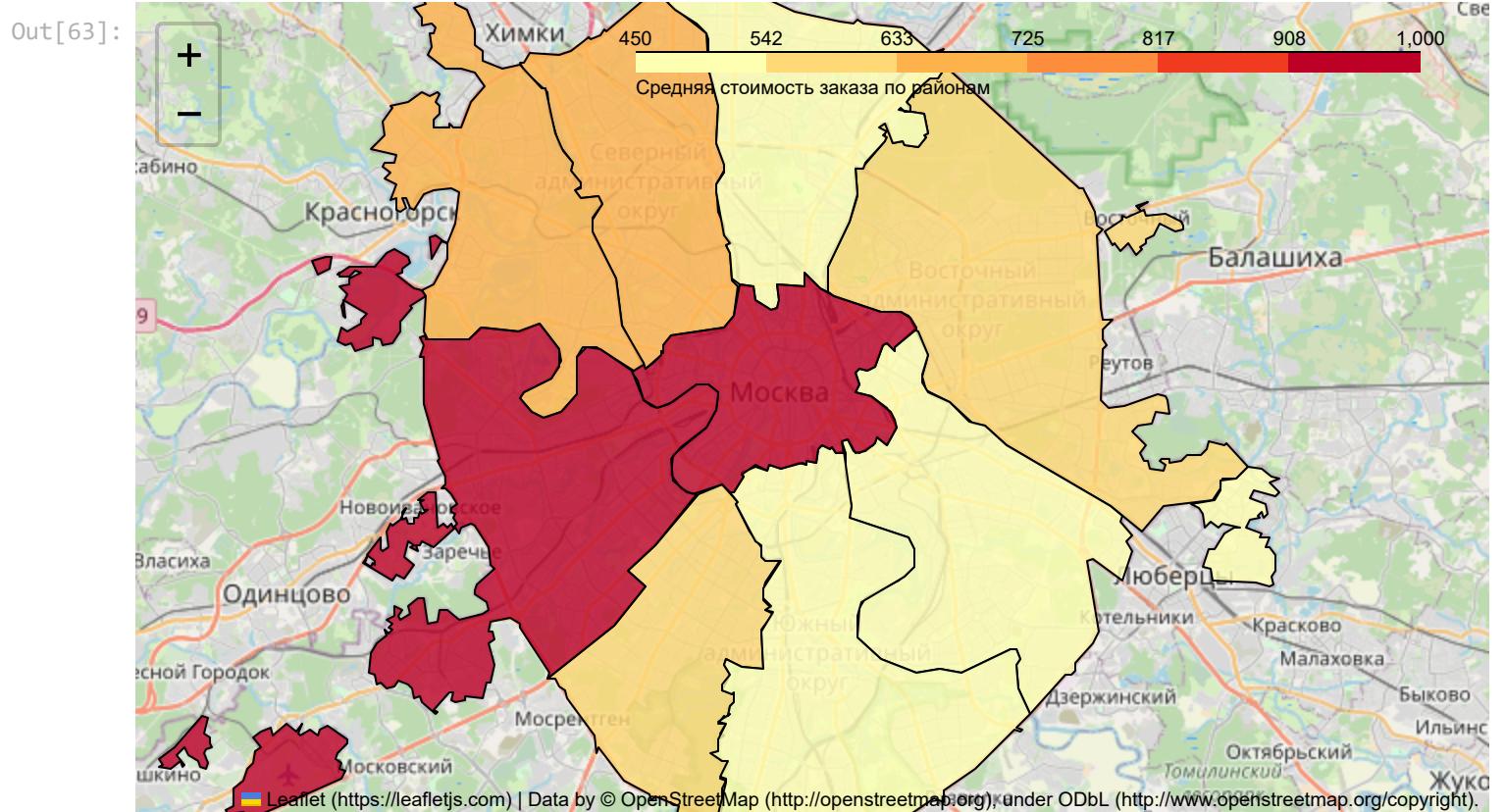
# создаём хороплет с помощью конструктора Choropleth и добавляем его на карту
```

```

Choropleth(
    geo_data=geo_json,
    data=rating_df,
    columns=['district', 'middle_avg_bill'],
    key_on='feature.name',
    fill_color='YlOrRd',
    fill_opacity=0.8,
    legend_name='Средняя стоимость заказа по районам',
).add_to(m)

# выводим карту
m

```



Судя по полученному хороплету -

**самая высокая цена среднего заказа в центре Москвы (ЦАО), а также в ЗАО (западный административный округ)**

Районы с самым низким средним чеком -

1. Северо-Восточный административный округ
2. Южный административный округ
3. Юго-Восточный административный округ

**Посмотрим как распределены круглосуточные заведения на карте Москвы**

In [64]:

```

rating_df = df[df['is_24/7']==True].groupby('district',as_index=False)[['name']].count() \
    .sort_values('name', ascending = False).reset_index(drop=True)
rating_df

```

Out[64]:

	district	name
0	Центральный административный округ	131

	district	name	
1	Восточный административный округ	97	
2	Юго-Восточный административный округ	93	
3	Северо-Восточный административный округ	75	
4	Южный административный округ	75	
5	Юго-Западный административный округ	73	
6	Западный административный округ	72	
7	Северный административный округ	71	
8	Северо-Западный административный округ	43	

In [65]:

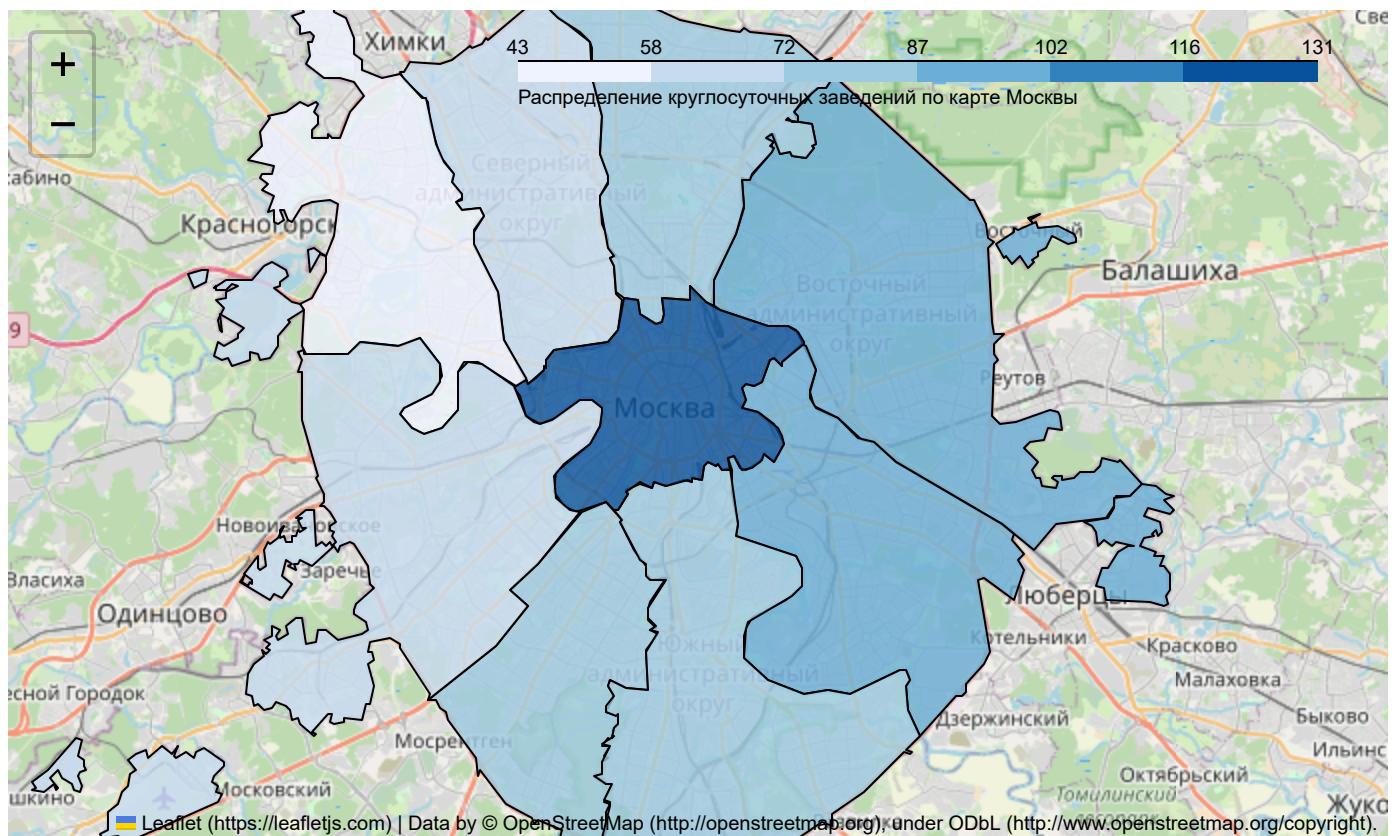
```
# moscow_lat - широта центра Москвы, moscow_lng - долгота центра Москвы
moscow_lat, moscow_lng = 55.751244, 37.618423

# создаём карту Москвы
m = Map(location=[moscow_lat, moscow_lng], zoom_start=10)

# создаём хороплет с помощью конструктора Choropleth и добавляем его на карту
Choropleth(
    geo_data=geo_json,
    data=rating_df,
    columns=['district', 'name'],
    key_on='feature.name',
    fill_color='Blues',
    fill_opacity=0.8,
    legend_name='Распределение круглосуточных заведений по карте Москвы',
).add_to(m)

# выводим карту
m
```

Out[65]:



Как мы видим, **большинство круглосуточных заведений находится в центре Москвы (район - ЦАО, сосредоточение туристов)**

На 2м и 3м месте по количеству заведений с графиком работы 24/7 находятся - Восточный административный округ и Юго-Восточный административный округ

## Посмотрим к каким категориям относятся круглосуточные заведения в Москве

In [66]:

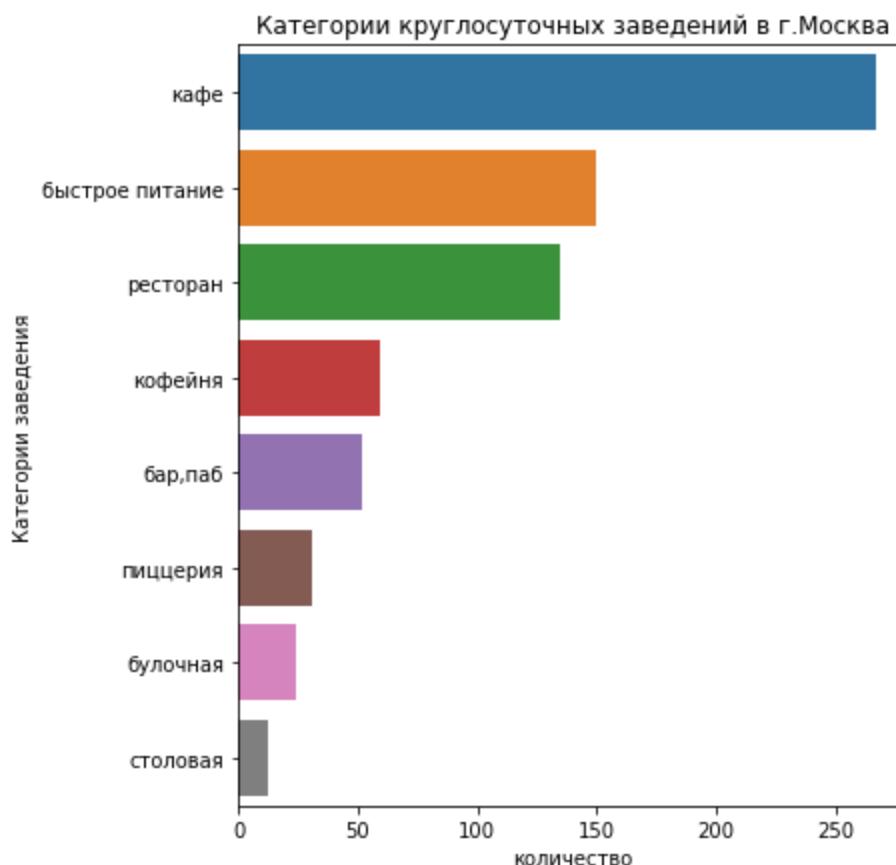
```
open_24 = df[df['is_24/7']==True].groupby('category',as_index=False) ['name'].count() \
    .sort_values('name',ascending=False).reset_index(drop=True)
open_24
```

Out[66]:

	category	name
0	кафе	267
1	быстрое питание	150
2	ресторан	135
3	кофейня	59
4	бар,паб	52
5	пиццерия	31
6	булочная	24
7	столовая	12

In [67]:

```
plt.figure(figsize=(6, 7))
ax = sns.barplot(y="category", x="name", data=open_24);
sns.color_palette('bright')
plt.title('Категории круглосуточных заведений в г.Москва')
ax.set(ylabel='Категории заведения', xlabel='количество')
plt.show()
```



Большинство круглосуточных заведений в Москве - кафе.

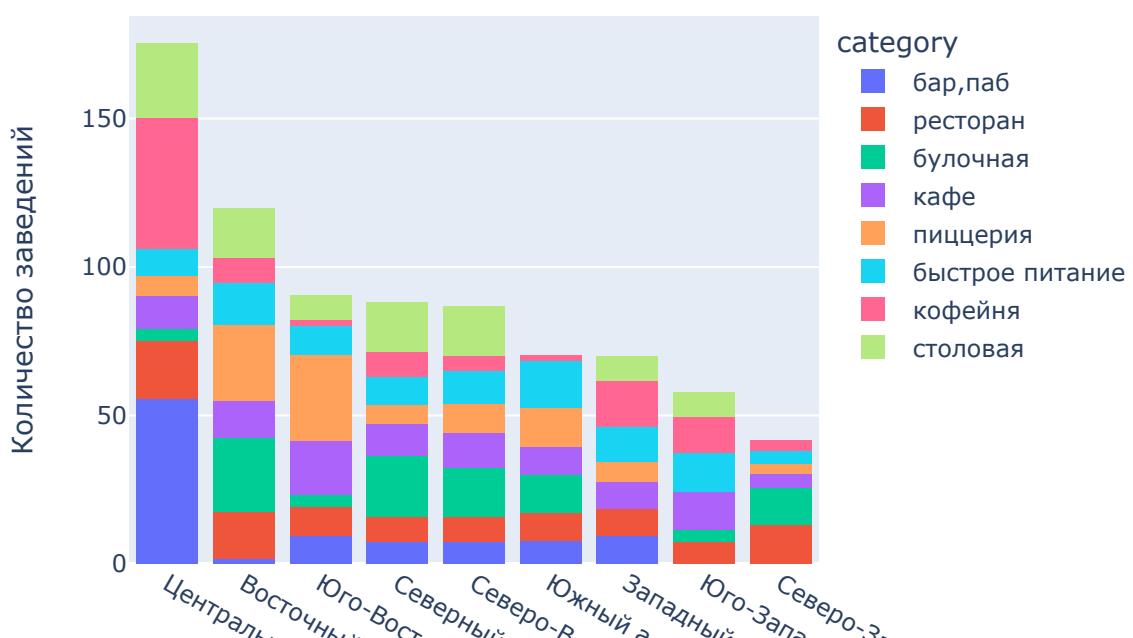
На 2м и 3м месте - быстрое питание и рестораны.

Покажем на графике - распределение круглосуточных заведений по районам и категориям

In [68]:

```
fig = px.histogram(df[df['is_24/7']==True], x="district", histnorm='percent', color="category"
    .update_xaxes(categoryorder='total descending')
    fig.update_xaxes(title_text='Район Москвы')
    fig.update_yaxes(title_text='Количество заведений')
    fig.update_layout(title_text='Круглосуточные заведения в Москве - по районам и типам', title_x=0.5)
    fig.show()
```

### Круглосуточные заведения в Москве - по районам и типам



Северо-западный административный округ  
Северо-восточный административный округ  
Южный административный округ  
Западный административный округ  
Северо-западный административный округ  
Северо-восточный административный округ  
Южный административный округ

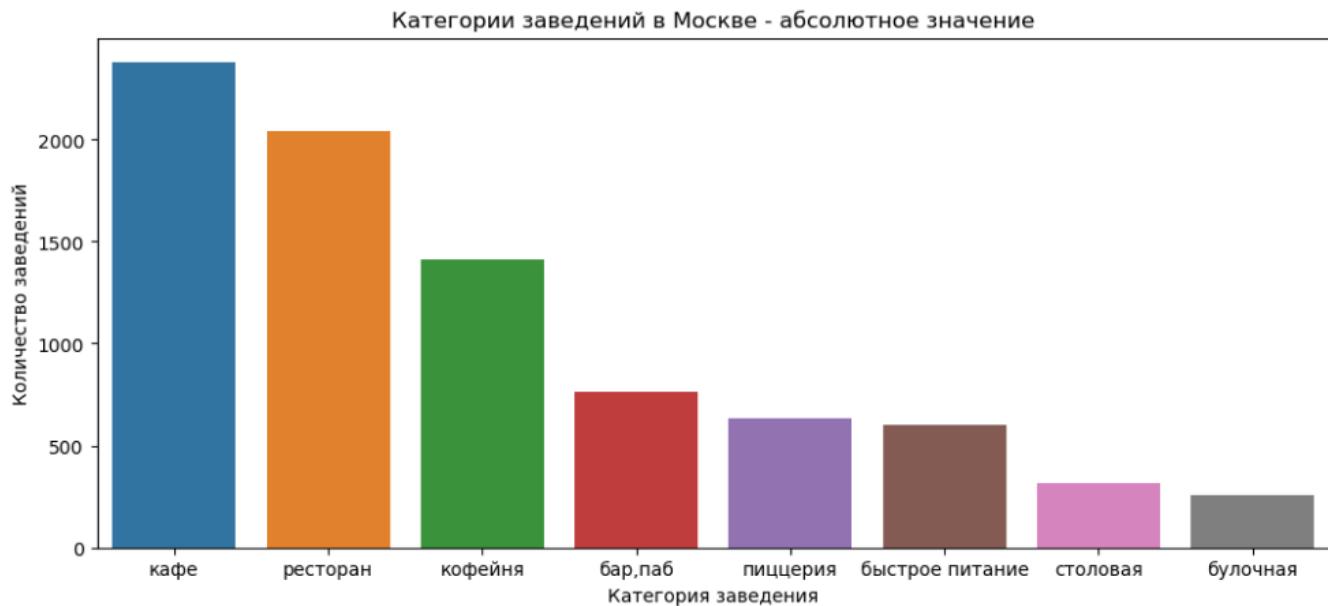
## Район Москвы

Судя по графику ЦАО вновь отличается от других районов -

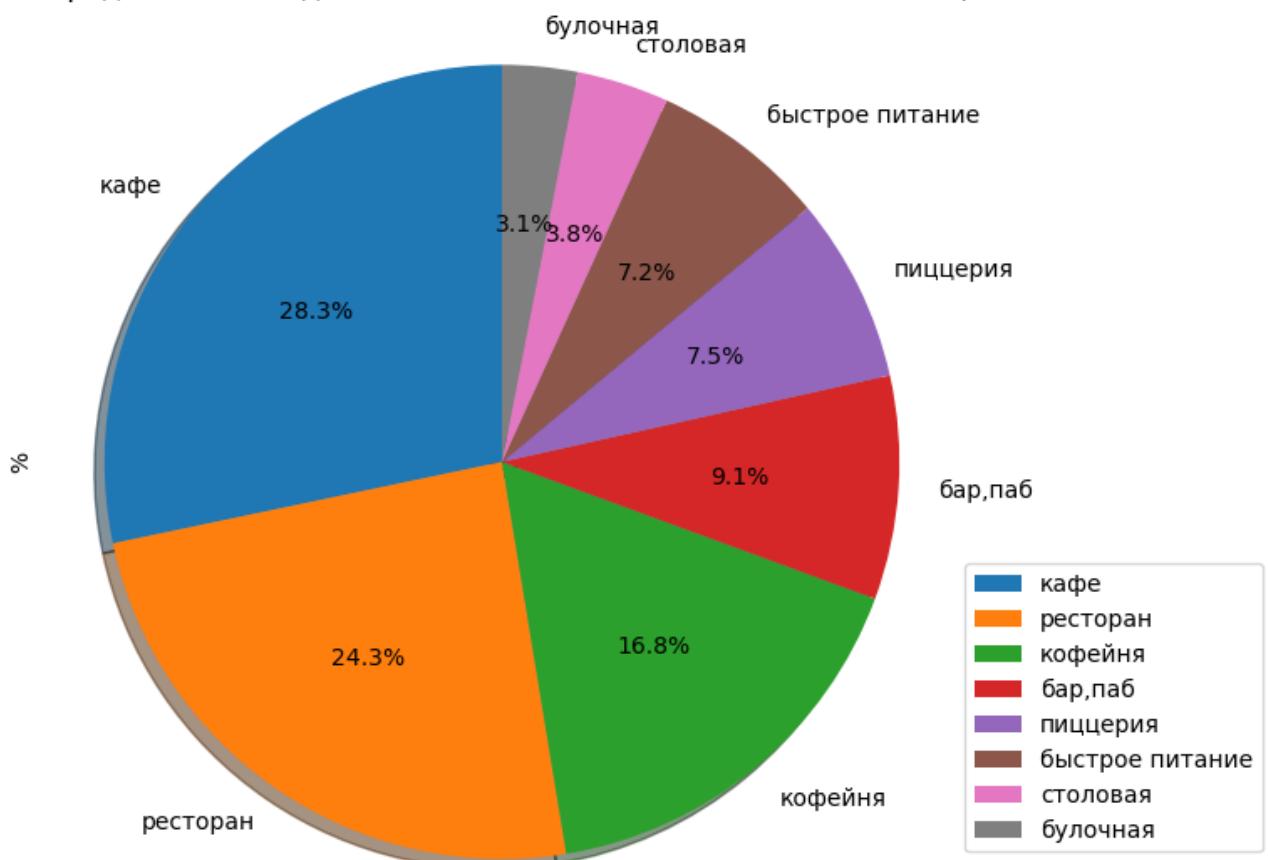
В ЦАО - преимущественно \*\*больше круглосуточных баров/пабов и ресторанов\*\*

### Вывод по Шагу №3. Анализ данных

- Мы изучили как распределены по количеству заведения Общепита в Москве, в зависимости от их категории



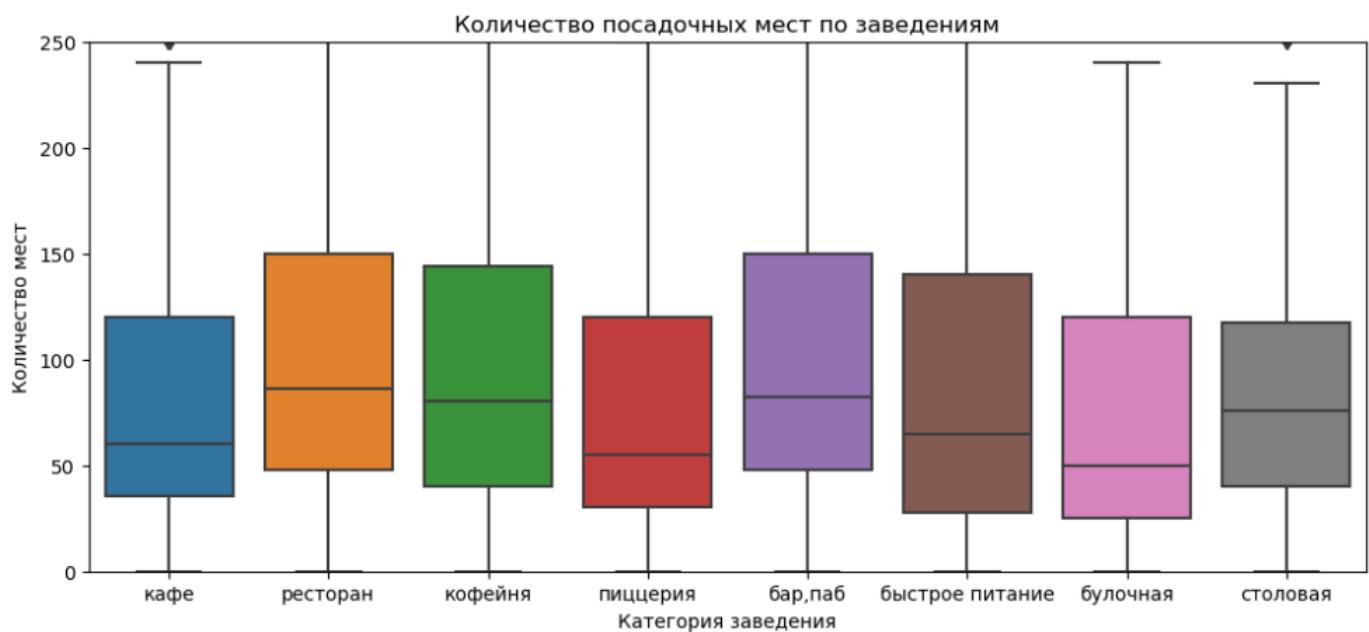
## Распределение заведений в Москве - относительное количество, %



Больше всего в Москве - кафе и ресторанов - 28.3% и 24.3% соответственно

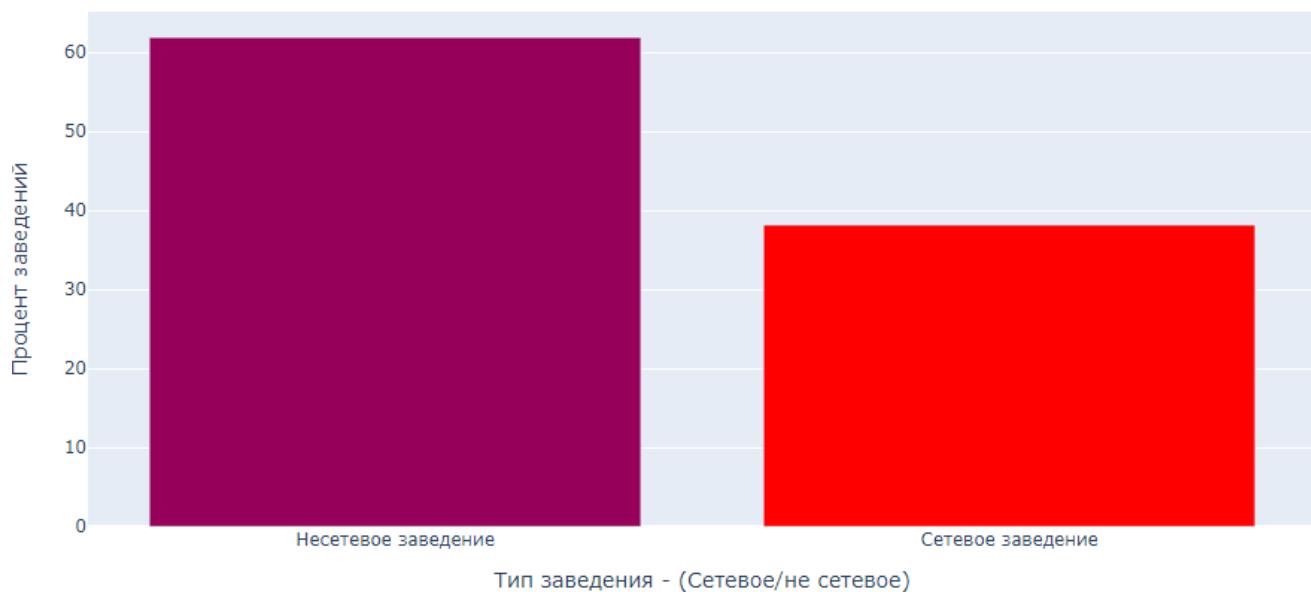
**медианное значение посадочных мест в больших заведениях (ресторан, бар/паб, кофейня, столовая) - составляет от 76 до 86 посадочных мест**

**В маленьких заведениях (Булочная, пиццерия, кафе, быстрое питание) - количество мест составляет - 55-65**



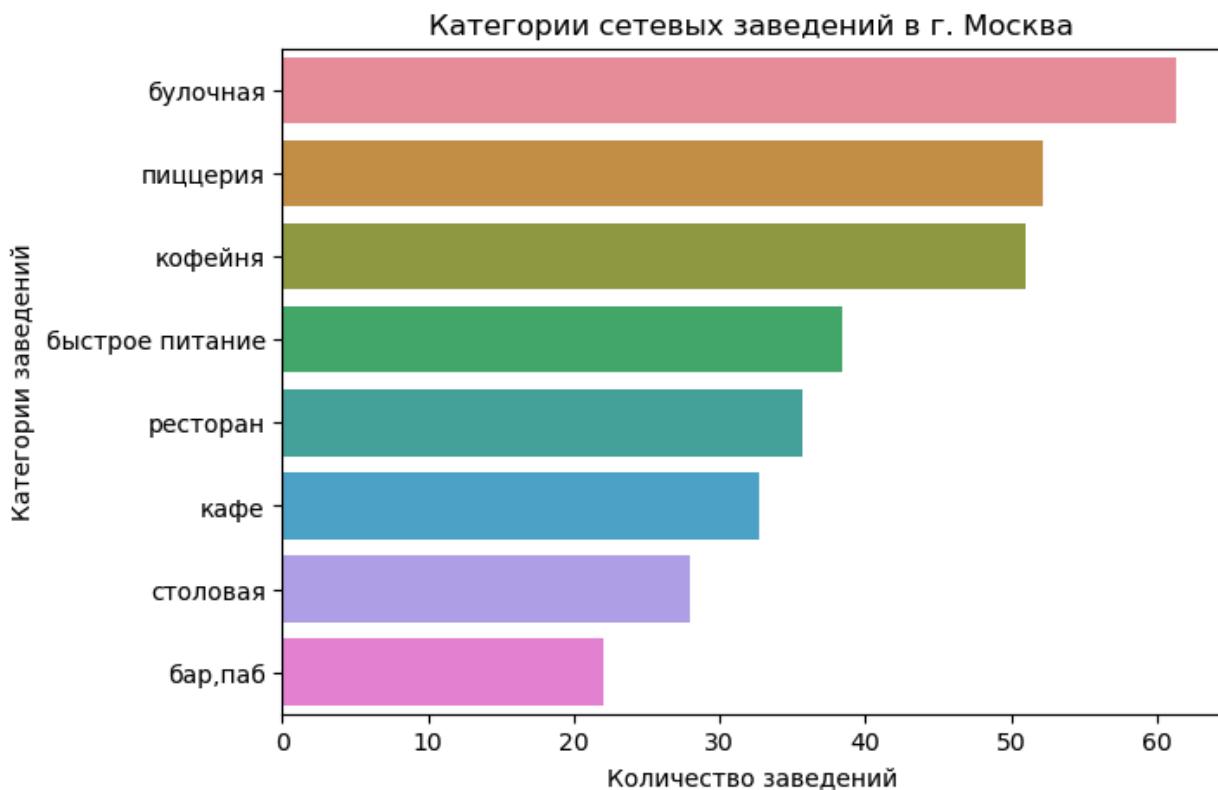
- Количество сетевых заведений составляет -62% , несетевых - 38%

Соотношение сетевых и несетевых заведений (относительное, %)



Чаще всего сетевыми заведениями являются -

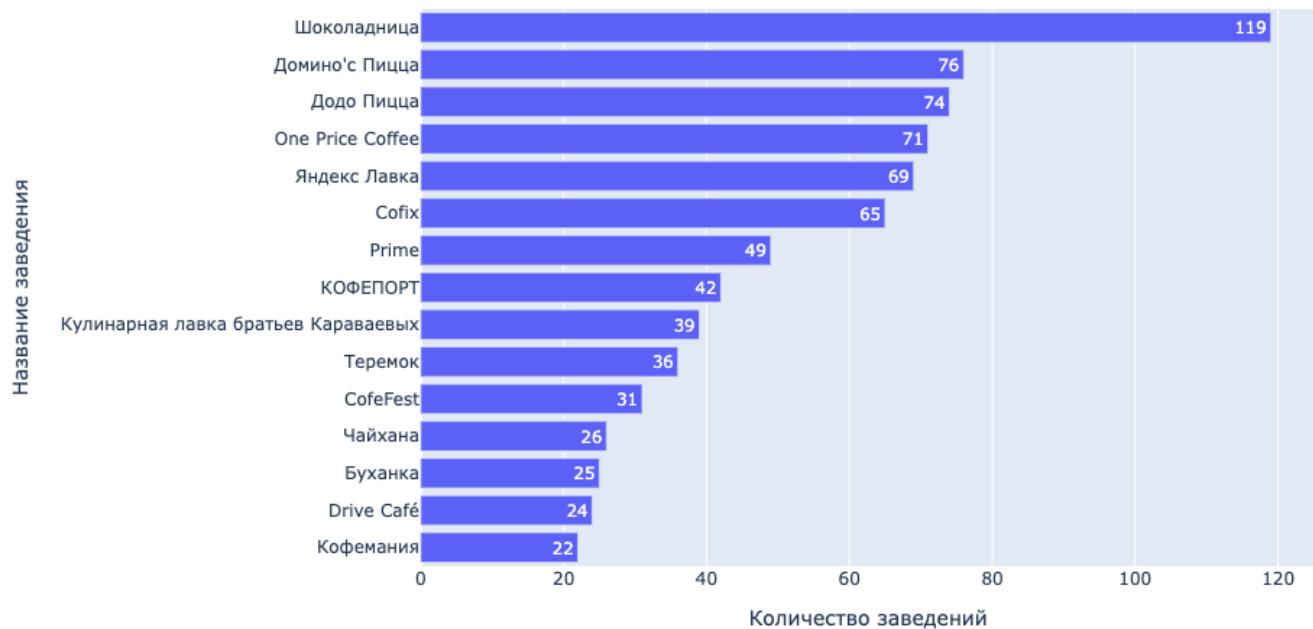
- Булочные
- Пиццерии
- Кофейни



		name	category	count
0		Шоколадница	кофейня	119
1		Домино'с Пицца	пиццерия	76
2		Додо Пицца	пиццерия	74
3		One Price Coffee	кофейня	71
4		Яндекс Лавка	ресторан	69
5		Cofix	кофейня	65
6		Prime	ресторан	49
7		КОФЕПОРТ	кофейня	42
8	Кулинарная лавка братьев Караваевых		кафе	39
9		Теремок	ресторан	36
10		CofeFest	кофейня	31
11		Чайхана	кафе	26
12		Буханка	булочная	25
13		Drive Café	кафе	24
14		Кофемания	кофейня	22

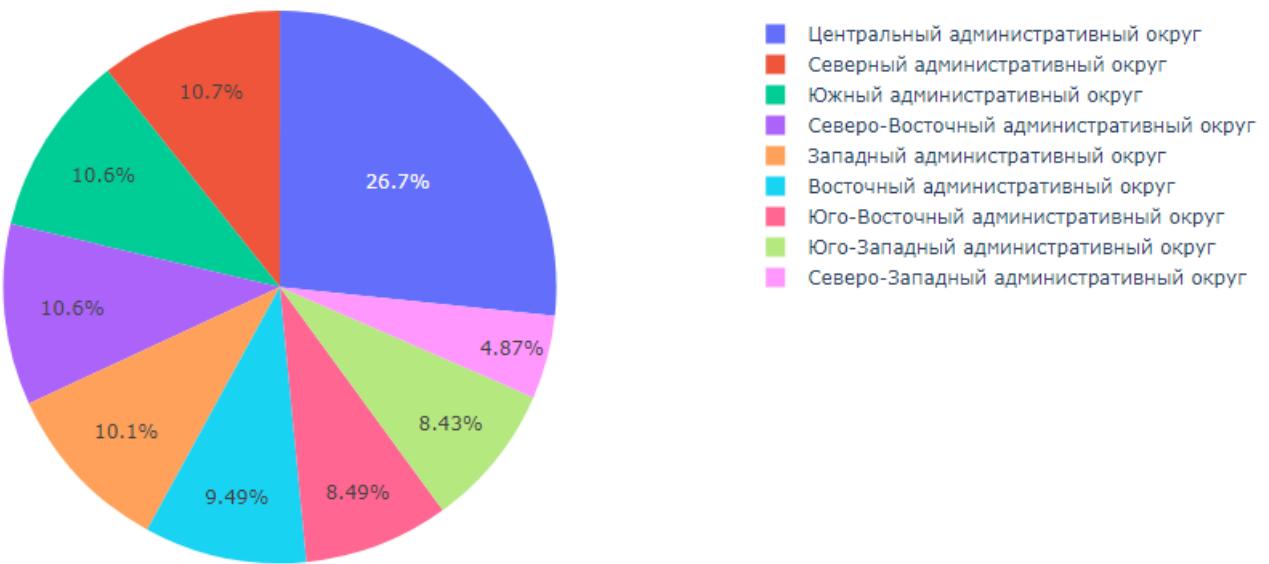
### Топ 15 сетевых заведений -

Топ 15 сетевых заведений в Москве



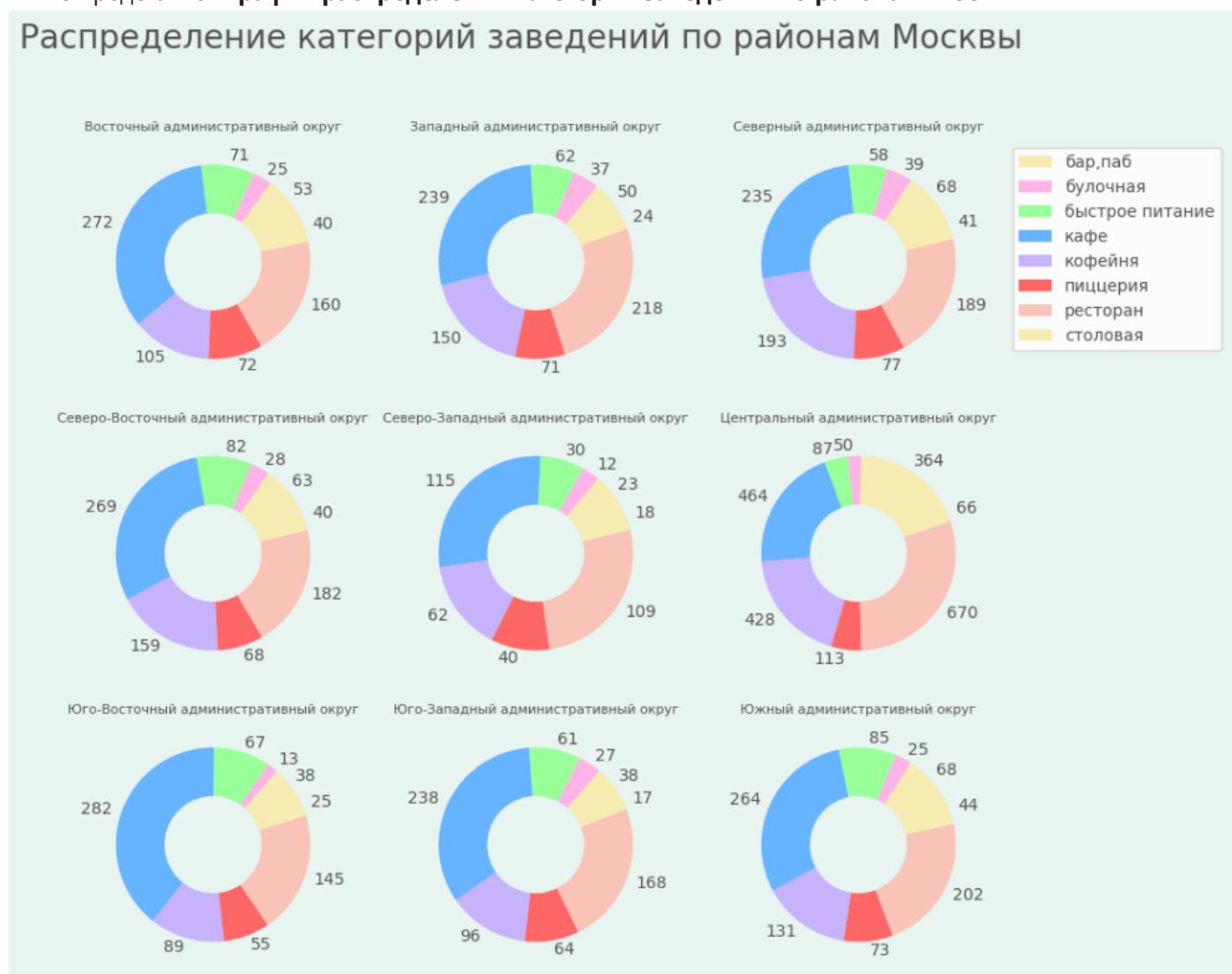
**Большая часть заведений Общепита находится в центральном административном округе (далее - ЦАО) -**, т.к. в ЦАО - основное скопление туристов и платежеспособных людей.

# Разделение заведений по районам Москвы

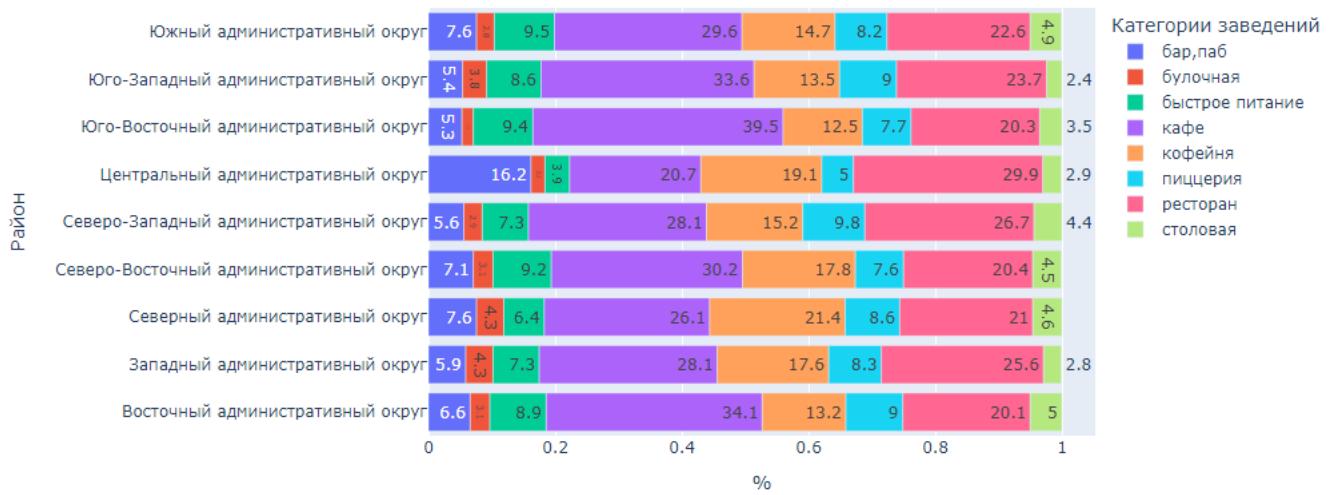


Ниже представлен **график распределения категорий заведений по районам Москвы** -

## Распределение категорий заведений по районам Москвы



# Распределение заведений в разрезе категорий по округам

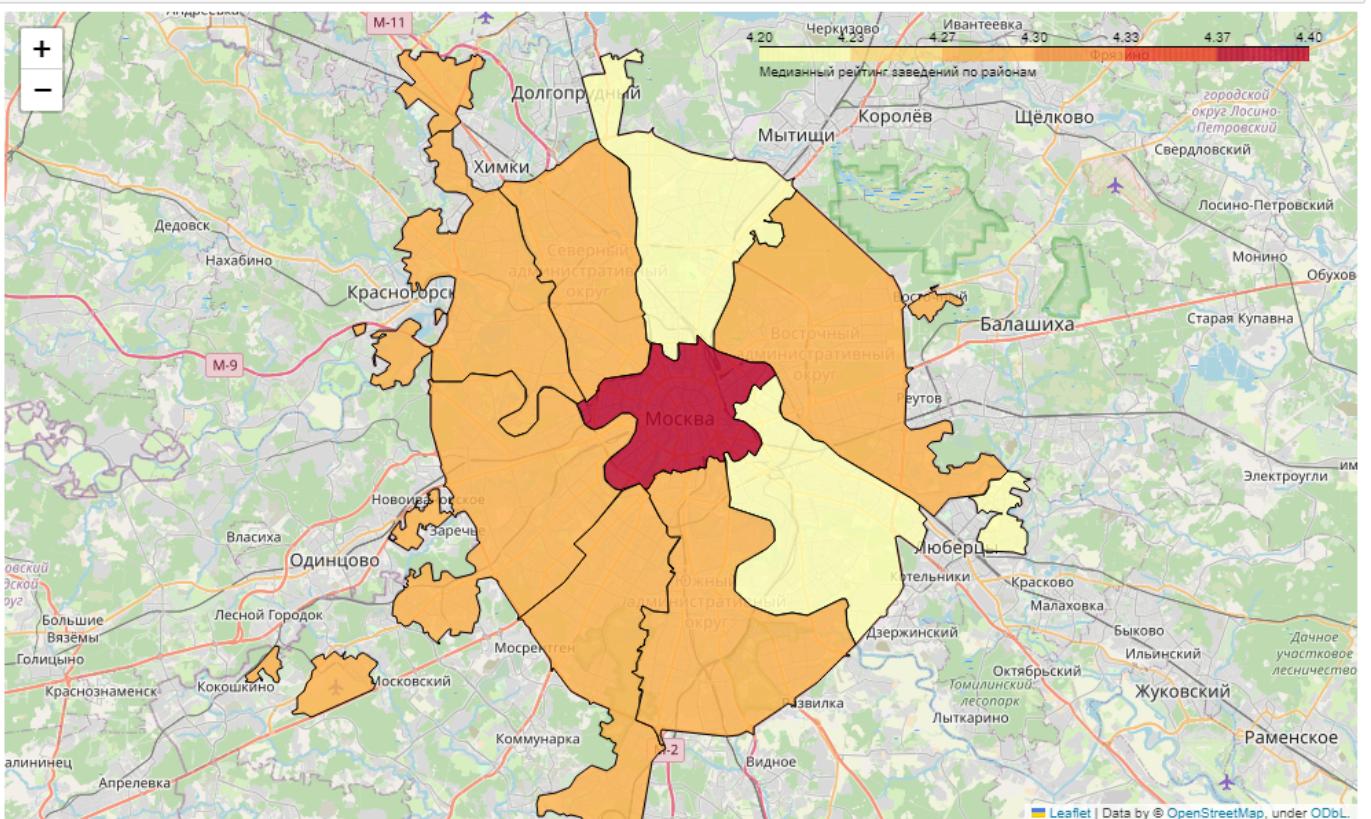


В ЦАО значительно больше ресторанов, в отличии от других административных округов.

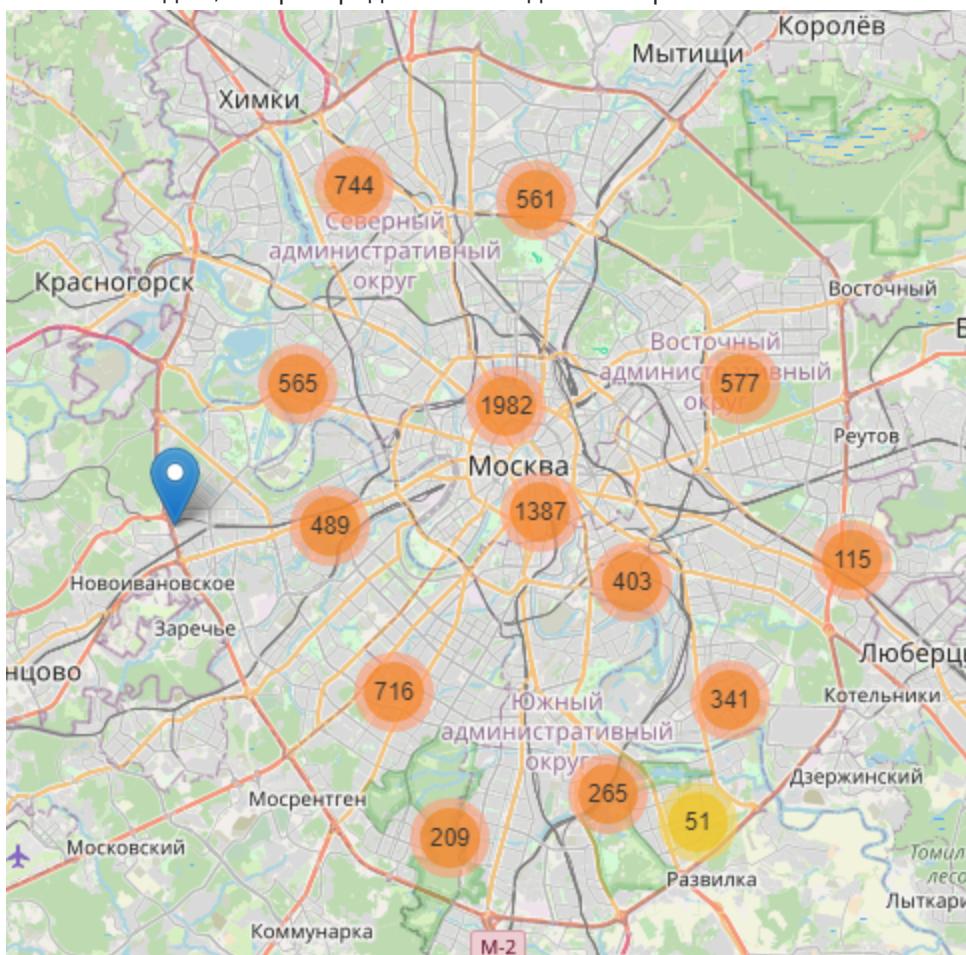
## Топ - 5 категорий мест по среднему рейтингу (медианному) -



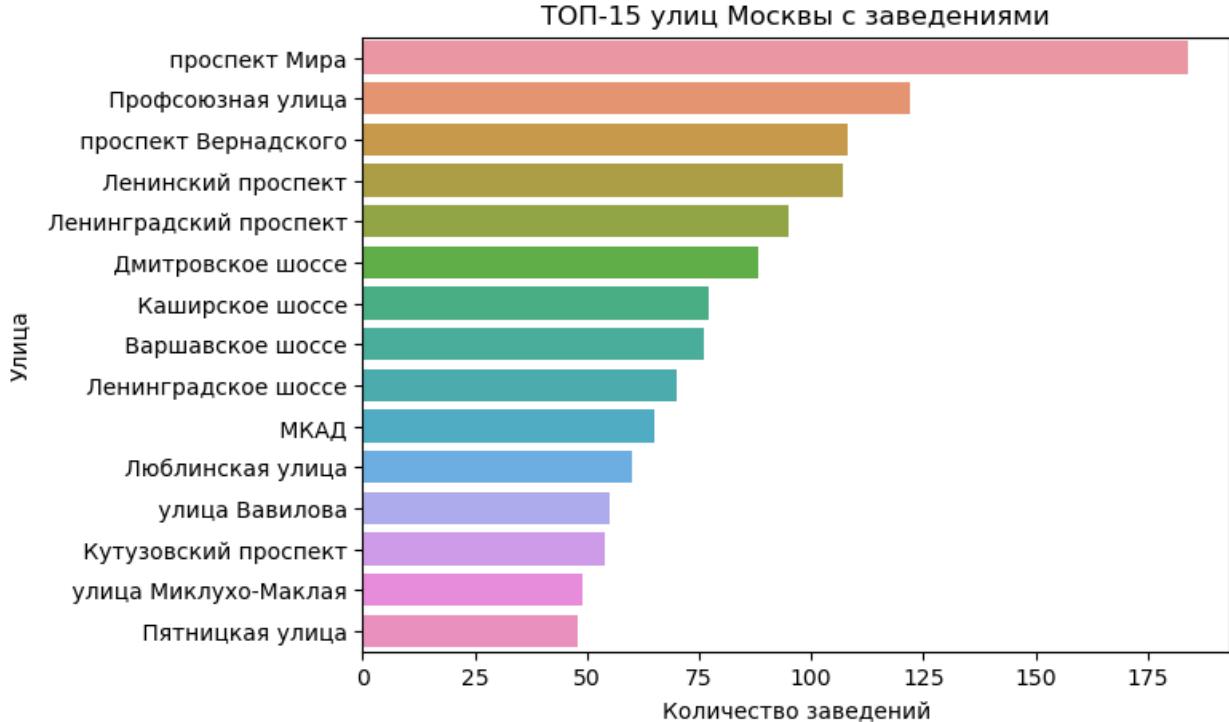
Лучшие по рейтингу заведения находятся в центре Москвы (район - ЦАО)



Ниже мы видим, как распределены заведения по районам Москвы

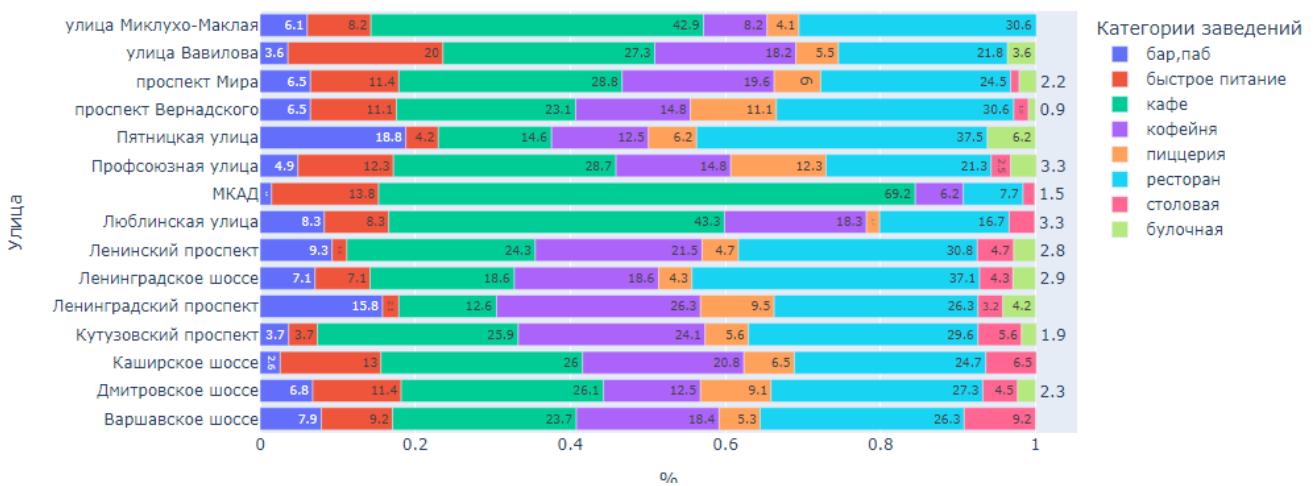


- Топ-15 улиц с заведениями Общепита



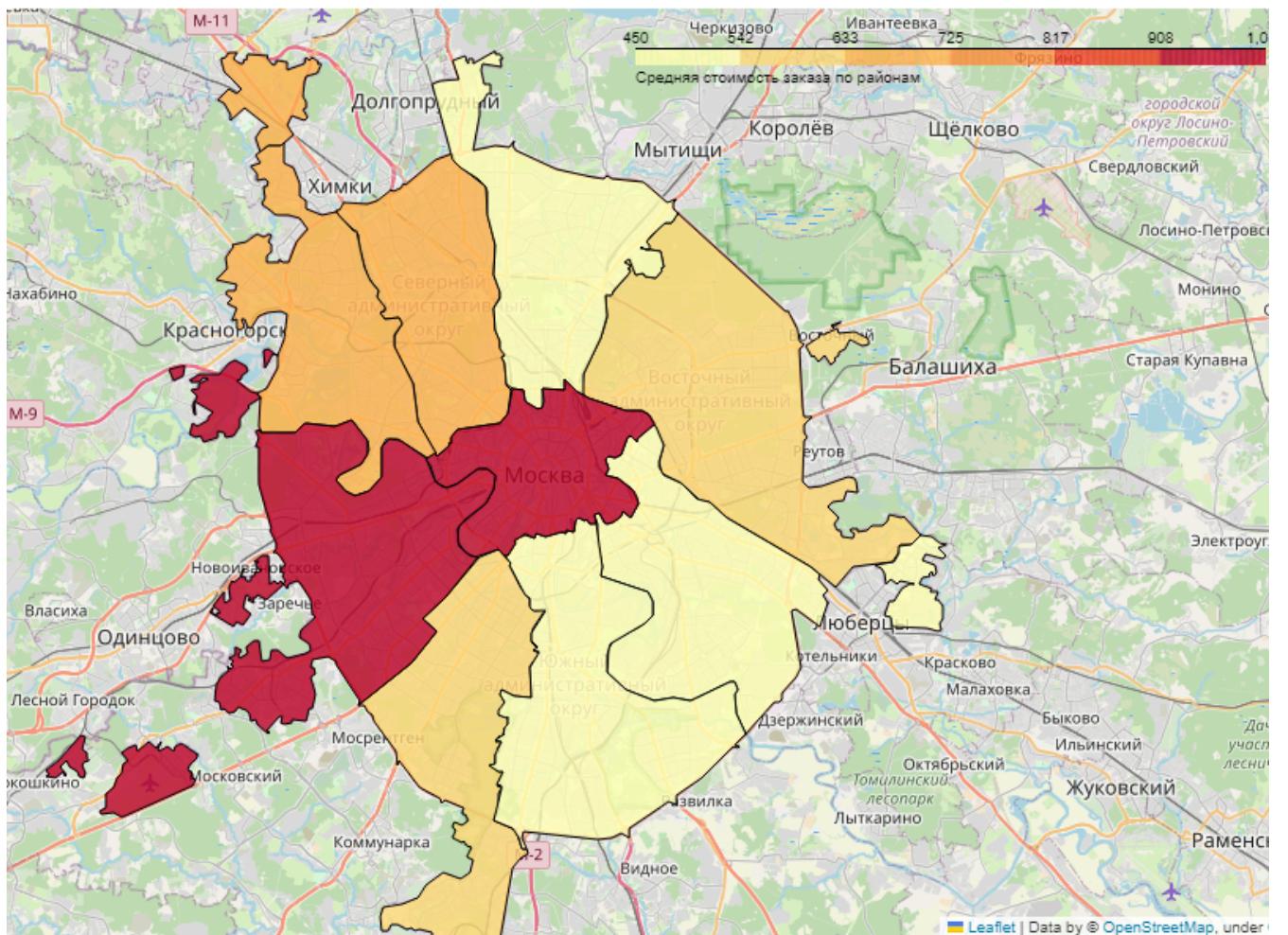
- На топ-15 улиц Москвы - преимущественно стоят кафе, рестораны и кофейни

Типы заведений на топ-15 улиц Москвы



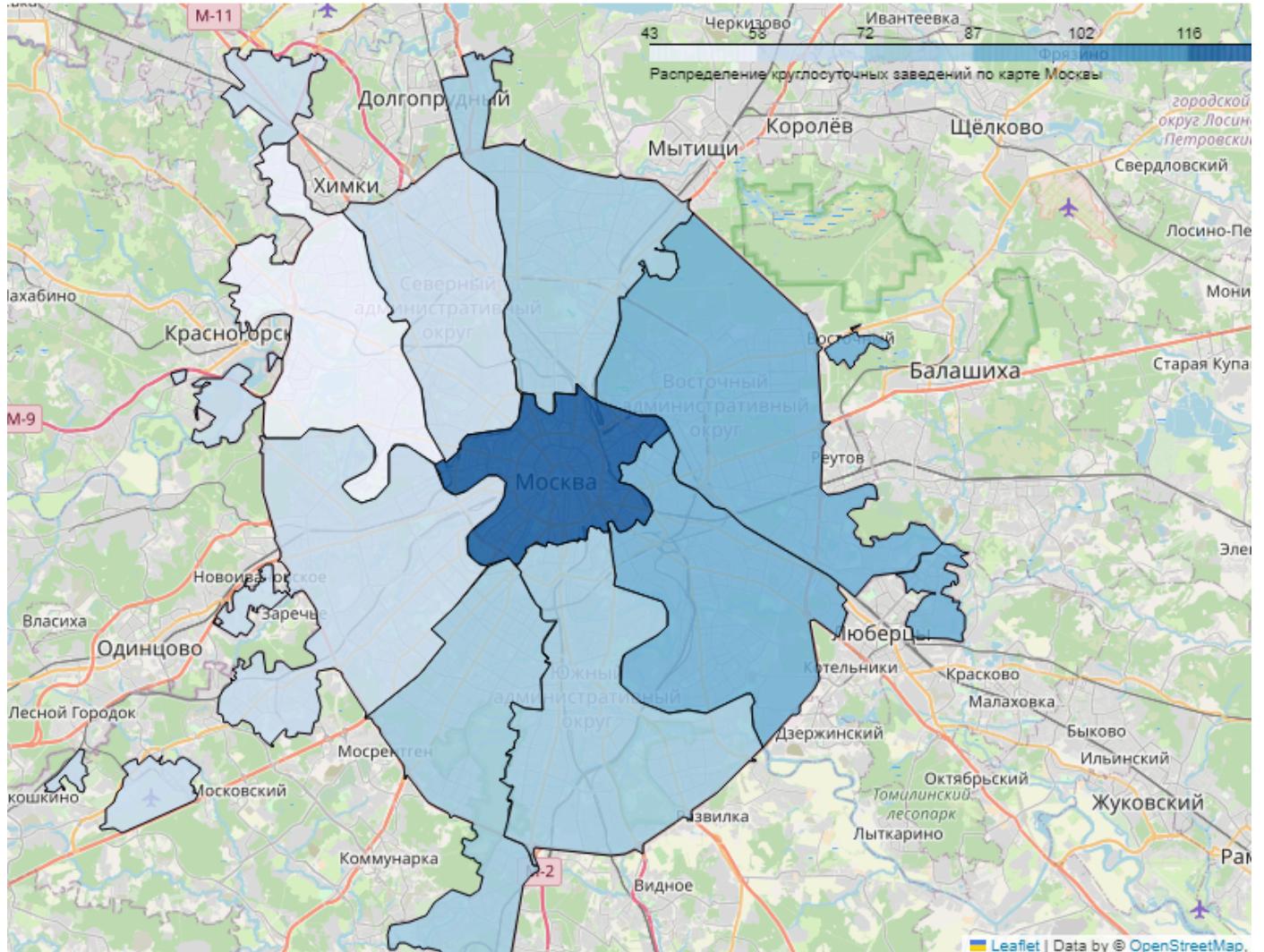
- В Москве есть 457 улиц, на которых расположено только одно заведение Общепита - чаще всего это кафе, рестораны, кофейни

- Самый дорогой средний чек в районе ЗАО и ЦАО

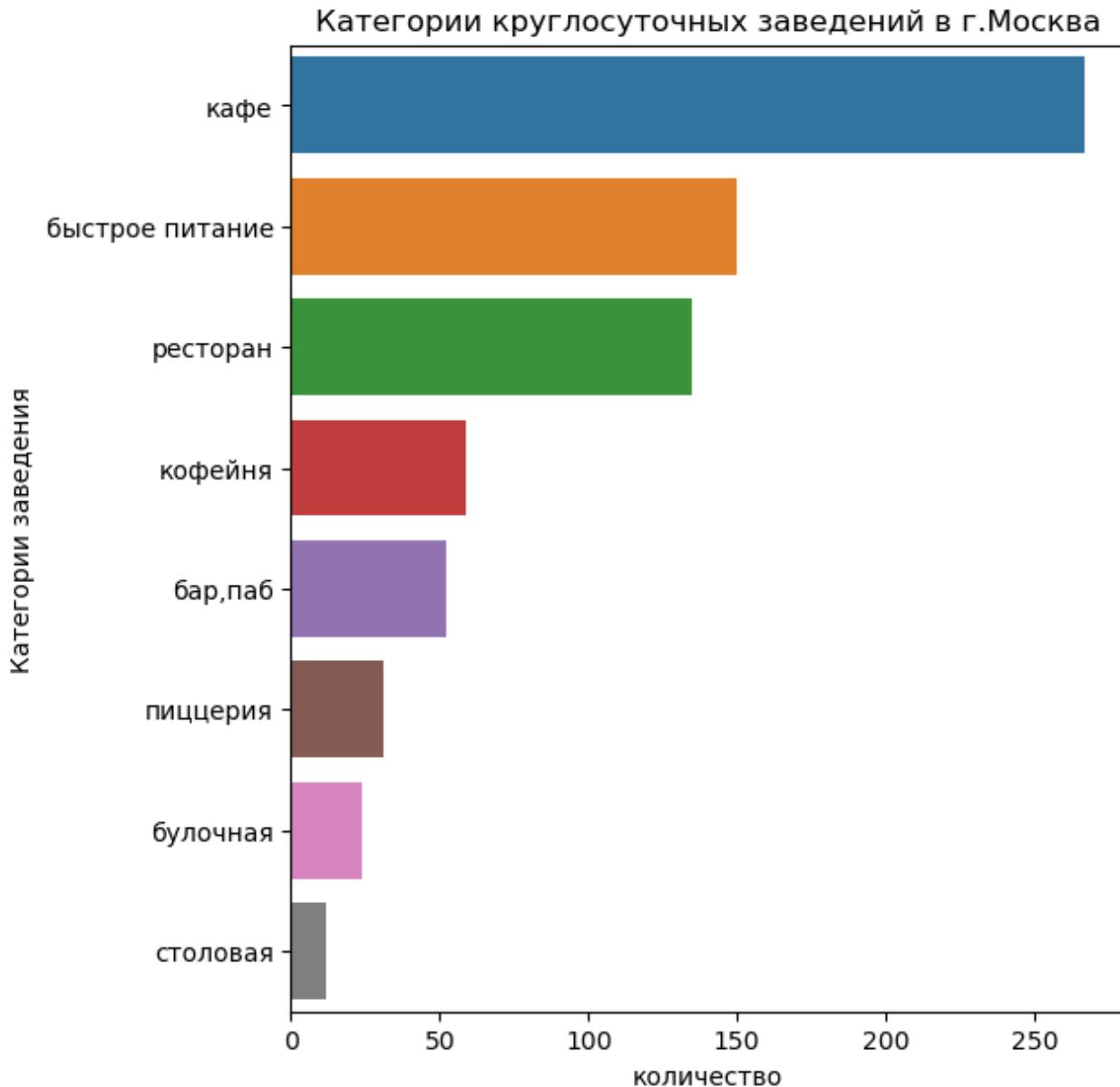


- **большинство круглосуточных заведений находится в центре Москвы (район - ЦАО, сосредоточение туристов).**

На 2м и 3м месте по количеству заведений с графиком работы 24/7 находятся - Восточный административный округ и Юго-Восточный административный округ

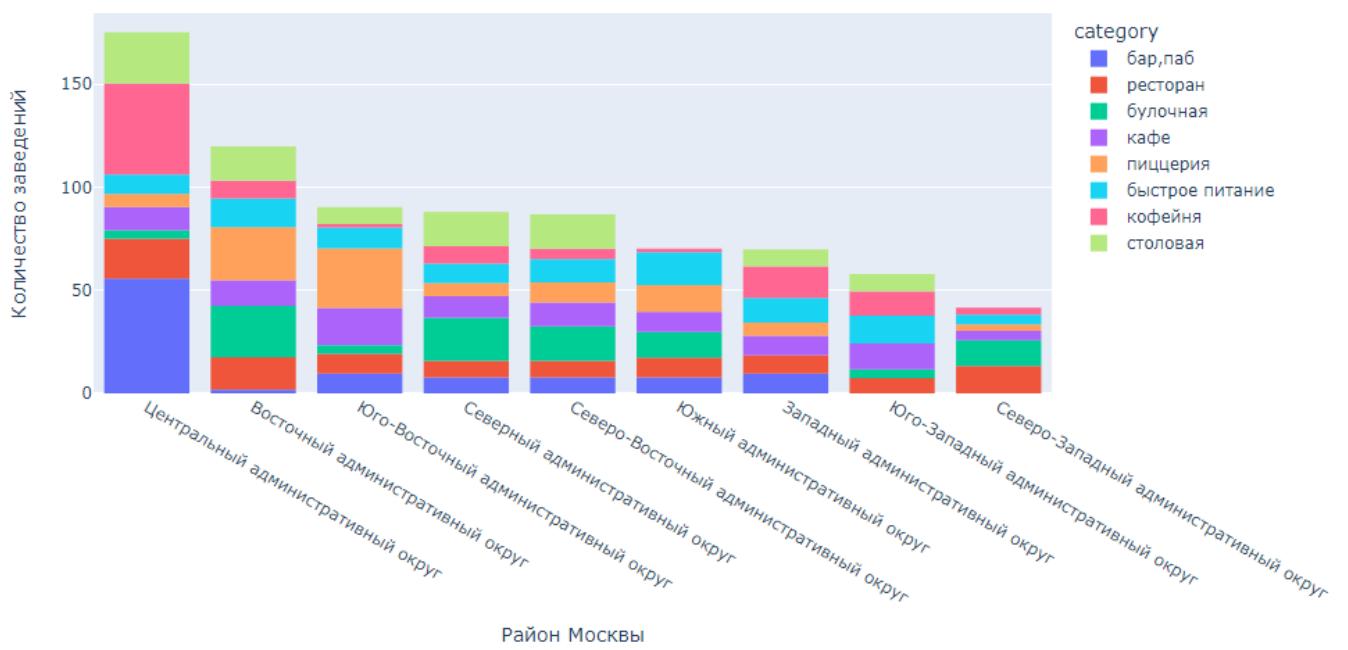


Чаще всего это кафе, фаст-фуд и рестораны



В центре (район ЦАО) - больше круглосуточных баров/пабов и ресторанов

Круглосуточные заведения в Москве - по районам и типам



# Шаг 4. Детализируем исследование: открытие кофейни

Основателям фонда «Shut Up and Take My Money» не даёт покоя успех сериала «Друзья». Их мечта — открыть такую же крутую и доступную, как «Central Perk», кофейню в Москве. Будем считать, что заказчики не боятся конкуренции в этой сфере, ведь кофеен в больших городах уже достаточно. Попробуйте определить, осуществима ли мечта клиентов.

## Посчитаем количество кофеен в датасете

```
In [69]: coffee_house = df[df['category']=='кофейня']
coffee_house['name'].count()
```

```
Out[69]: 1413
```

В датасете 1 413 кофеен

Посчитаем уникальные названия кофеен (т.к. в датасете много сетевых заведений)

```
In [70]: coffee_house['name'].nunique()
```

```
Out[70]: 826
```

В датасете 839 кофеен с уникальными названиями

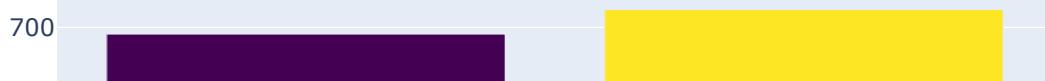
Посмотрим процент сетевых/несетевых кофеен

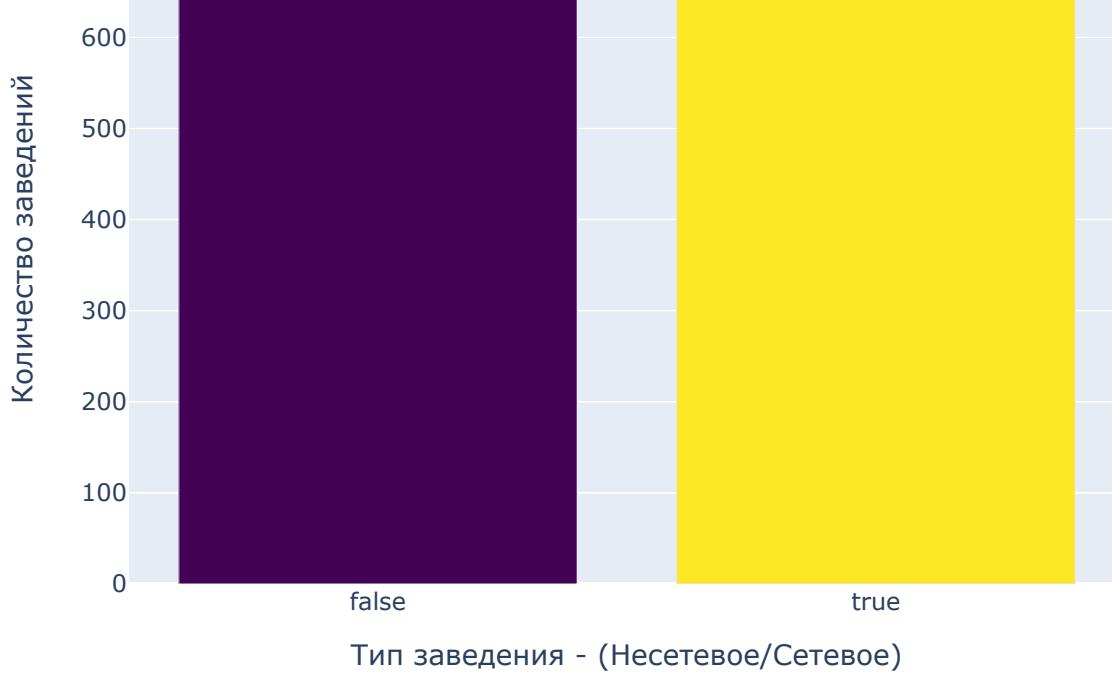
```
In [71]: net_stat = coffee_house.groupby('chain', as_index=False) ['name'].count()
net_stat.columns = ('сетевое заведение', 'количество')
net_stat
```

	сетевое заведение	количество
0	False	693
1	True	720

```
In [72]: fig = go.Figure(go.Bar(
    x=net_stat['сетевое заведение'],
    y=net_stat['количество'],
    orientation='v',
    marker=dict(color = [1,2],
                colorscale='viridis')))\n.update_layout(
    xaxis_title="Тип заведения - (Несетевое/Сетевое)", yaxis_title="Количество заведений",
    title="Соотношение несетевых и сетевых заведений", title_x=0.5
)
fig.show()
```

Соотношение несетевых и сетевых заведений





Количество сетевых и несетевых кофеен в нашем датасете примерно равное.

## Посмотрим на топ-5 популярных сетевых кофеен

```
In [73]: coffee_house[coffee_house['chain']==True].groupby('name', as_index=False)[['address']].count().sort_values('address', ascending=False).reset_index(drop=True).head(5)
```

	name	address
0	шоколадница	119
1	one price coffee	71
2	cofix	65
3	кофепорт	42
4	cofefest	31

Топ-5 популярных сетевых кофеен -

1. Шоколадница
2. One Price Coffee
3. Cofix
4. КОФЕПОРТ
5. CofeFest

## Узнаем, в каких районах больше всего кофеен

```
In [74]: coffee_district = coffee_house.groupby('district', as_index=False)[['name']].count()\n.sort_values('name', ascending=False).reset_index(drop=True)\ncoffee_district
```

	district	name
0	Центральный административный округ	428

	district	name
1	Северный административный округ	193
2	Северо-Восточный административный округ	159
3	Западный административный округ	150
4	Южный административный округ	131
5	Восточный административный округ	105
6	Юго-Западный административный округ	96
7	Юго-Восточный административный округ	89
8	Северо-Западный административный округ	62

Визуализируем кофейни на карте

In [75]:

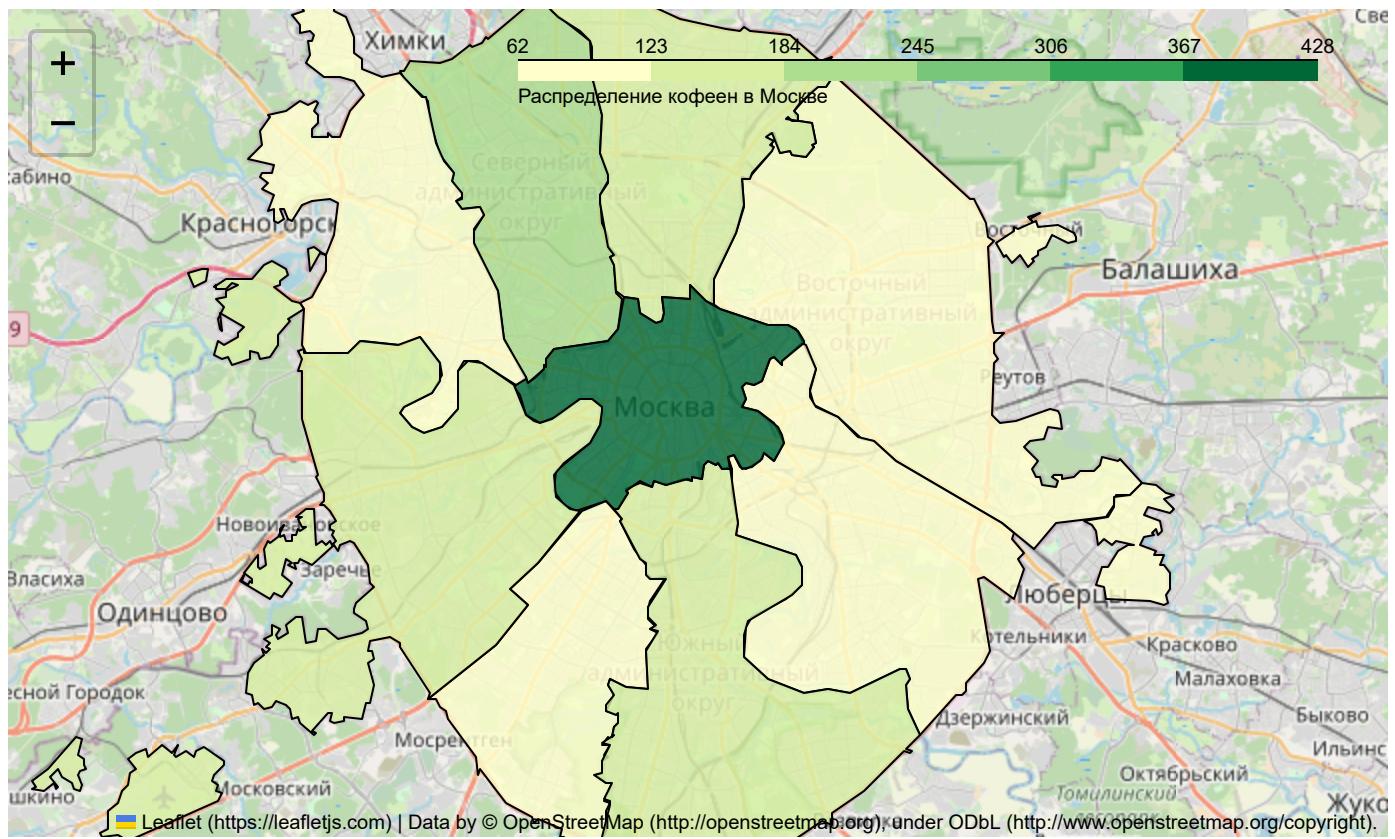
```
# moscow_lat - широта центра Москвы, moscow_lng - долгота центра Москвы
moscow_lat, moscow_lng = 55.751244, 37.618423

# создаём карту Москвы
m = Map(location=[moscow_lat, moscow_lng], zoom_start=10)

# создаём хороплет с помощью конструктора Choropleth и добавляем его на карту
Choropleth(
    geo_data=geo_json,
    data=coffee_district,
    columns=['district', 'name'],
    key_on='feature.name',
    fill_color='YlGn',
    fill_opacity=0.8,
    legend_name='Распределение кофеен в Москве',
).add_to(m)

# выводим карту
m
```

Out[75]:



Судя по карте - по количеству кофеен на районе лидирует ЦАО

### ТОП 3 районов по количеству кофеен -

- 1 Центральный административный округ 428 шт.
- 2 Северный административный округ 193 шт.
- 3 Северо-Восточный административный округ 159 шт.

### Топ 3 по минимальному количеству кофеен

- 1 Северо-Западный административный округ 62 шт.
- 2 Юго-Восточный административный округ 89 шт.
- 3 Юго-Западный административный округ 96 шт.

## Узнаем долю круглосуточных кофеен

In [76]:

```
coffee_24 = coffee_house.groupby('is_24/7', as_index=False) ['name'].count() \
    .sort_values('name', ascending=False).reset_index(drop=True)
coffee_24 = coffee_24.rename(columns={'name': 'count'})
coffee_24['%'] = round(coffee_24['count']*100/coffee_24['count'].sum(), 2)
coffee_24
```

Out[76]:

	is_24/7	count	%
0	False	1339	95.78
1	True	59	4.22

96 % (1353) кофейни \*\*не являются\*\* круглосуточными.

В Москве всего 59 кофеен - 4% от Общего числа - которые являются круглосуточными

## Какие у кофеен рейтинги? Как они распределяются по районам?

Посчитаем **медианный рейтинг** кафеен по каждому из районов Москвы -

In [77]:

```
coffee_house.groupby('district', as_index=False) ['rating'].median() \
    .sort_values('rating', ascending = False).reset_index(drop=True)
```

Out[77]:

	district	rating
0	Восточный административный округ	4.3
1	Северный административный округ	4.3
2	Северо-Восточный административный округ	4.3
3	Северо-Западный административный округ	4.3
4	Центральный административный округ	4.3
5	Юго-Восточный административный округ	4.3
6	Юго-Западный административный округ	4.3
7	Южный административный округ	4.3
8	Западный административный округ	4.2

Медианные рейтинг кофеен во всех районах примерно равный - 4.3.

В ЗАО (Западный административный округ) самый минимальный медианный рейтинг кофеен - 4.2 - но отличие в 0.1 несущественно.

Построим график BOXPLOT

In [78]:

```
ax = sns.boxplot(y='district', x='rating', data=coffee_house)
plt.title('Распределение рейтингов кофеен по районам Москвы')
ax.set(ylabel='Районы Москвы', xlabel='Рейтинг')
plt.show()
```

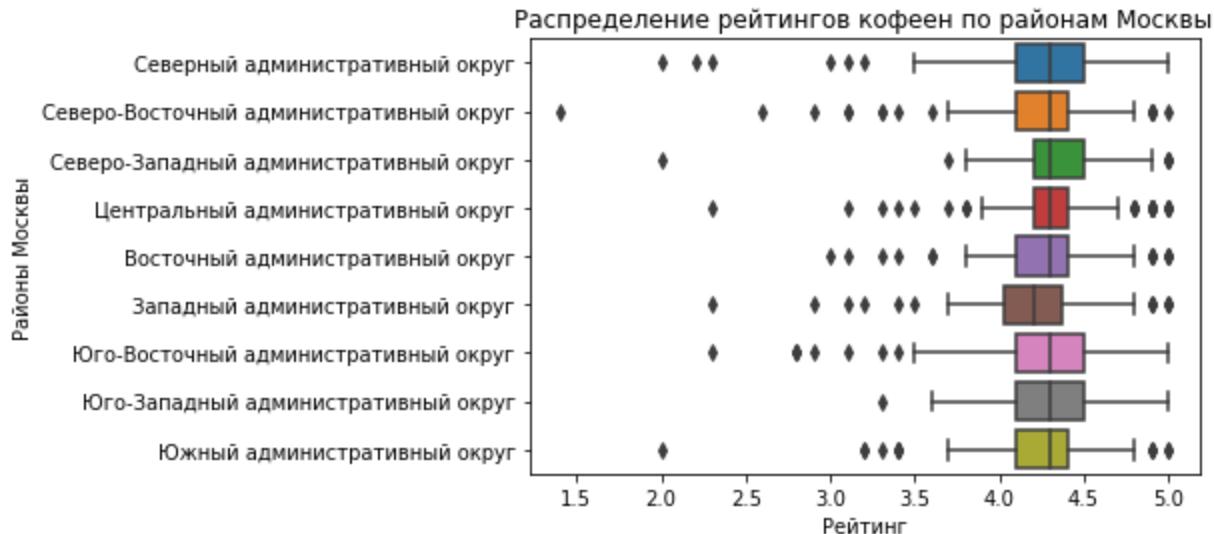


График BoxPlot показывает - что более 50% кофеен имеют рейтинг от 4.0 до 4.5 - что говорит нам о том, что кофейни, в целом, нравятся жителям города

## На какую стоимость чашки капучино стоит ориентироваться при открытии и почему?

Посмотрим на медианное значение стоимости чашки кофе по всем кофейням Москвы (где есть информация в нашем датасете)

In [79]:

```
coffee_house['middle_coffee_cup'].median()
```

Out[79]:

170.0

**Медианная цена для чашки кофе по всей Москве - 170 рублей**

Посчитаем медианную цену для каждого из районов и визуализируем -

In [80]:

```
coffee_price = coffee_house.groupby('district', as_index=False)[['middle_coffee_cup']].median()
.sort_values('middle_coffee_cup', ascending=False).reset_index(drop=True)
coffee_price
```

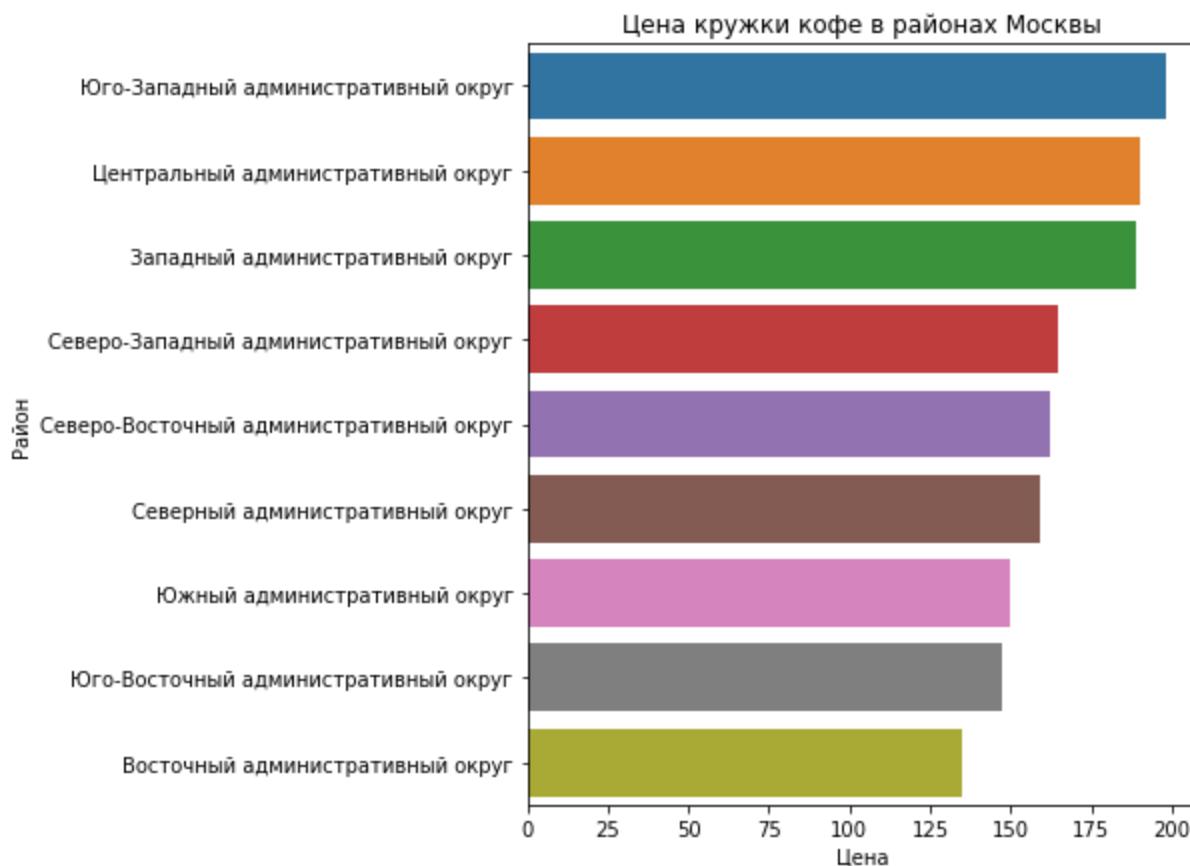
Out[80]:

	district	middle_coffee_cup
0	Юго-Западный административный округ	198.0
1	Центральный административный округ	190.0
2	Западный административный округ	189.0
3	Северо-Западный административный округ	165.0

	district	middle_coffee_cup
4	Северо-Восточный административный округ	162.5
5	Северный административный округ	159.0
6	Южный административный округ	150.0
7	Юго-Восточный административный округ	147.5
8	Восточный административный округ	135.0

In [81]:

```
plt.figure(figsize=(6, 7))
ax = sns.barplot(y="district", x="middle_coffee_cup", data=coffee_price);
sns.color_palette('bright')
plt.title('Цена кружки кофе в районах Москвы')
ax.set(ylabel='Район', xlabel='Цена')
plt.show()
```



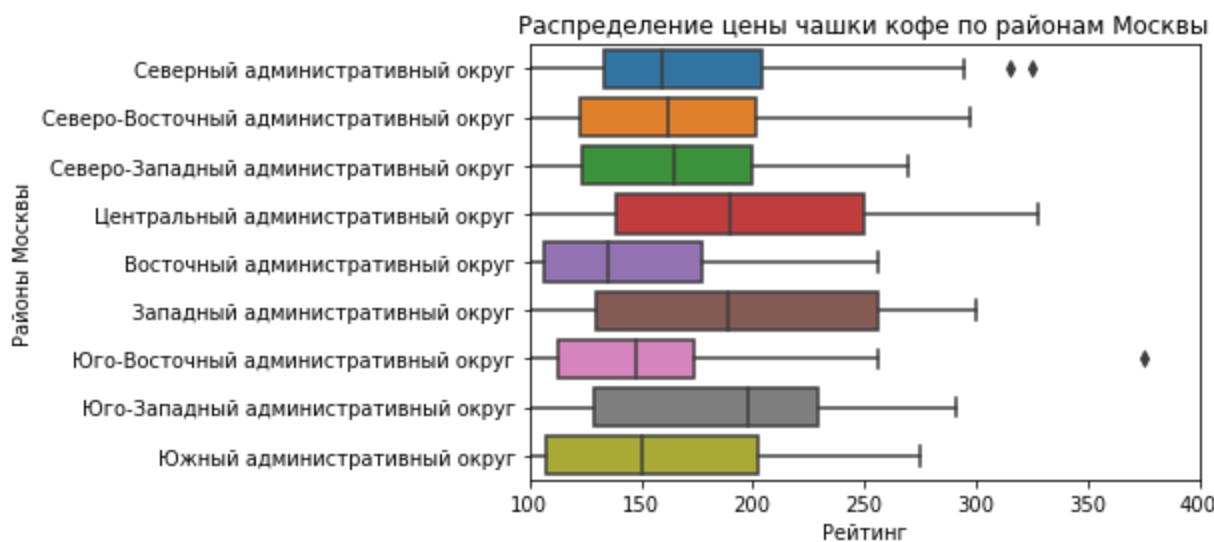
На удивление - **самая большая цена за чашку кофе в Юго-западном административном округе**, даже больше чем в ЦАО.

Топ 3 по стоимости чашки кофе -

1. ЮЗАО - 198 рублей
2. ЦАО - 190 рублей
3. ЗАО - 189 рублей

In [82]:

```
ax = sns.boxplot(y='district', x='middle_coffee_cup', data=coffee_house)
plt.title('Распределение цены чашки кофе по районам Москвы')
ax.set(ylabel='Районы Москвы', xlabel='Рейтинг', xlim = (100, 400))
plt.show()
```



Как мы видим, цена за кружку кофе варьируется от 100 до 250 рублей, в зависимости от района.

## Посчитаем медианное число посадочных мест для несетевых кофейен

In [83]:

```
coffee_seats = coffee_house[coffee_house['chain']==False].groupby('district', as_index=False)
.sort_values('seats', ascending = False).reset_index(drop=True)
coffee_seats
```

Out[83]:

	district	seats
0	Западный административный округ	98.0
1	Южный административный округ	80.0
2	Северный административный округ	70.0
3	Юго-Восточный административный округ	54.0
4	Восточный административный округ	52.0
5	Северо-Восточный административный округ	52.0
6	Центральный административный округ	50.0
7	Юго-Западный административный округ	50.0
8	Северо-Западный административный округ	25.5

Больше всего посадочных мест в кофейнях (нестетевых) в следующих районах Москвы -

1. ЗАО - 98 мест
2. ЮАО - 80 мест
3. САО - 70 мест

Посчитаем медианное значение по всем несетевым кофейням -

In [84]:

```
coffee_house[coffee_house['chain']==False]['seats'].median()
```

Out[84]:

60.0

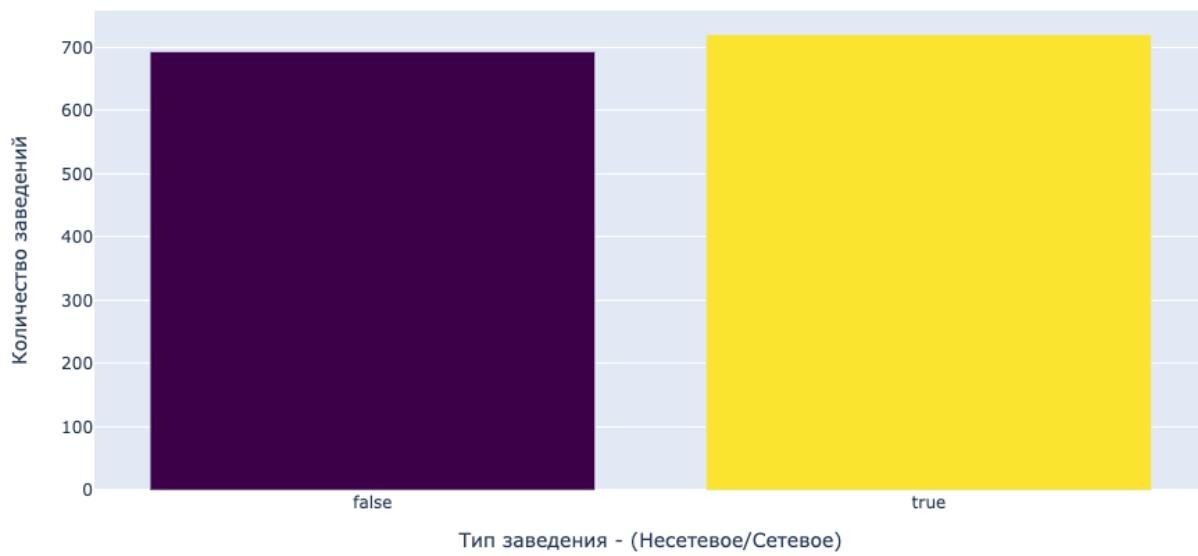
**Медианное значение по всем нетесетевым кофейням Москвы - 60 посадочных мест**

**Предложение инвесторам фонда по открытию кофейни (вывод по исследованию кофеен)**

На рынке Москвы, судя по предоставленным данным, представлено 1 413 кофеен.

Из них сетевых заведений - 720 шт, оставшиеся кофейни (693 шт.) - несетевые.

#### Соотношение несетевых и сетевых заведений



#### Топ-5 популярных сетевых кофеен -

1. Шоколадница
2. One Price Coffee
3. Cofix
4. КОФЕПОРТ
5. CofeFest

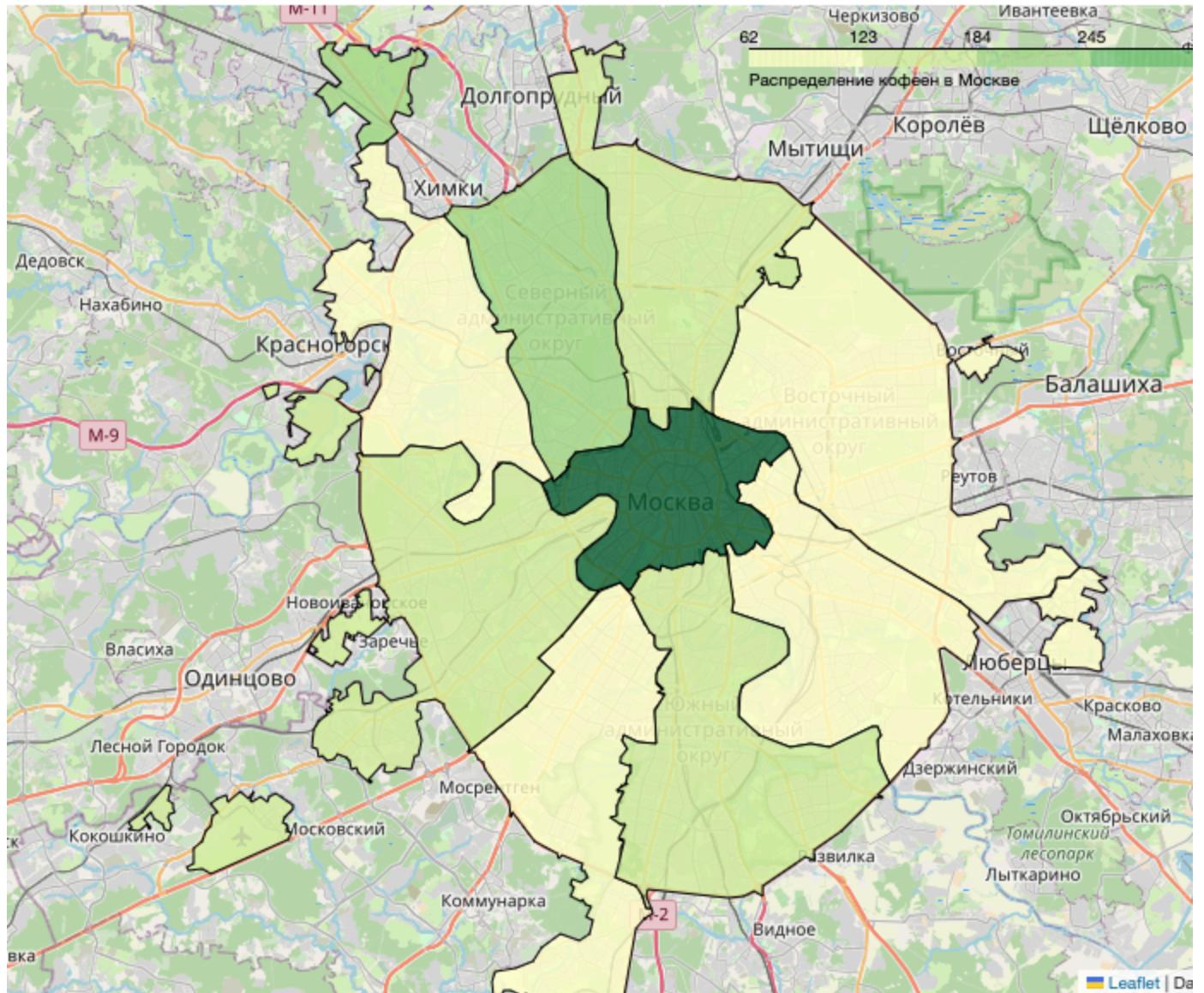
Т.к. мечта инвесторов фонда открыть уникальное заведение - **будем рассматривать открытие уникального кафе - несетевого.**

#### ТОП 3 районов по количеству кофеен -

- 1 Центральный административный округ 428 шт.
- 2 Северный административный округ 193 шт.
- 3 Северо-Восточный административный округ 159 шт.

#### Топ 3 по минимальному количеству кофеен

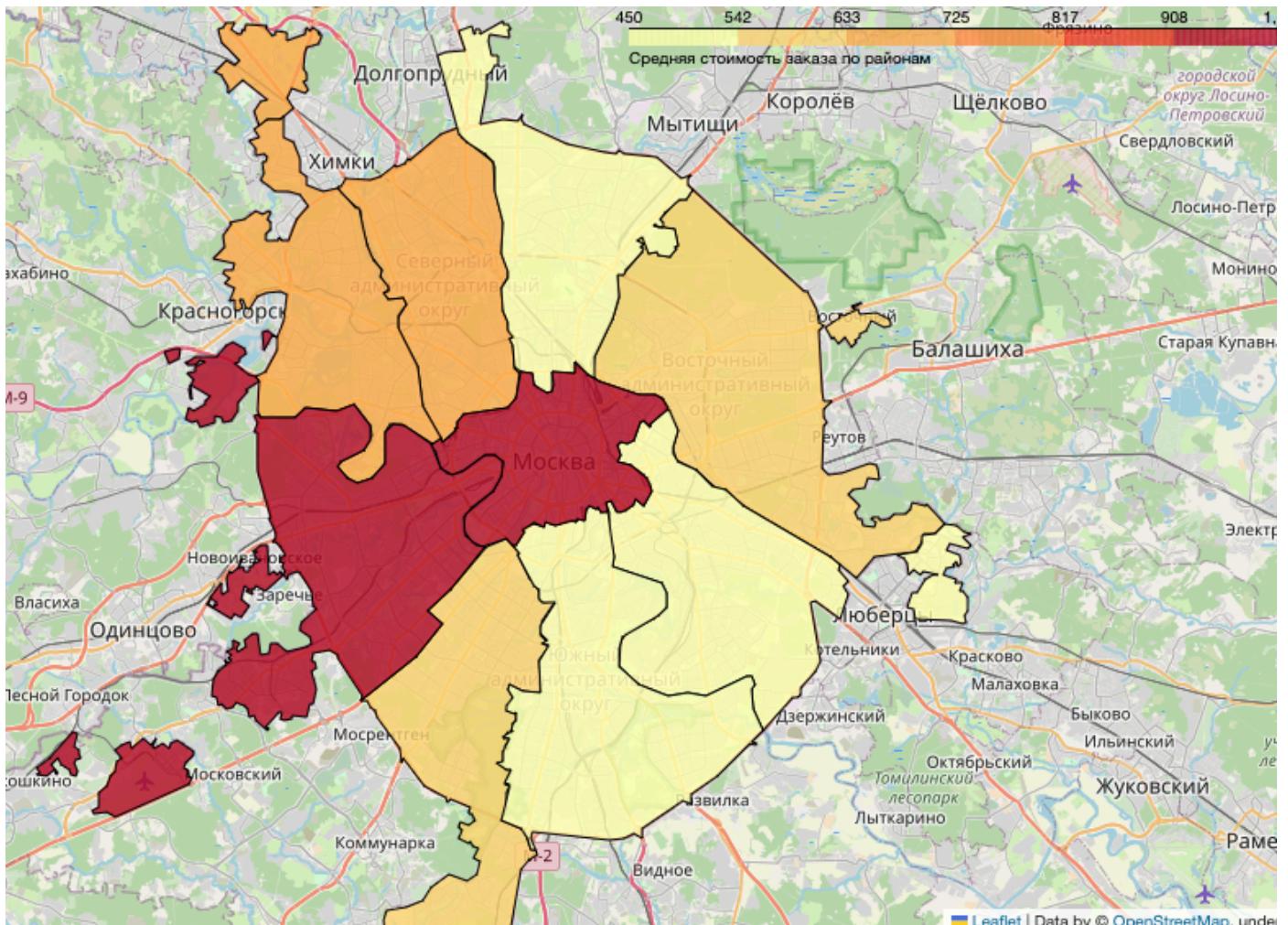
- 1 Северо-Западный административный округ 62 шт.
- 2 Юго-Восточный административный округ 89 шт.
- 3 Юго-Западный административный округ 96 шт.



Предлагаю рассмотреть инвесторам для открытия своей кофейни - Юго-Западный район Москвы - здесь меньше конкуренция (занимает 7 из 9 мест по количеству кофеен), а так же согласно нашему исследованию - Данный район входит в ТОП-5 районов по цене среднего чека заказа - то есть доля платежеспособного населения - велика.

	district	middle_avg_bill
--	----------	-----------------

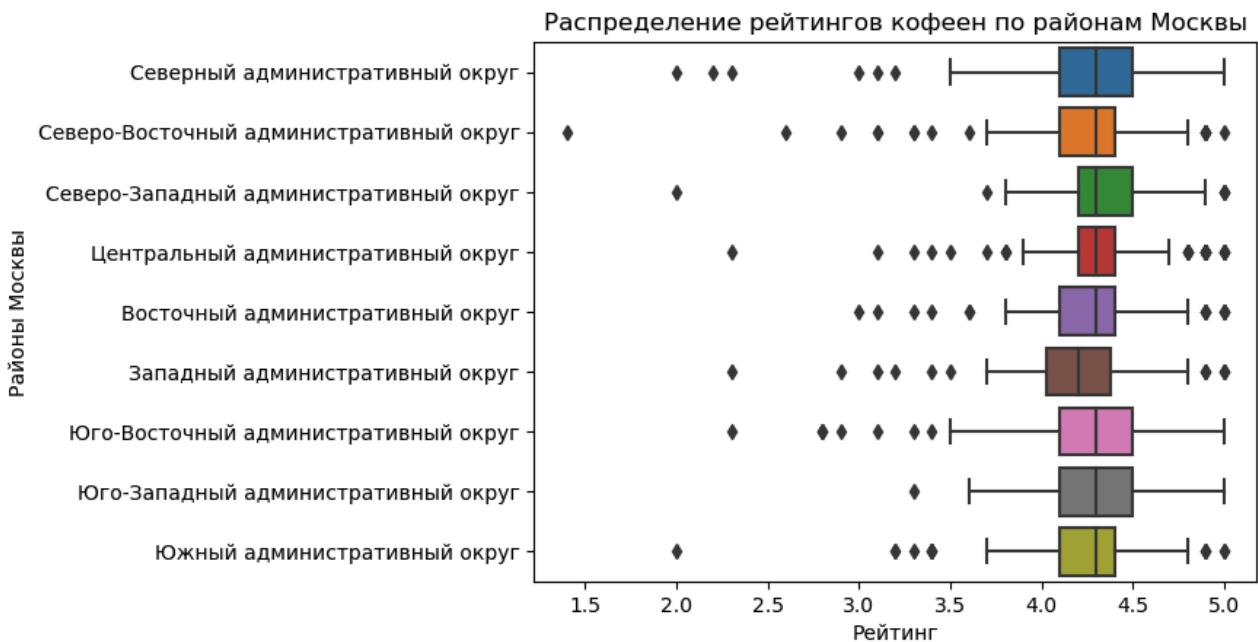
0	Западный административный округ	1000.0
1	Центральный административный округ	1000.0
2	Северо-Западный административный округ	700.0
3	Северный административный округ	650.0
4	Юго-Западный административный округ	600.0



Кофеен, работающих круглосуточно - всего 4%. Для нового кафе - **\*\*предлагаю не использовать\*\*** круглосуточный график работы, для сокращения издержек на зарплату сотрудникам.

В дальнейшем - возможно будет рассмотреть переход на круглосуточный режим работы, в зависимости от потока людей.

Медианный рейтинг кафеен во всех районах равен ~ 4.3 - то есть, в целом, заведения данной категории популярны у местных жителей.



Медианная цена чашки кофе в Москве - 170 рублей.

Для инвесторов **\*\*я рекомендую ориентироваться на цену кружки кофе (170-180 рублей)\*\***, т.к. медианная цена в районе ЮЗАО - 198 рублей, что даже выше чем в ЦАО (190 рублей).

Данная цена (~ 170-180 рублей) будет конкурентным преимуществом кофейни, а также удовлетворит условию инвесторов открыть **доступную** кофейню.

*Изначально в данных в поле со средней ценой чашки кофе было около 93,6% пропусков - поэтому для определению цены для нашего кафе - предлагаю дополнительно изучить этот вопрос по району ЮЗАО.*

**Медианное значение по количеству посадочных мест по всем нетесетевым кофейням Москвы - 60 шт.**

Судя по нашему датасету - медианное значение посадочных мест по району ЮЗАО - 50 шт. Оно занимает

		district	seats
0	Западный административный округ	98.0	
1	Южный административный округ	80.0	
2	Северный административный округ	70.0	
3	Юго-Восточный административный округ	54.0	
4	Восточный административный округ	52.0	
5	Северо-Восточный административный округ	52.0	
6	Центральный административный округ	50.0	
7	Юго-Западный административный округ	50.0	
8	Северо-Западный административный округ	25.5	

предпоследнее место в рейтинге.

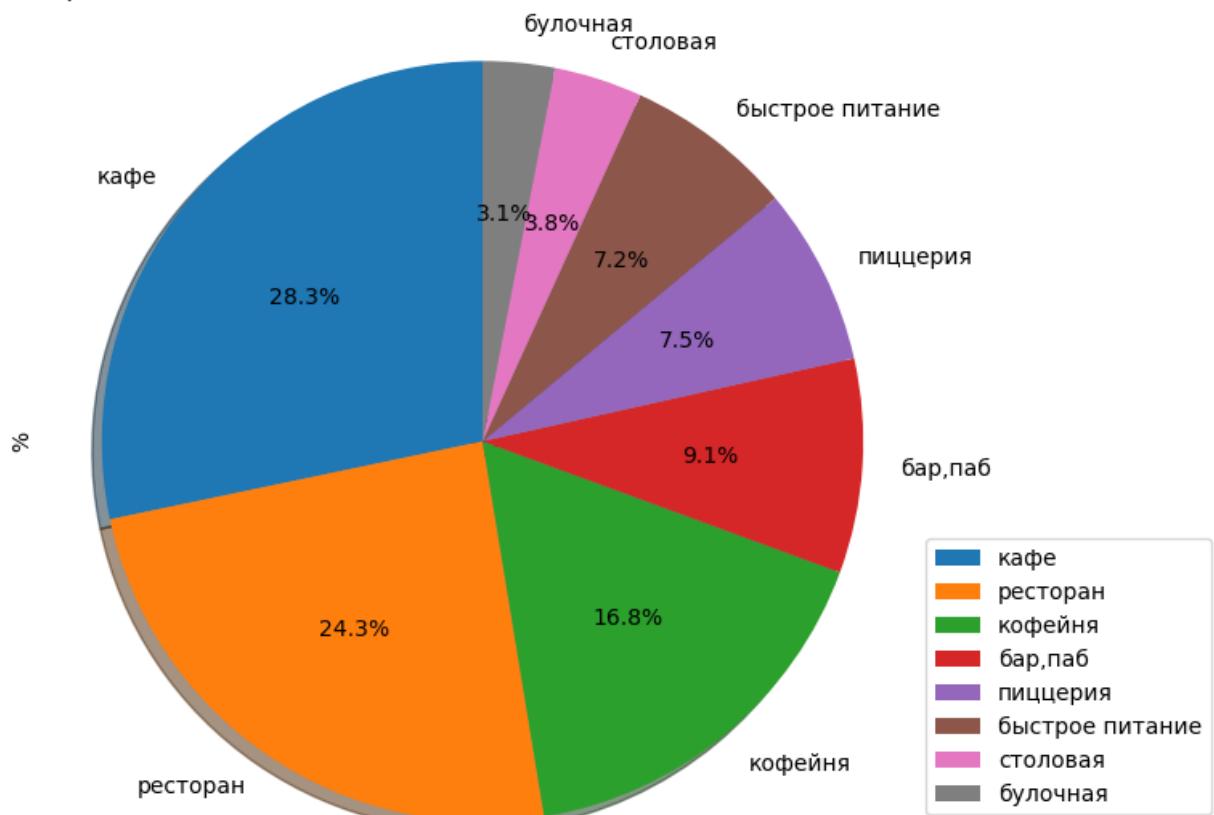
Нашим инвесторам я рекомендую ориентироваться на **\*\*60 посадочных мест\*\*** в заведении

**Вывод по исследованию - Шаг №5**

В результате анализа датасета , с заведениями общественного питания Москвы, составленного на основе данных сервисов Яндекс Карты и Яндекс Бизнес на лето 2022 года, мы подготовили исследование рынка Москвы, нашли интересные особенности, подготовили рекомендации инвесторам, которые в будущем помогут в выборе подходящего места для кофейни.

- Категории Заведений Общепита в Москве

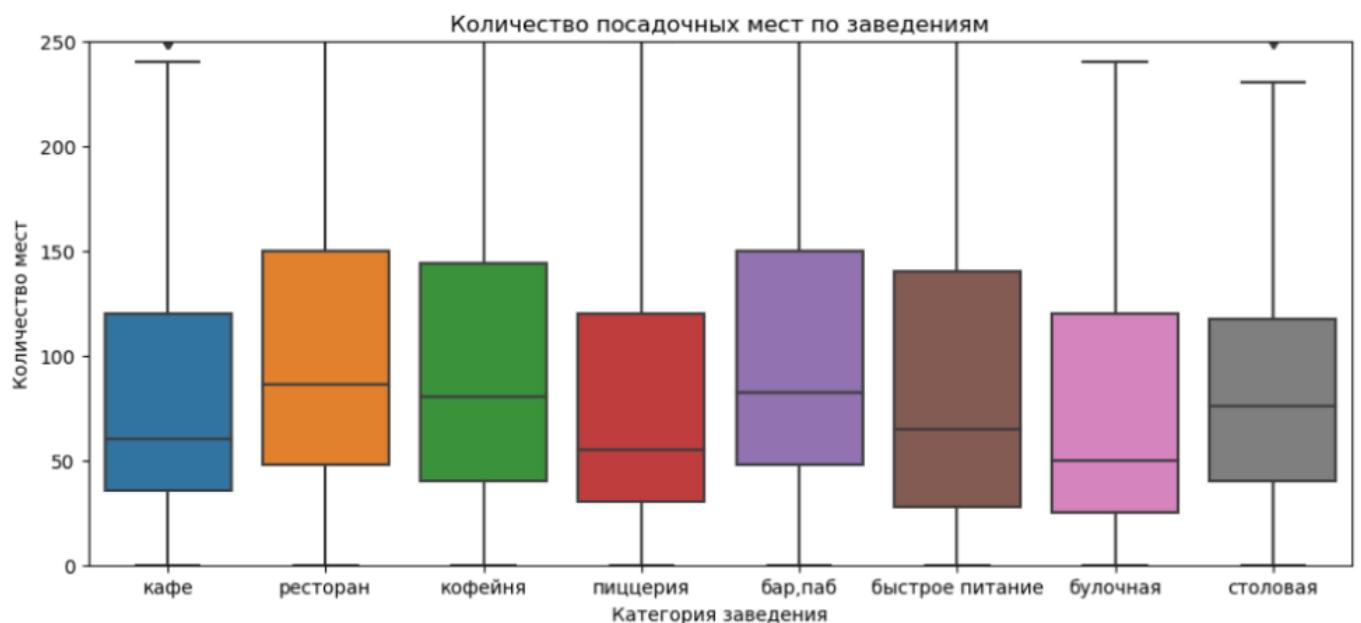
**Распределение заведений в Москве - относительное количество, %**



Больше всего в Москве - кафе и ресторанов - 28.3% и 24.3% соответственно

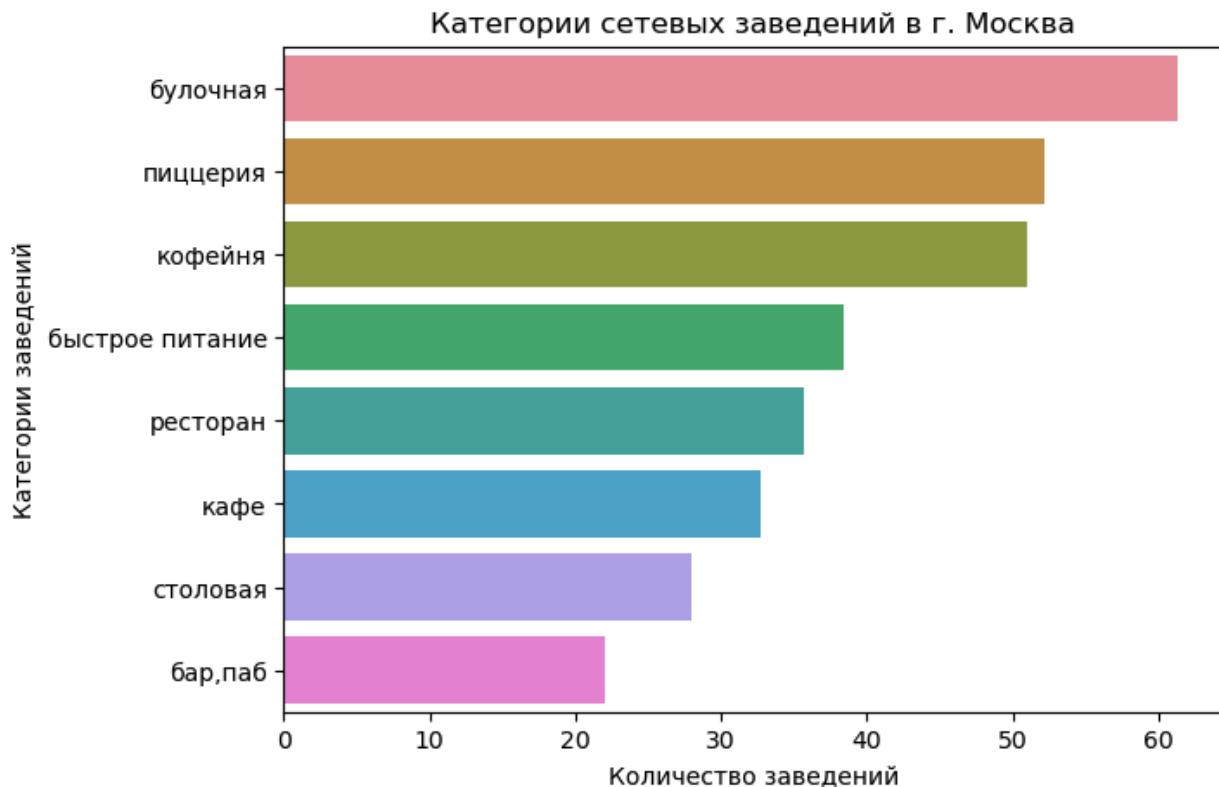
**Медианное значение посадочных мест в больших заведениях (ресторан, бар/паб, кофейня, столовая) - составляет от 76 до 86 посадочных мест**

**В маленьких заведениях (Булочная, пиццерия, кафе, быстрое питание) - количество мест составляет - 55-65**



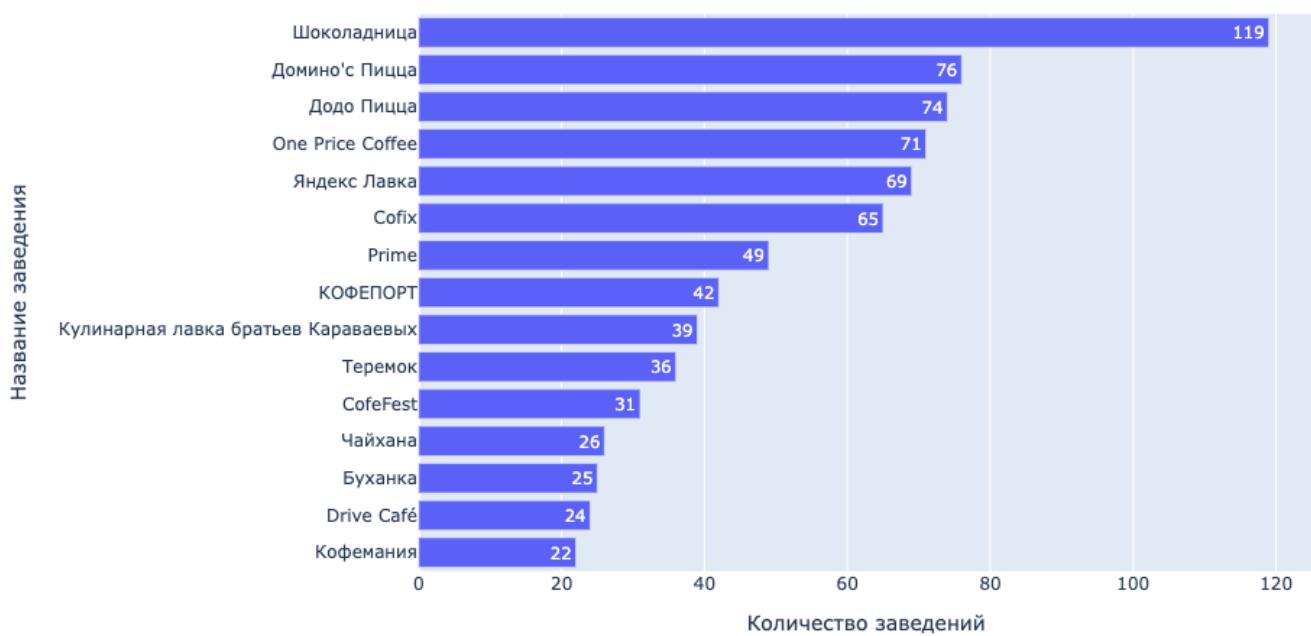
## Самые популярные сетевые заведения Общепита по категориям -

- Булочные
- Пиццерии
- Кофейни



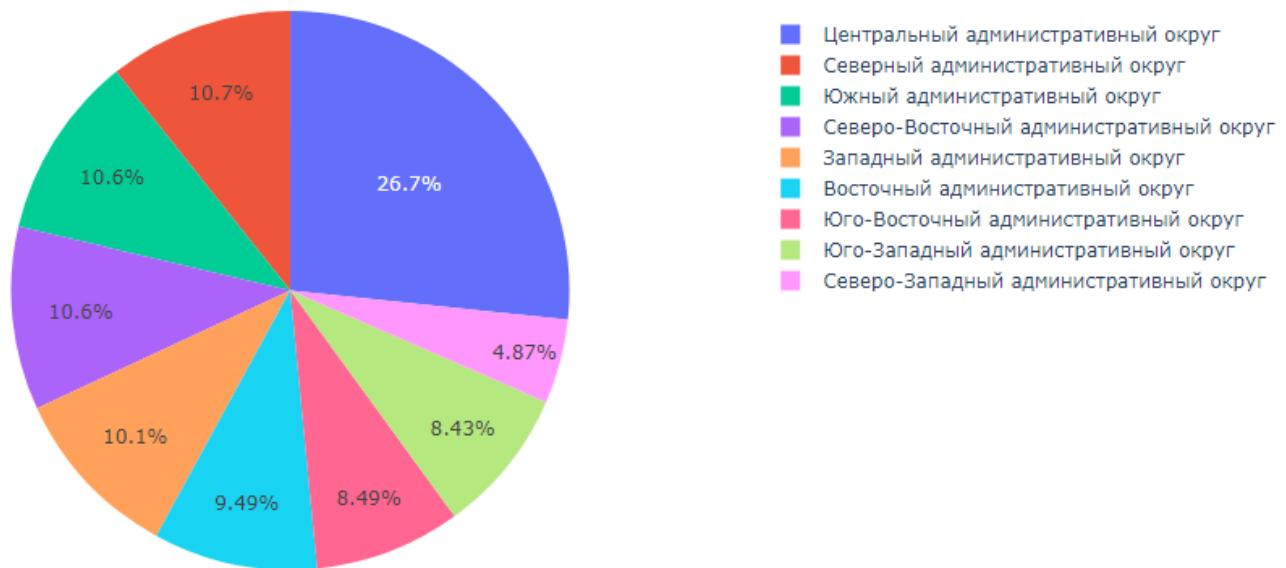
## Топ 15 сетевых заведений -

Топ 15 сетевых заведений в Москве



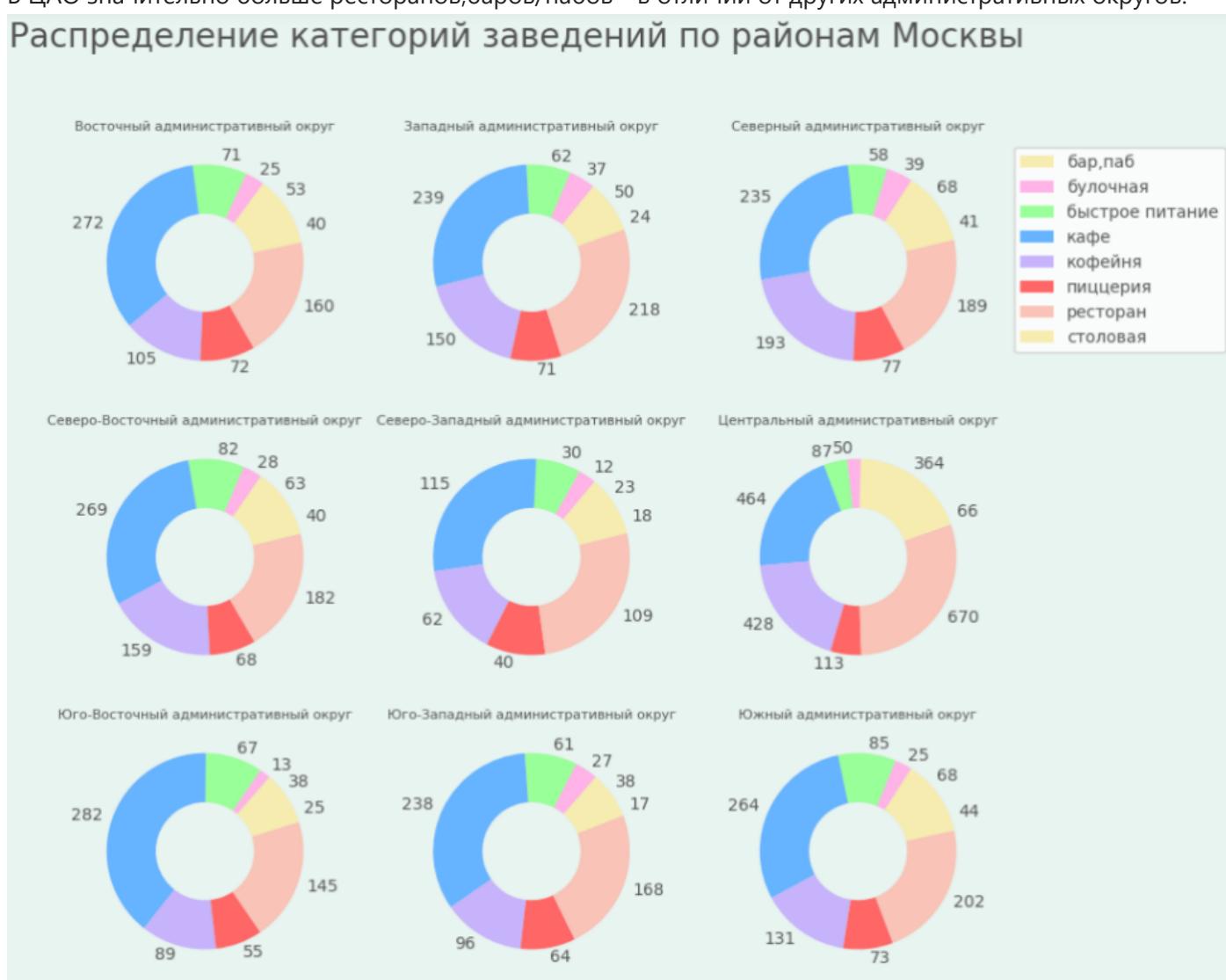
**Большая часть заведений Общепита** находится в центральном административном округе (далее - ЦАО) - , т.к. в ЦАО - основное скопление туристов и платежеспособных людей.

# Разделение заведений по районам Москвы

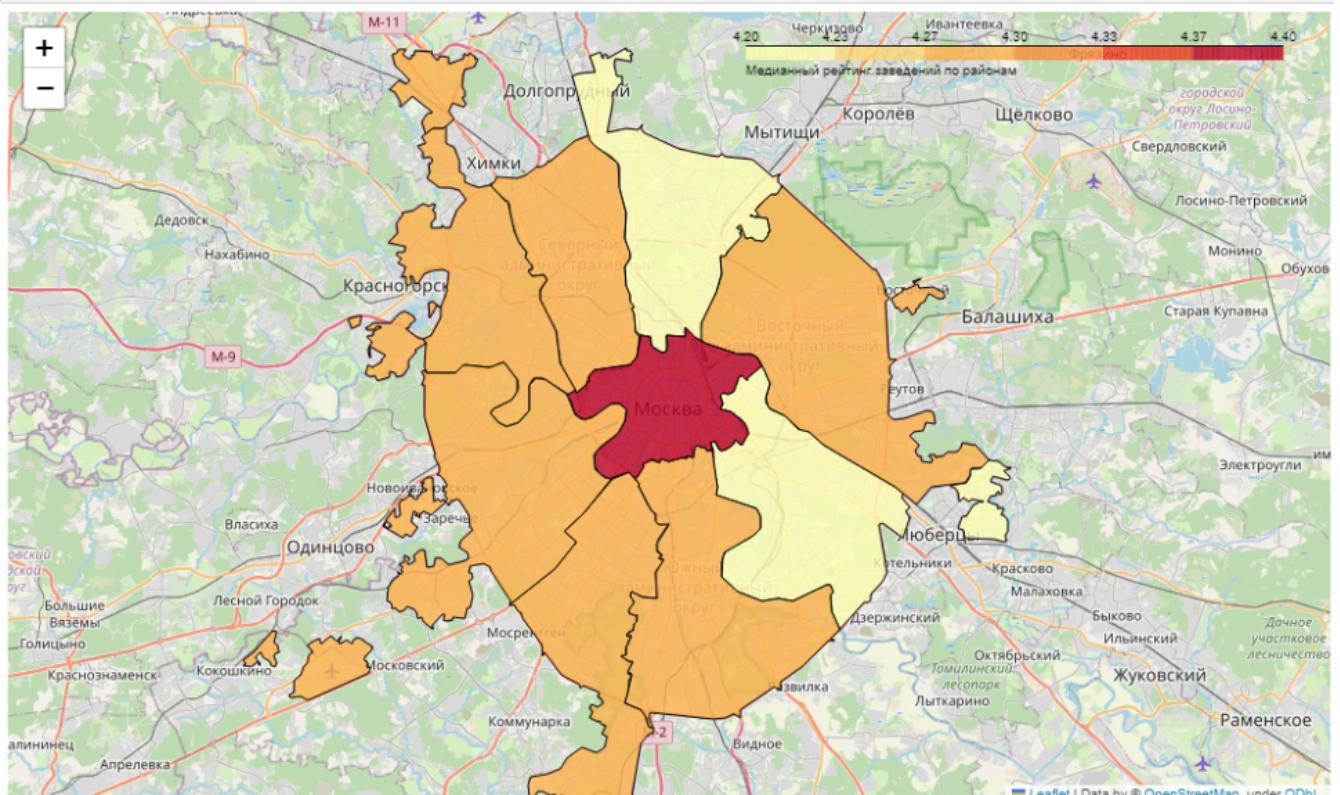


В ЦАО значительно больше ресторанов, баров/пабов - в отличии от других административных округов.

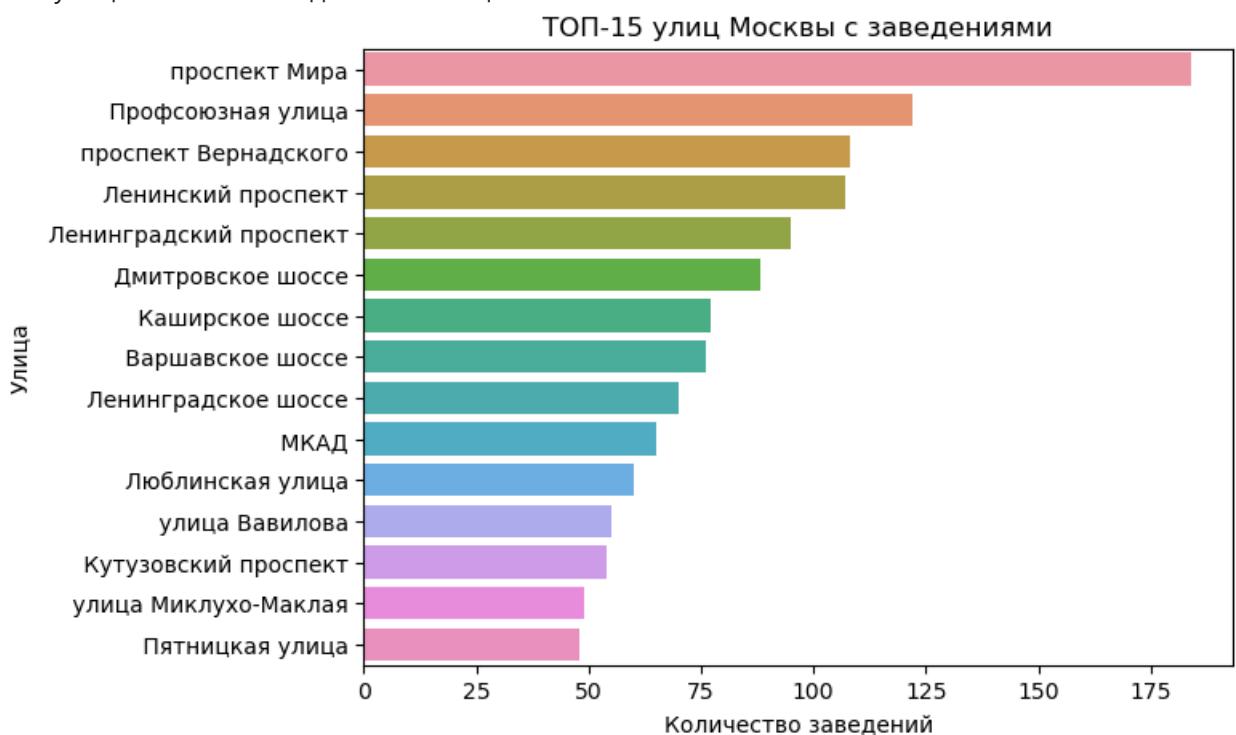
## Распределение категорий заведений по районам Москвы



- Лучшие по рейтингу заведения находятся в центре Москвы (район - ЦАО)

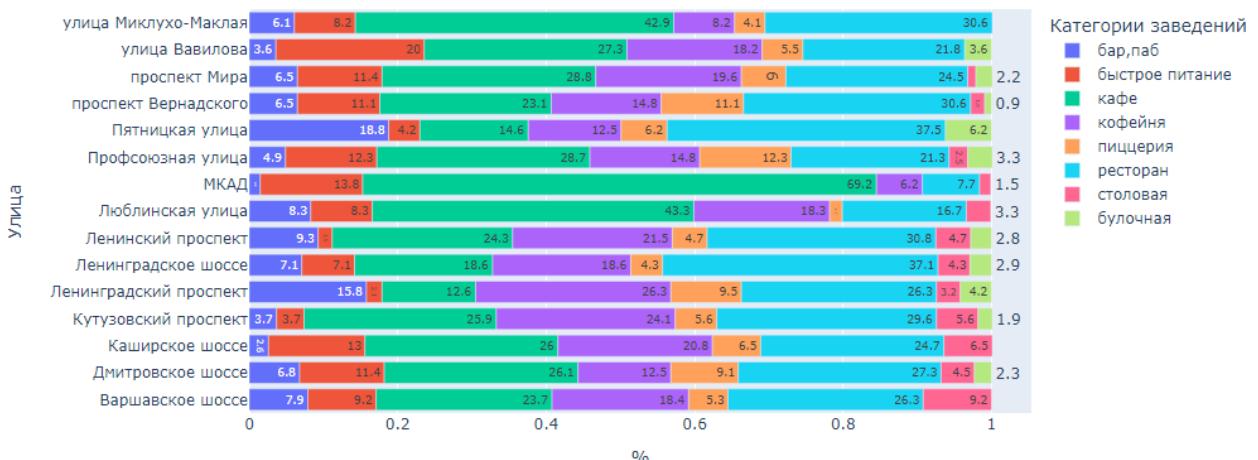


- Топ-15 улиц Москвы с заведениями Общепита

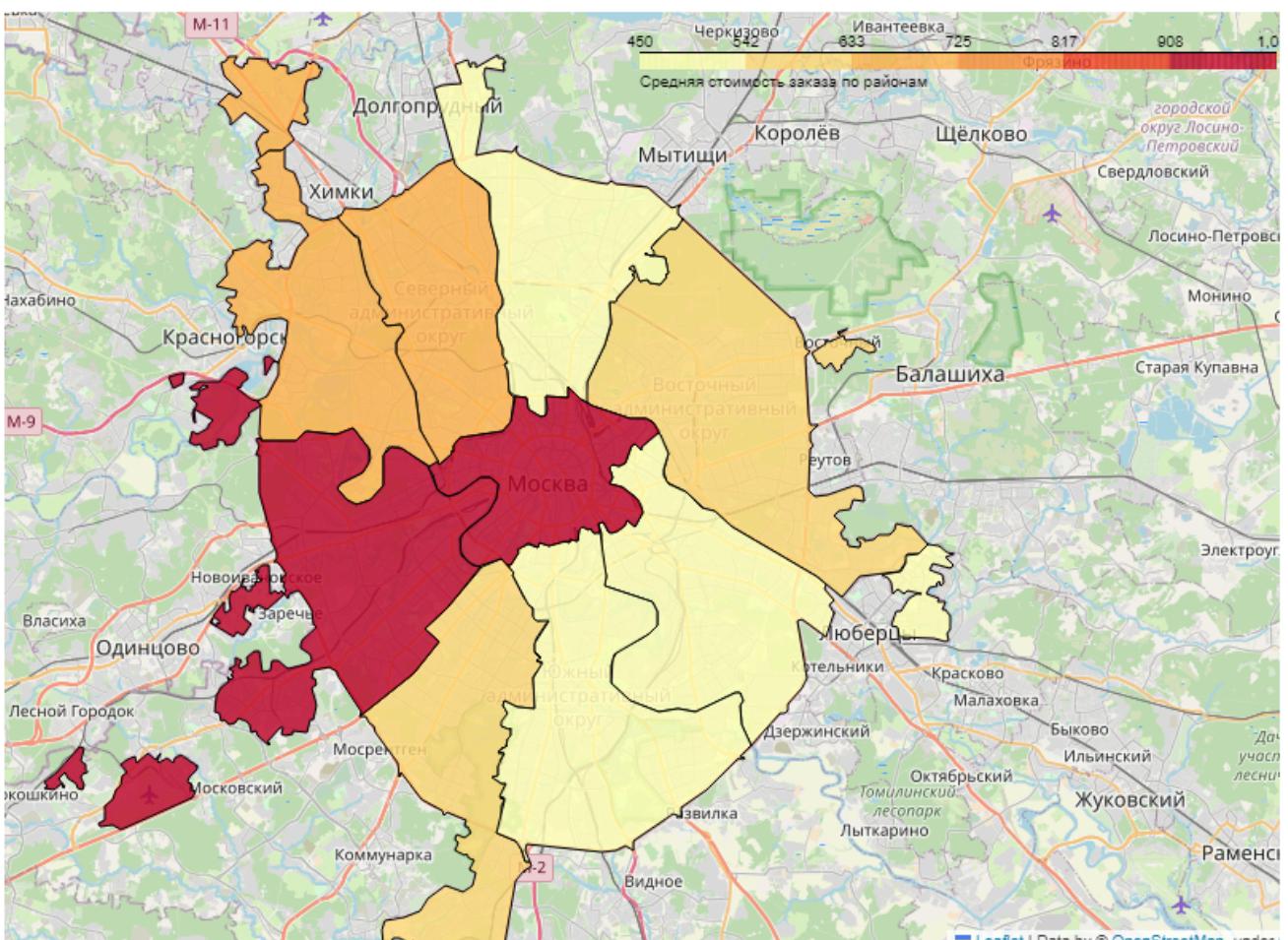


- На топ-15 улиц Москвы - преимущественно стоят кафе, рестораны и кофейни

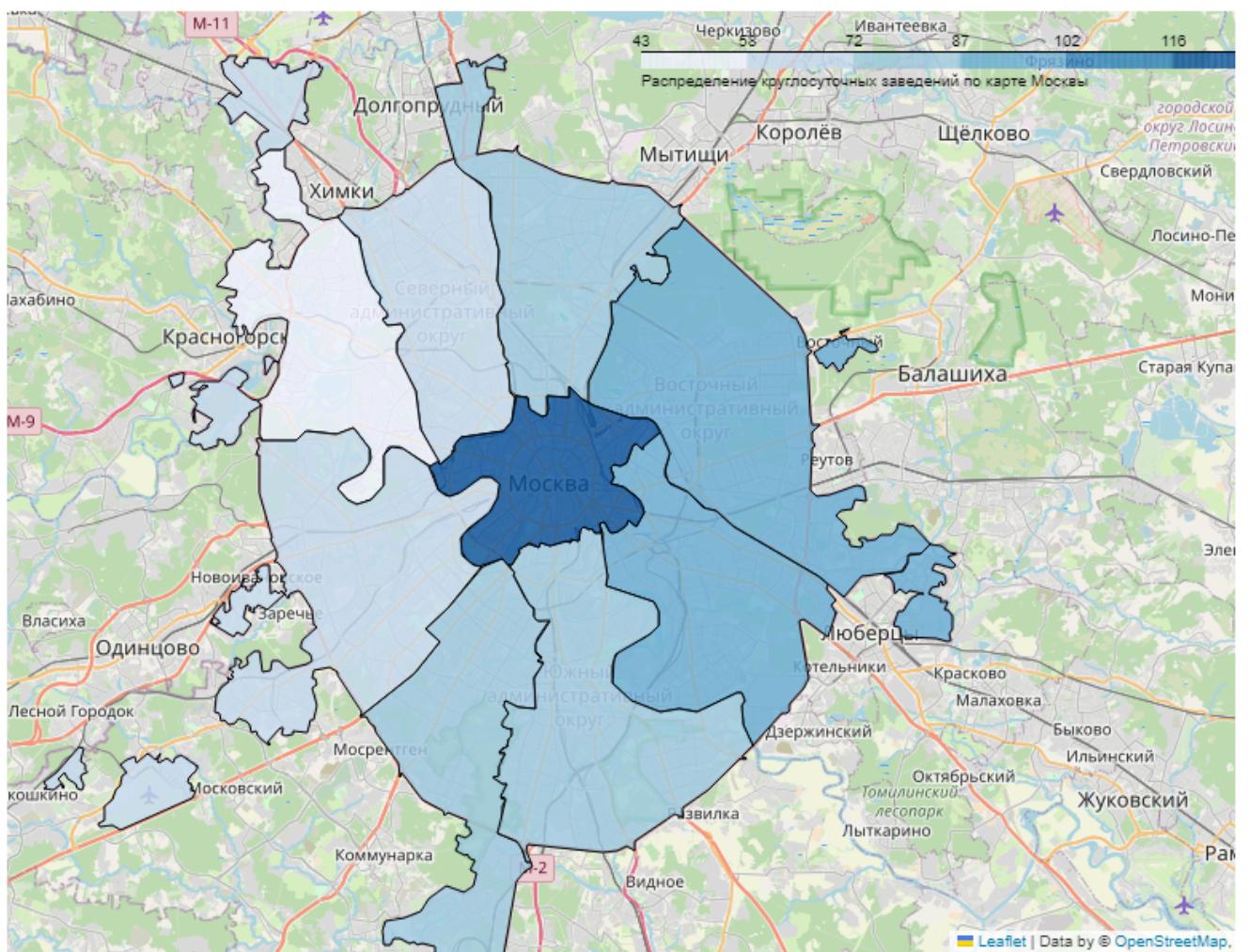
## Типы заведений на топ-15 улиц Москвы



- Самый дорогой средний чек в районах - ЗАО и ЦАО



- Большинство круглосуточных заведений находится в центре Москвы (район - ЦАО, сосредоточение туристов).

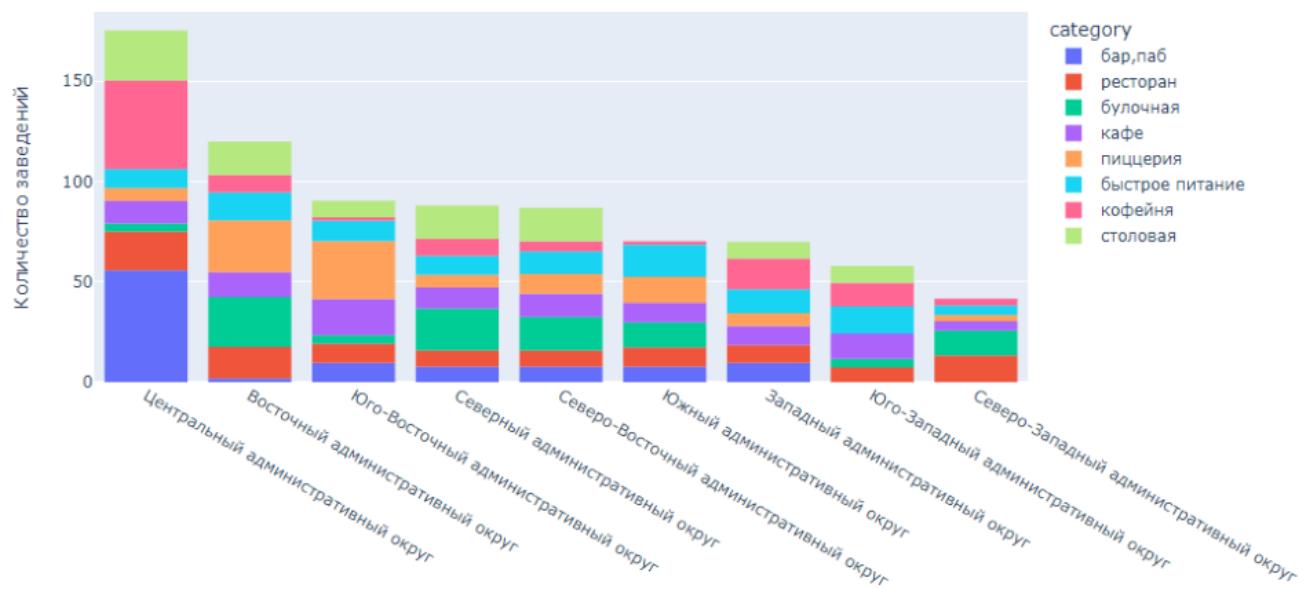


- Самые популярные категории заведений обещпита с графиком работы 24/7 -

- Кафе
- Быстрое питание
- Ресторан

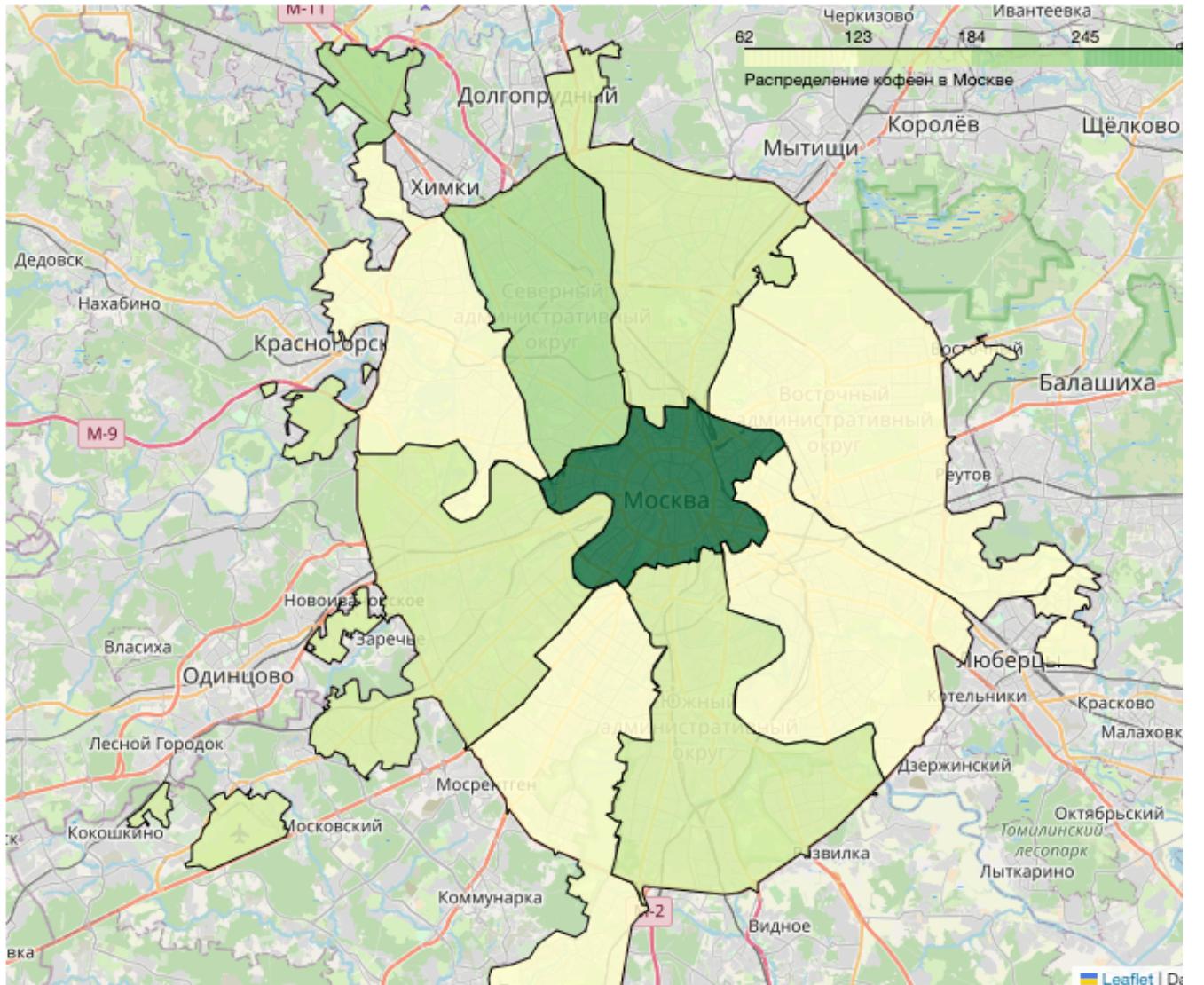
- В центре (район ЦАО) - больше круглосуточных баров/пабов и ресторанов

Круглосуточные заведения в Москве - по районам и типам

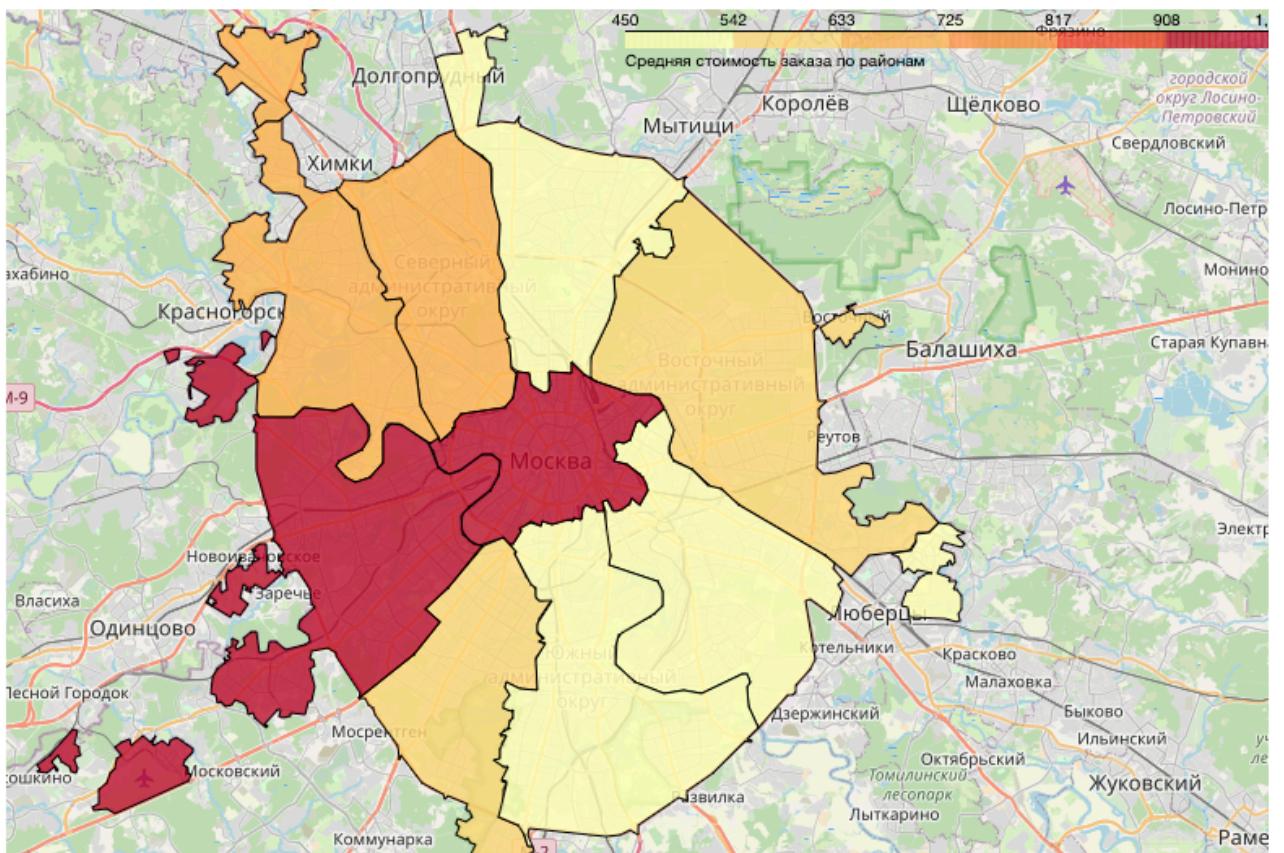


## Предложения инвесторам по открытию кофейни -

1. Т.к. мечта инвесторов фонда открыть уникальное заведение - **будем рассматривать** открытие уникальной кофейни - **несетевого**. Количество несетевых кофеен в Москве - 693, что составляет 49% от всех кофеен.
2. Предлагаю рассмотреть инвесторам для открытия своей кофейни - Юго-Западный район Москвы - здесь меньше конкуренция (занимает 7 из 9 мест по количеству кофеен), а так же согласно нашему исследованию - Данный район входит в ТОП-5 районов по цене среднего чека заказа - то есть доля платежеспособного населения - велика.



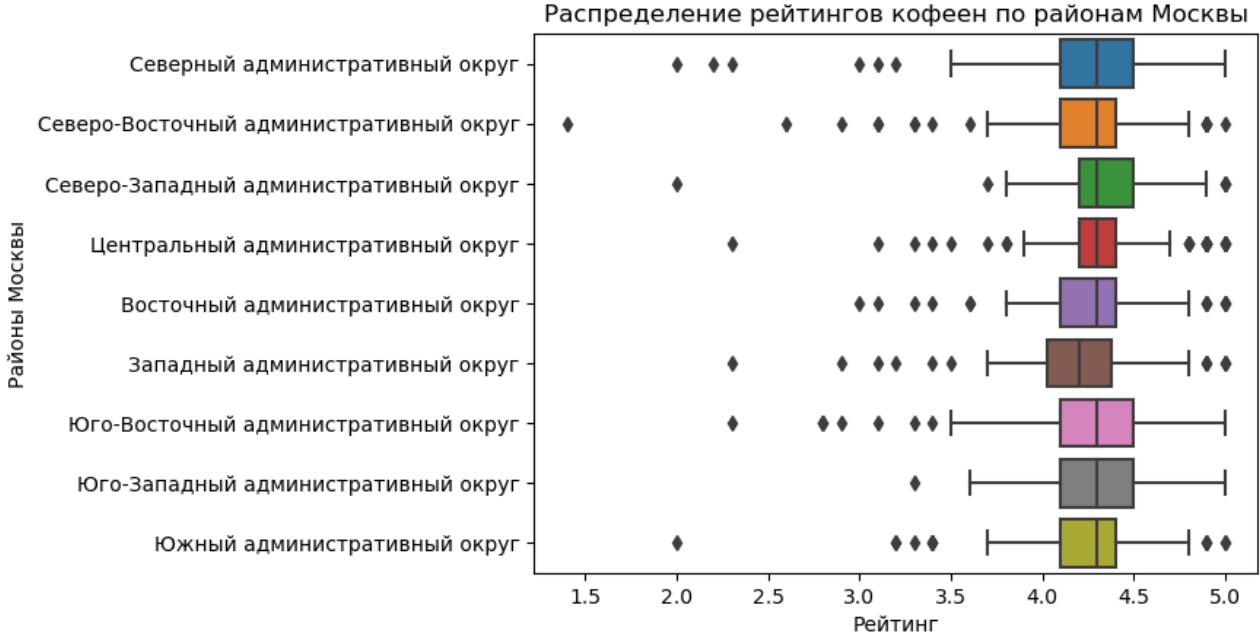
	district	middle_avg_bill
0	Западный административный округ	1000.0
1	Центральный административный округ	1000.0
2	Северо-Западный административный округ	700.0
3	Северный административный округ	650.0
4	Юго-Западный административный округ	600.0



3. Кофеен, работающих круглосуточно - всего 4%. Для нового кафе - предлагаю **не использовать круглосуточный график работы**, для сокращения издержек на зарплату сотрудникам и других сопутствующих расходов.

В дальнейшем - возможно будет рассмотреть переход на круглосуточный режим работы, в зависимости от потока людей.

1. Медианный рейтинг кафеен во всех районах равен ~ 4.3 - то есть, в целом, заведения данной категории популярны у местных жителей.



## 2. Медианная цена чашки кофе в Москве - 170 рублей.

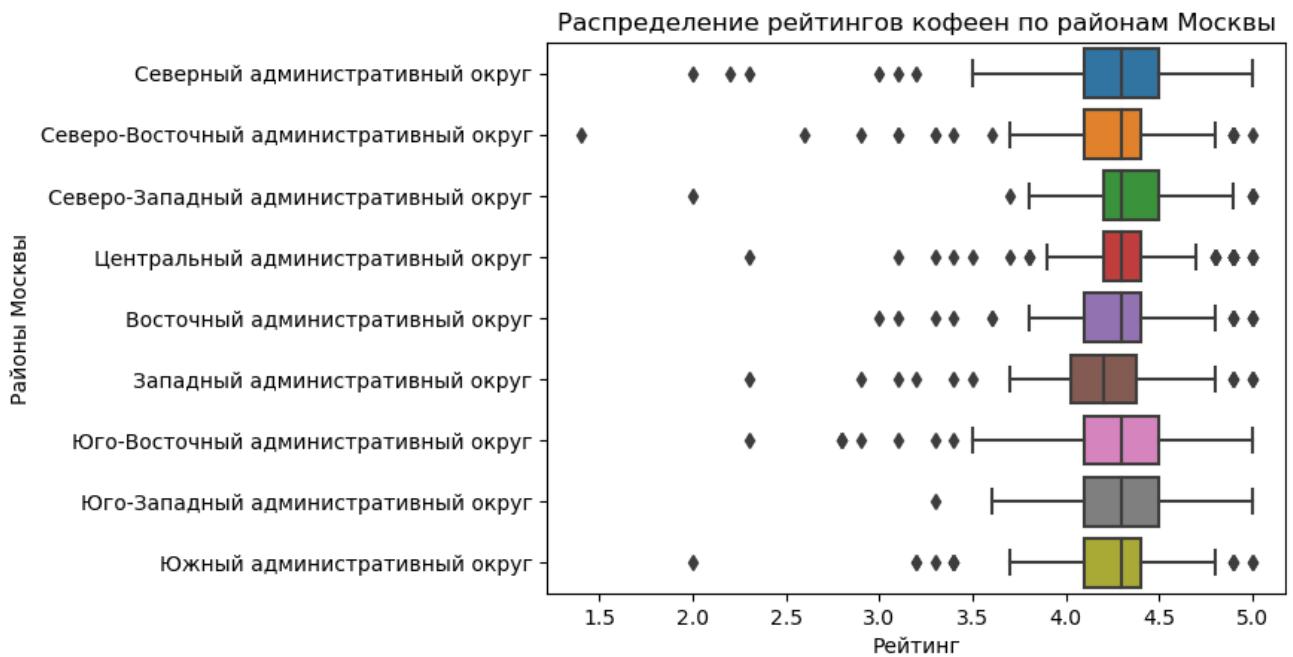
Для инвесторов я рекомендую ориентироваться на цену кружки кофе (170-180 рублей), т.к. медианная цена в районе ЮЗАО - 198 рублей, что даже выше чем в ЦАО (190 рублей).

Данная цена (~ 170-180 рублей) будет конкурентным преимуществом кофейни, а также удовлетворит условию инвесторов открыть доступную кофейню.

1. Медианное значение по количеству посадочных мест по всем нетесетевым кофейням Москвы - 60 шт. Нашим инвесторам я рекомендую ориентироваться на 60 посадочных мест в заведении

	district	seats
0	Западный административный округ	98.0
1	Южный административный округ	80.0
2	Северный административный округ	70.0
3	Юго-Восточный административный округ	54.0
4	Восточный административный округ	52.0
5	Северо-Восточный административный округ	52.0
6	Центральный административный округ	50.0
7	Юго-Западный административный округ	50.0
8	Северо-Западный административный округ	25.5

2. Медианный рейтинг кофеен по всей Москве - 4.3. в целом, заведения данной категории популярны у местных жителей.



## Презентация

<https://disk.yandex.ru/i/1J1fiDCIc9lhAQ>