Name:SHEIK PAREETH

zomato dataset

1. Importing the libraries

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sb
import plotly.express as px
import plotly.io as pio
pio.renderers.default = "svg"
```

2. Importing the dataset

In [2]:

```
# Load the dataset
# Delemeter is character and it is used to seperate the data transporting to another applic
# ("|") this is pipe symbol
dataset=pd.read_csv("Zomato_Mumbai.csv",delimiter='|')
```

In [3]:

```
dataset
```

Out[3]:

	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Mumbai	First International Financial Centre-- Bandra ...	https://www.zomato.
1	Baba Falooda	400	Desserts,Ice Cream,Beverages	Mumbai	Mahim	https://www.zomato
2	Chin Chin Chu	1800	Asian,Chinese	Mumbai	Juhu	https://www.zomato.co
3	Butterfly High	1000	Modern Indian	Mumbai	Bandra Kurla Complex	https://www.zomato.co
4	BKC DIVE	1200	North Indian,Chinese,Continental	Mumbai	Bandra Kurla Complex	https://www.zomato.co
...	...	...	...	...	...	
15076	Hari Om Snack Bar	350	Fast Food,South Indian,Chinese	Mumbai	Kandivali West	https://www.zomato.c
15077	PitaBurg	400	Fast Food,Lebanese	Mumbai	Lower Parel	https://www.zomato.co
15078	Uncha Otlawala	300	Desserts,Ice Cream	Mumbai	Kandivali West	https://www.zomato.
15079	Mandarin Panda	400	Desserts,Chinese,Thai	Mumbai	Malad West	https://www.zomato.com
15080		NaN	NaN	NaN	NaN	

15081 rows × 12 columns

In [4]:

```
# using .head() is shows first 5 rows
dataset.head()
```

Out[4]:

	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Mumbai	First International Financial Centre--Bandra ...	https://www.zomato.com/mu
1	Baba Falooda	400	Desserts,Ice Cream,Beverages	Mumbai	Mahim	https://www.zomato.com/m fa
2	Chin Chin Chu	1800	Asian,Chinese	Mumbai	Juhu	https://www.zomato.com/
3	Butterfly High	1000	Modern Indian	Mumbai	Bandra Kurla Complex	https://www.zomato.com/mun
4	BKC DIVE	1200	North Indian,Chinese,Continental	Mumbai	Bandra Kurla Complex	https://www.zomato.com/ d

3. Getting Basic Information about the Dataset

In [5]:

```
# shape is number of rows and columns are presented this dataset
dataset.shape
```

Out[5]:

```
(15081, 12)
```

In [6]:

```
# size is total value of dataset
dataset.size
```

Out[6]:

```
180972
```

In [7]:

```
# Check the information about the dataset
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15081 entries, 0 to 15080
Data columns (total 12 columns):
NAME               15081 non-null object
PRICE              15080 non-null object
CUSINE_CATEGORY    15079 non-null object
CITY               15080 non-null object
REGION             15080 non-null object
URL                15080 non-null object
PAGE NO            15080 non-null object
CUSINE TYPE        15080 non-null object
TIMING             15015 non-null object
RATING_TYPE        15080 non-null object
RATING             15080 non-null object
VOTES              15080 non-null object
dtypes: object(12)
memory usage: 1.4+ MB
```

In [8]:

```
# returns description of the data in the dataset
dataset.describe()
```

Out[8]:

	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMIN
count	15081	15080	15079	15080	15080	15080	15080	15080	150
unique	12720	67	3183	2	241	13823	944	23	25
top	NAME	400	CUSINE_CATEGORY	Mumbai	REGION	URL	PAGE NO	Quick Bites	11am 11pm(Mo Su
freq	942	2042	942	14138	942	942	942	5262	11

4. Cleaning the Dataset

a. Removing the redundunt rows of data

In [9]:

```
# Checking redundunt rows of data
# page no is a wrong unwanted row so i will remove the row
wrong_data = dataset['PAGE NO'] == 'PAGE NO'
dataset[wrong_data]
```

Out[9]:

	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
15	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
31	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
47	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
63	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
79	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
...	...	...	...	...	...	...	...	...	...	
15000	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
15016	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
15032	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
15048	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT
15064	NAME	PRICE	CUSINE_CATEGORY	CITY	REGION	URL	PAGE NO	CUSINE TYPE	TIMING	RAT

942 rows × 12 columns

In [10]:

```
## Performing Negation of the wrong dataset and then storing the correct data back in the a
## This permamnently remove the wrong data from the original dataframe
dataset = dataset[~wrong_data]
```

In [11]:

```
# Dropping columns which are not required for further analysis
# URL,PAGE NO,CITY these columns also wrong unwanted columns
# axis=1 is represents columns
# inplace = true  the data is modified in place, which means it will return nothing and the
dataset.drop(['URL', 'PAGE NO', 'CITY'], axis = 1, inplace=True)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\pandas\core\frame.py:4102: Settin
gWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/s
table/user_guide/indexing.html#returning-a-view-versus-a-copy (http://panda
s.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-ve
rsus-a-copy)
```

In [12]:

```
# check the dataset after removed the wrong value
dataset.head()
```

Out[12]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATING_TYF
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	First International Financial Centre-- Bandra ...	Casual Dining	12noon to 130am(Mon-Sun)	Excelle
1	Baba Falooda	400	Desserts,Ice Cream,Beverages	Mahim	Dessert Parlor	2pm to 1am(Mon-Sun)	Very Goo
2	Chin Chin Chu	1800	Asian,Chinese	Juhu	Casual Dining	12noon to 1am(Mon-Sun)	Very Goo
3	Butterfly High	1000	Modern Indian	Bandra Kurla Complex	Bar	12noon to 130am(Mon-Sun)	Very Goo
4	BKC DIVE	1200	North Indian,Chinese,Continental	Bandra Kurla Complex	Bar	1130am to 1am(Mon-Sun)	Veľmi dob

b. Removing the Null Records

In [13]:

```
# Checking for Null records
dataset.isnull().sum()
```

Out[13]:

```
NAME                0
PRICE               1
CUSINE_CATEGORY     2
REGION              1
CUSINE TYPE         1
TIMING             66
RATING_TYPE         1
RATING              1
VOTES               1
dtype: int64
```

In [14]:

```
# Checking for a null row
dataset[dataset['PRICE'].isnull()]
```

Out[14]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATING_TYPE	RATING
15080		NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [15]:

```
# Droping the above row from the dataset
# axis=0 is represents rows
dataset=dataset.drop(labels=15080, axis=0)
```

In [16]:

```
# Replacing the other null values with 0
dataset.fillna('0', inplace=True)
```

In [17]:

```
# Confirming all the null values are correct
dataset.isnull().sum()
```

Out[17]:

```
NAME               0
PRICE              0
CUSINE_CATEGORY    0
REGION             0
CUSINE TYPE        0
TIMING             0
RATING_TYPE        0
RATING             0
VOTES              0
dtype: int64
```

c. Converting the DataTypes of numerical columns to numeric dataype

In [18]:

```
# Checking for text values in the column before converting it to numeric datatype
dataset['RATING'].value_counts()
```

Out[18]:

```
-          2360
3.5        1094
3.4        1036
3.6         960
NEW         953
3.3         926
3.7         917
3.2         801
3.8         782
3.1         734
3.0         622
3.9         596
2.9         409
4.0         408
2.8         309
4.1         298
4.2         199
2.7         170
4.3         148
4.4          99
2.6          77
Opening      57
4.5          46
2.5          39
4.6          32
2.4          26
4.7          13
2.3          10
2.1           5
4.8           4
2.2           4
4.9           2
2.0           1
1.8           1
Name: RATING, dtype: int64
```

In [19]:

```
# Replacing the text values with '0' for ratings
dataset['RATING'].replace(to_replace=['-','NEW','Opening'], value='0', inplace=True)
```

In [20]:

```
# Checking for text values in the column before converting it to numeric datatype
dataset['VOTES'].value_counts()
```

Out[20]:

```
-       2360
NEW      953
4        364
5        320
6        288
        ... 
534        1
1229       1
1957       1
3          1
1242       1
Name: VOTES, Length: 1123, dtype: int64
```

In [21]:

```
# Replacing the text values with '0' for votes
dataset['VOTES'].replace(to_replace=['-','NEW','Opening'], value='0', inplace=True)
```

In [22]:

```
# Changing Data Type of the numerical columns
dataset['PRICE'] = dataset['PRICE'].astype('int64')
dataset['RATING'] = dataset['RATING'].astype('float64')
dataset['VOTES'] = dataset['VOTES'].astype('int64')
```

In [23]:

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 14138 entries, 0 to 15079
Data columns (total 9 columns):
NAME                 14138 non-null object
PRICE                14138 non-null int64
CUSINE_CATEGORY      14138 non-null object
REGION               14138 non-null object
CUSINE TYPE          14138 non-null object
TIMING               14138 non-null object
RATING_TYPE          14138 non-null object
RATING               14138 non-null float64
VOTES                14138 non-null int64
dtypes: float64(1), int64(2), object(6)
memory usage: 1.1+ MB
```

d. Working with 'Timing' column

In [24]:

```
# Time and date are presented in one column
dataset['TIMING'].value_counts()
```

Out[24]:

```
11am to 11pm(Mon-Sun)                        1192
11am to 12midnight(Mon-Sun)                   632
12noon to 12midnight(Mon-Sun)                 467
11am to 1130pm(Mon-Sun)                       309
10am to 10pm(Mon-Sun)                         267
                                             ... 
4pm to 1150pm,12midnight to 4am(Mon-Sun)        1
9am to 5pm,7pm to 1130pm(Mon-Sun)               1
1130am to 330pm,615pm to 11pm(Mon-Sun)          1
730am to 1030am,12noon to 1130pm(Mon-Sun)       1
1030am to 8pm(Mon-Sun)                          1
Name: TIMING, Length: 2551, dtype: int64
```

In [25]:

```
# Splitting the column and storing it in temp_dataset dataset
temp_dataset = dataset['TIMING'].str.split("(", n = 1, expand = True)
```

In [26]:

```
temp_dataset
```

Out[26]:

	0	1
0	12noon to 130am	Mon-Sun)
1	2pm to 1am	Mon-Sun)
2	12noon to 1am	Mon-Sun)
3	12noon to 130am	Mon-Sun)
4	1130am to 1am	Mon-Sun)
...	...	...
15075	8am to 11pm,12midnight to 115am	Mon-Sun)
15076	11am to 230am	Mon-Sun)
15077	11am to 11pm	Mon,Tue,Wed,Thu,Sun),11am to ...
15078	9am to 1230AM	Mon-Sun)
15079	12noon to 330pm,7pm to 1am	Mon-Sun)

14138 rows × 2 columns

In [27]:

```
# Assigning the columns back to the dataset
# Removing the bracket character from DAYS_OPEN column
dataset['TIMING'] = temp_dataset[0]
dataset['DAYS_OPEN'] = temp_dataset[1]
dataset['DAYS_OPEN'] = dataset['DAYS_OPEN'].str.replace(')','',regex=True)
```

In [28]:

```
dataset.head()
```

Out[28]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATING_TYPE	I
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	First International Financial Centre-- Bandra ...	Casual Dining	12noon to 130am	Excellent	
1	Baba Falooda	400	Desserts,Ice Cream,Beverages	Mahim	Dessert Parlor	2pm to 1am	Very Good	
2	Chin Chin Chu	1800	Asian,Chinese	Juhu	Casual Dining	12noon to 1am	Very Good	
3	Butterfly High	1000	Modern Indian	Bandra Kurla Complex	Bar	12noon to 130am	Very Good	
4	BKC DIVE	1200	North Indian,Chinese,Continental	Bandra Kurla Complex	Bar	1130am to 1am	Veľmi dobré	

In [29]:

```
# Checking for Null values in DAYS_OPEN column
dataset.isnull().sum()
```

Out[29]:

```
NAME                  0
PRICE                 0
CUSINE_CATEGORY       0
REGION                0
CUSINE TYPE           0
TIMING                0
RATING_TYPE           0
RATING                0
VOTES                 0
DAYS_OPEN           160
dtype: int64
```

In [30]:

```
# Replacing the Null values with '0'
dataset.fillna('0', inplace=True)
```

In [31]:

```
# Checking info of all the columns
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 14138 entries, 0 to 15079
Data columns (total 10 columns):
NAME               14138 non-null object
PRICE              14138 non-null int64
CUSINE_CATEGORY    14138 non-null object
REGION             14138 non-null object
CUSINE TYPE        14138 non-null object
TIMING             14138 non-null object
RATING_TYPE        14138 non-null object
RATING             14138 non-null float64
VOTES              14138 non-null int64
DAYS_OPEN          14138 non-null object
dtypes: float64(1), int64(2), object(7)
memory usage: 1.2+ MB
```

e. Removing the restaurant records whose Rating or Votes is 0

In [32]:

```
# Finding those restaurant whose has 0 Rating or Votes
useless_data = (dataset['RATING'] == 0.0) | (dataset['VOTES'] == 0)
dataset[useless_data]
```

Out[32]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RA
32	Hotel Annapoorna Refreshments	400	Maharashtrian,Mughlai,Chinese	Ghansoli	Quick Bites	1030am to 1230AM	
34	Biryani 9	600	Biryani,North Indian	Near Andheri East Station	none	11am to 3am	
36	D Fusion Flavours	350	Chinese	Goregaon East	none	12noon to 330pm,7pm to 3am	
39	Nation Tadka	400	North Indian,South Indian,Chinese,Fast Food	Worli	none	12noon to 1230AM	
83	Link Way Restaurant	500	North Indian,Chinese	Jogeshwari	Quick Bites	12noon to 4pm,8pm to 1am	
...	...	...	...	...	...	...	
14998	Foodies House	0	Chinese	Goregaon East	none	12noon to 4am	
14999	Khansama	0	Biryani	Lower Parel	none	12noon to 3am	
15010	Earth Cafe @ Waterfield	800	Cafe,Healthy Food,Italian,Pizza,Beverages	Linking Road--Bandra West	Café	10am to 10pm	
15023	How About Some Cream	200	Beverages	Mumbai Central	Beverage Shop	12noon to 3am	
15046	Food And Taste Theory	800	Continental,Italian	Phoenix Marketcity--Kurla	Casual Dining	9am to 12midnight	

3371 rows × 10 columns

In [33]:

```
# Performing Negation of the useless dataset and then storing the correct data back in the
# This permamnently remove the wrong data from the original dataset
dataset = dataset[~useless_data]
```

f. Working on 'RATING_TYPE' Column

In [34]:

```
# Checking the unique values in the column
dataset['RATING_TYPE'].value_counts()
```

Out[34]:

```
Average          5111
Good             4330
Very Good        1137
Excellent          95
Poor               47
Veľmi dobré         6
Dobrze              4
Skvělá volba        4
Bardzo dobrze       3
Buono               2
Bueno               2
Promedio            2
Ortalama            2
Průměr              2
Dobré               2
İyi                 2
Bom                 2
Muito Bom           2
Priemer             2
Baik                1
Muy Bueno           1
Çok iyi             1
Biasa               1
Excelente           1
Sangat Baik         1
Média               1
Skvělé              1
Media               1
Velmi dobré         1
Name: RATING_TYPE, dtype: int64
```

In [35]:

```
# Translating the texts into proper English text
dataset['RATING_TYPE'].replace(to_replace='Excelente' , value='Excellent', inplace=True)
dataset['RATING_TYPE'].replace(to_replace=['Veľmi dobré','Bardzo dobrze','Muy Bueno','Velmi
dataset['RATING_TYPE'].replace(to_replace=['Skvělá volba','Dobrze','Bueno','Buono','Dobré',
dataset['RATING_TYPE'].replace(to_replace=['Priemer','Média','Çok iyi'] , value='Average',
dataset['RATING_TYPE'].replace(to_replace=['Průměr','Promedio','Ortalama','Muito Bom','İyi'
dataset['RATING_TYPE'].replace(to_replace=['Baik','Biasa','Media','Sangat Baik'] , value='V
```

In [36]:

```
# Checking all the values correctly mapped
dataset['RATING_TYPE'].value_counts()
```

Out[36]:

```
Average      5115
Good         4347
Very Good    1148
Excellent      96
Poor           57
Very Poor       4
Name: RATING_TYPE, dtype: int64
```

g. Working on 'REGION' Column

In [37]:

```
# Checking the unique values in the column of region
dataset['REGION'].value_counts()
```

Out[37]:

```
Mira Road                              405
Malad West                             308
Chembur                                277
Kharghar                               268
Borivali West                          264
                                      ... 
Hotel Sai Palace Grand-- Malad West      1
Cooling Tower-- Goregaon East            1
Trident-- Bandra Kurla Complex           1
K Star Mall-- Chembur                    1
Ramee Guestline Hotel-- Dadar            1
Name: REGION, Length: 237, dtype: int64
```

In [38]:

```
# Removing the irrelevant text from the Region column
dataset['REGION'] =dataset['REGION'].str.replace('[a-zA-Z].+-- ','',regex=True)
dataset['REGION'].value_counts()
```

Out[38]:

```
Thane West           712
Mira Road            412
Andheri West         407
Malad West           316
Bandra West          282
                    ... 
Kalyan West            2
Goregaon               1
Dadar                  1
CBD Belapur            1
Girgaon Chowpatty      1
Name: REGION, Length: 120, dtype: int64
```

In [39]:

```
# Replacing Small regions with Known region name
dataset['REGION'] = dataset['REGION'].str.replace('4 Bungalows|7 Andheri|Azad Nagar|Near Ar
dataset['REGION'] = dataset['REGION'].str.replace('Bandra Kurla Complex','Bandra',regex=Tru
dataset['REGION'] = dataset['REGION'].str.replace('CBD-Belapur','CBD Belapur',regex=True)
dataset['REGION'] = dataset['REGION'].str.replace('Girgaon Chowpatty','Chowpatty',regex=Tru
dataset['REGION'] = dataset['REGION'].str.replace('Dadar Shivaji Park','Dadar',regex=True)
dataset['REGION'] = dataset['REGION'].str.replace('Flea Bazaar Café|Kamala Mills Compound',
dataset['REGION'] = dataset['REGION'].str.replace('Runwal Green','Mulund',regex=True)
dataset['REGION'] = dataset['REGION'].str.replace('Mumbai CST Area','Mumbai Central',regex=
dataset['REGION'] = dataset['REGION'].str.replace('Kopar Khairane|Seawoods|Turbhe|Ulwe','Na
dataset['REGION'] = dataset['REGION'].str.replace('New Panvel|Old Panvel','Panvel',regex=Tr
dataset['REGION'] = dataset['REGION'].str.replace('Kamothe','Sion',regex=True)
dataset['REGION'] =dataset['REGION'].str.replace('Ghodbunder Road|Majiwada','Thane',regex=T
```

h. Removing Duplicate records

In [40]:

```
# Finding all the duplicate rows
dataset[dataset.duplicated()]
```

Out[40]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RA
4064	Sai Sannidhi Restaurant & Bar	1000	North Indian,Konkan	Dahisar East	Casual Dining	11am to 12midnight	
4068	Konkan Katta	400	Seafood,Maharashtrian,Malwani	Mahakali	Quick Bites	11am to 330pm,630pm to 1130pm	
4082	Usmaniya Hotel	600	Mughlai	Fort	Casual Dining	1030am to 1130pm	
4083	Gina's Cakes	450	Bakery	Dombivali West	none	11am to 11pm	
4084	Konkanastha Lunch Home	400	Seafood,Malwani	Chakala	Casual Dining	12noon to 3pm,730pm to 1030pm	
...	...	...	...	...	...	...	
14200	Mezbaan Family Restaurant	350	Chinese,Mughlai	Mumbra	Dhaba	12noon to 1230AM	
14204	Jyoti Lunch Home	650	Chinese,North Indian,Seafood,Mughlai	Mulund West	Casual Dining	11am to 1230AM	
14253	On Toes	900	Italian,North Indian,Chinese	Malad West	Casual Dining	12noon to 3pm,7pm to 1230AM	
14761	Frosty Farm	400	Ice Cream,Desserts,Fast Food	Malad East	Dessert Parlor	1pm to 1215AM	
14928	Shree Manu Sagar	300	North Indian,Chinese,Indian	Ghansoli	Quick Bites	1130am to 415pm,7pm to 1215AM	

220 rows × 10 columns

In [41]:

```
# Dropping all the duplicate rows
dataset = dataset.drop_duplicates()
```

4. Copying the cleaned data into a new DataFrame

In [42]:

```
zomato_data = dataset.copy()
```

In [43]:

```
zomato_data.head()
```

Out[43]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATING_TYPE	RAT
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Bandra	Casual Dining	12noon to 130am	Excellent	
1	Baba Falooda	400	Desserts,Ice Cream,Beverages	Mahim	Dessert Parlor	2pm to 1am	Very Good	
2	Chin Chin Chu	1800	Asian,Chinese	Juhu	Casual Dining	12noon to 1am	Very Good	
3	Butterfly High	1000	Modern Indian	Bandra	Bar	12noon to 130am	Very Good	
4	BKC DIVE	1200	North Indian,Chinese,Continental	Bandra	Bar	1130am to 1am	Very Good	

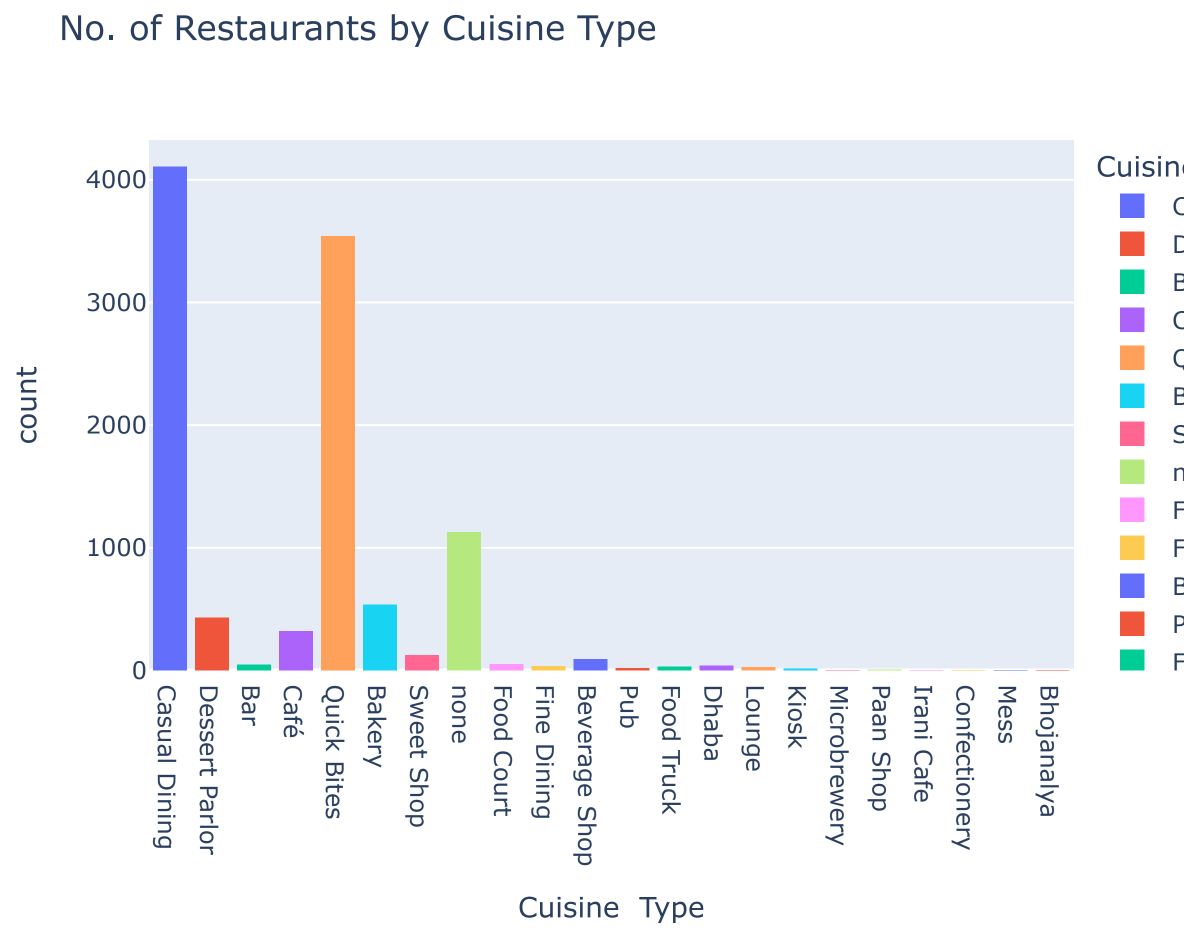
5. Performing Exploratory Data Analysis

Q1) How many restaurants are in Mumbai for each type of cuisine?

In [44]:

```
# draw the graph use ploty.express
fig = px.histogram(zomato_data, x='CUSINE TYPE', color='CUSINE TYPE', title= 'No. of Restau
                                                                  labels={'CUSINE

fig.show()
```



Q2) What are the percentage of restaurants by Rating Type in Mumbai?

In [45]:

```
rating_data = zomato_data['RATING_TYPE'].value_counts().reset_index()
rating_data.rename(columns={'index':'RATING TYPE', 'RATING_TYPE':'COUNT OF RESTAURANTS'}, i
rating_data
```

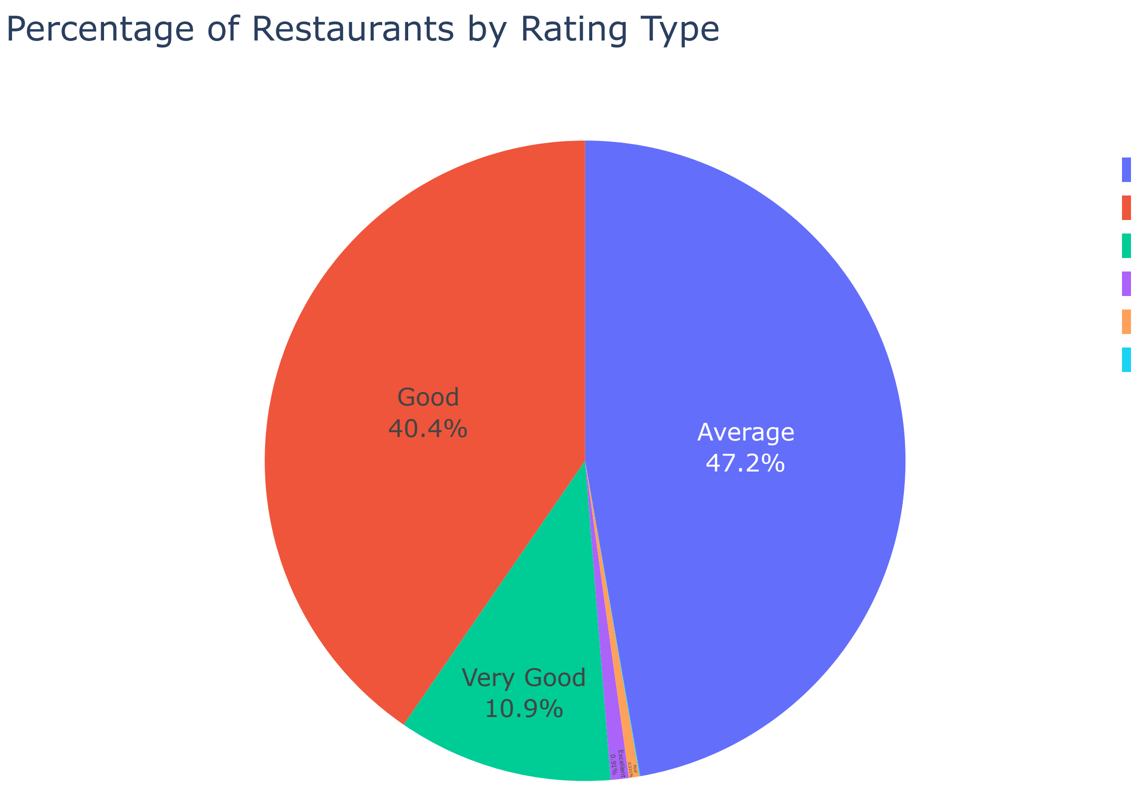
Out[45]:

	RATING TYPE	COUNT OF RESTAURANTS
0	Average	4983
1	Good	4263
2	Very Good	1145
3	Excellent	96
4	Poor	56
5	Very Poor	4

In [46]:

```
# draw the pie chart graph use ploty.express
fig = px.pie(rating_data, names='RATING TYPE', values='COUNT OF RESTAURANTS', color='RATING
             title='Percentage of Restaurants by Rating Type').update_traces(textposition='

fig.show()
```



Q3) Which are the Top 10 highest rated Seafood Restaurant in Mumbai?

In [47]:

```
seafood_data = zomato_data[zomato_data['CUSINE_CATEGORY'].str.contains('Seafood')]
seafood_data.sort_values(by='RATING',ascending=False).head(10)
```

Out[47]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	
7104	Thangabali	1000	Seafood,South Indian,Mangalorean,Andhra,Kerala	Khar	Bar	1 4pr
76	Ceremonial Kitchen & Co	1000	Seafood,Maharashtrian,North Indian,Chinese	Thane	Casual Dining	11
13685	Maharashtra Lunch Home	600	Maharashtrian,Malwani,Konkan,Seafood	Kharghar	Casual Dining	345 to
12433	Quarter Canteen	1100	North Indian,Seafood,Chinese	Bandra West	Casual Dining	1 330
902	The Harbour Bay - SeaFood Kitchen & Bar	2400	Seafood,Beverages	Bandra West	Casual Dining	1
884	Rajmanya-Seafood family restaurant	800	Maharashtrian,Konkan,Seafood	Vashi	Casual Dining	
3380	Peco Peco	700	Chinese,Seafood,Asian	Powai	none	1 330 to
9954	Pi Bar and Kitchen	1600	Continental,European,Italian,Seafood,Pizza,Des...	Andheri West	Bar	1 6pr 12
903	Ferry Wharf	1500	Seafood,Mangalorean	Bandra West	Casual Dining	330 to
915	Monis Bar and Restaurant	1000	North Indian,Chinese,Continental,Seafood,Bever...	Thane West	Casual Dining	11 330 to

Q4) Which is the best Food Truck in Mumbai?

In [48]:

```
foodtruck_data = zomato_data[zomato_data['CUSINE TYPE'] == 'Food Truck']
foodtruck_data.sort_values(by='RATING',ascending=False).head(2)
```

Out[48]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATING_TYPE	RATING
262	Dumpling Delights	200	Momos	Matunga East	Food Truck	430pm to 930pm	Very Good	4.3
1017	Street Food Co.	250	Fast Food,Chinese	Virar	Food Truck	6pm to 3am	Very Good	4.1

Q5) Which places have the highest rated restaurant for each Cuisine Type in Mumbai?

In [49]:

```
# Assuming restaurants having rating above 4.5
highest_rated_data = zomato_data[zomato_data['RATING'] >= 4.5]
highest_rated_data
```

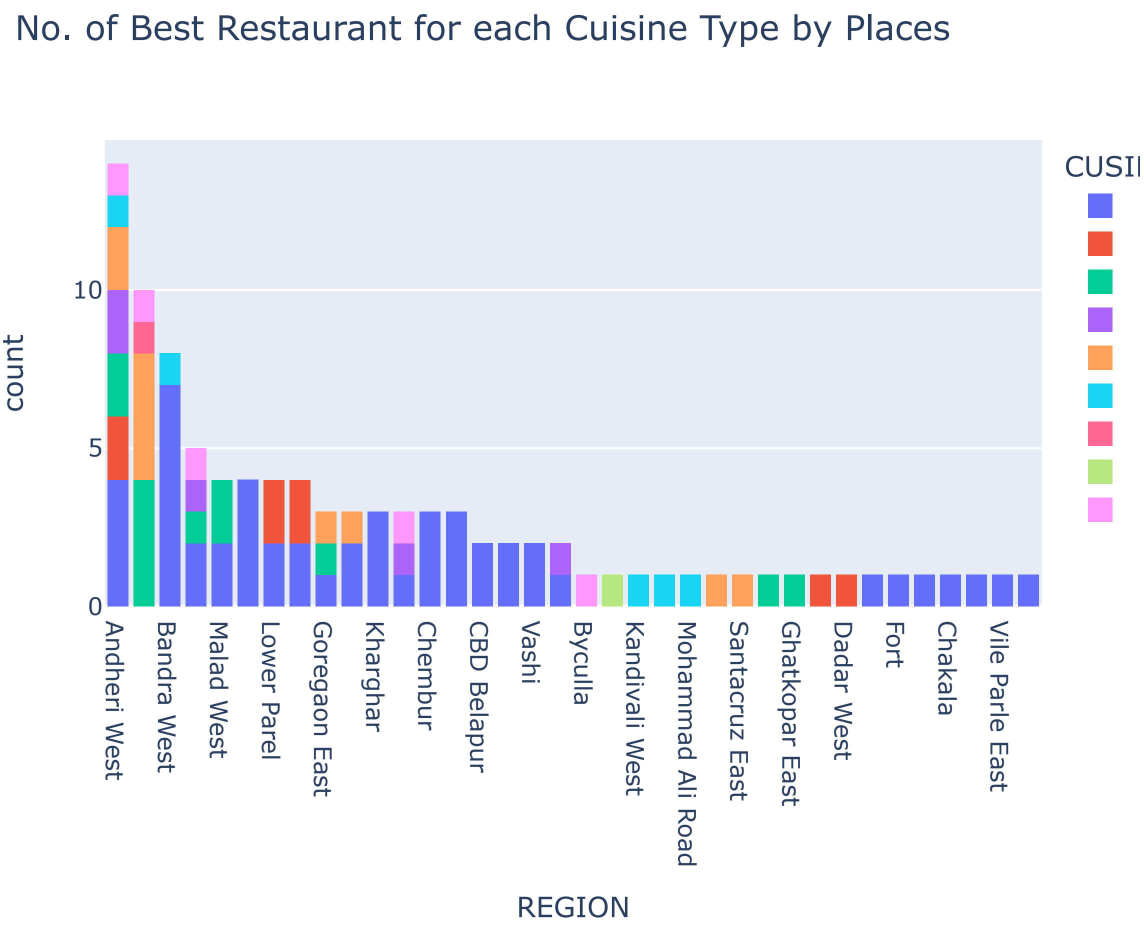
Out[49]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMIN
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Bandra	Casual Dining	12noon 130a
6	Persian Darbar	1300	Biryani,North Indian,Chinese,Mughlai	Marol	Casual Dining	10am 3a
7	Tanatan	1500	Modern Indian	Juhu	Casual Dining	12noon 130a
9	Plum by Bent Chair	1800	Asian	Lower Parel	Casual Dining	12noon 1a
10	Angrezi Dhaba	1500	North Indian,Chinese,Thai,European	Dadar West	Bar	12noon 1a
...	...	...	...	...	...	
14228	Zaika Crave - Club Aquaria	1300	North Indian,Continental,Chinese,Desserts	Borivali West	Casual Dining	11am 330pm,7p to 1130p
14234	Cone Culture	250	European	Kharghar	Casual Dining	Close
15007	Dessertino	300	Desserts,Ice Cream	Kandivali West	Dessert Parlor	11am 12midnig
15051	Tick-eat	800	North Indian,Italian,Chinese,Mexican,Lebanese	Mulund West	Casual Dining	1130am 330pm,7p to 1130p
15056	Daftar Goregaon	750	Pizza,Chinese,North Indian,Beverages	Goregaon East	Casual Dining	12noon 1130p

97 rows × 10 columns

In [50]:

```
fig = px.histogram(highest_rated_data, x='REGION', color='CUSINE TYPE',
                   title= 'No. of Best Restaurant for each Cuisine Type by Places').update_
fig.show()
```



Q6) What is the Avg Price Distibution of highest rated restaurant for each Cuisine Type in Mumbai?

In [51]:

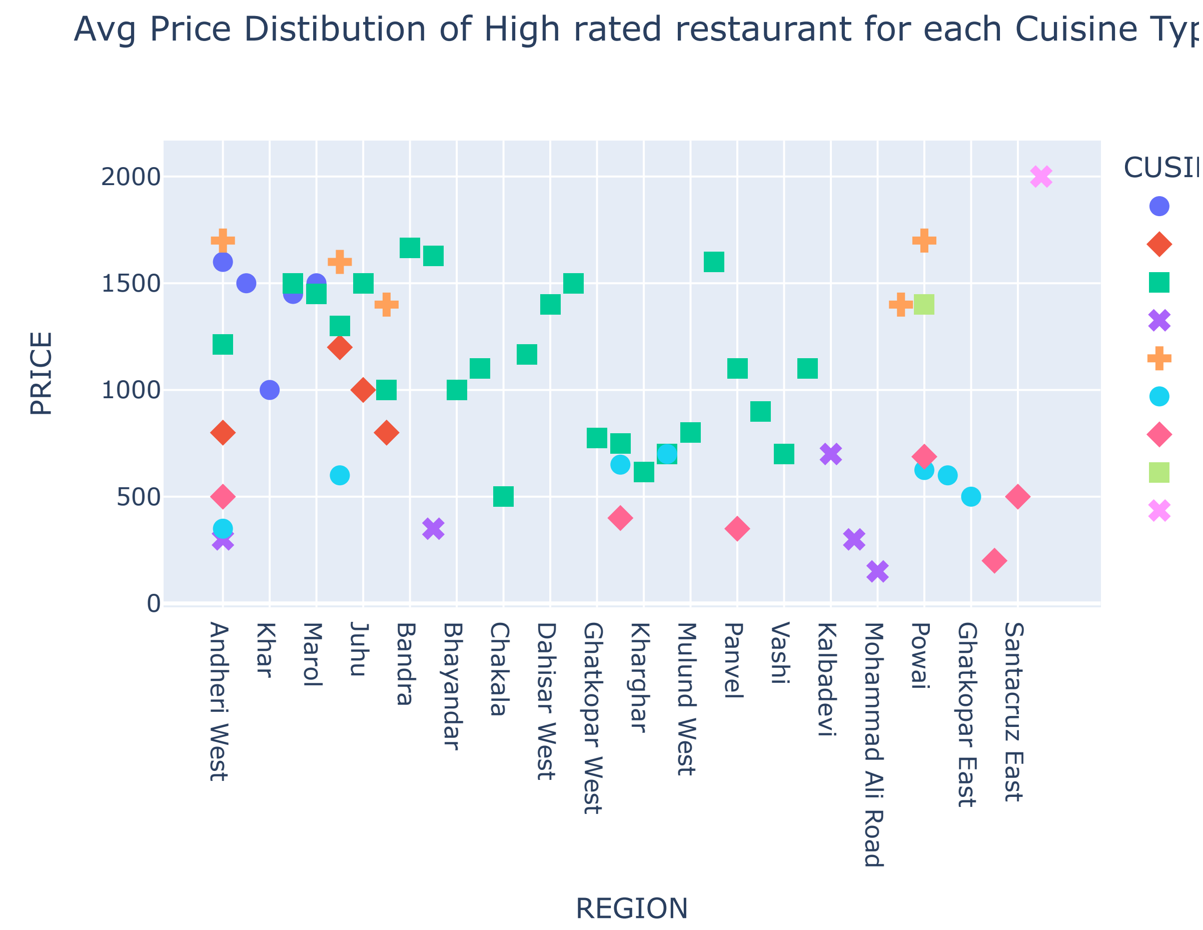
```
highest_rated_price_data = highest_rated_data.groupby(by=['REGION', 'CUSINE TYPE'])['PRICE'
highest_rated_price_data.head()
```

Out[51]:

	REGION	CUSINE TYPE	PRICE
0	Andheri West	Bar	1600.0
1	Andheri West	Café	800.0
2	Andheri West	Casual Dining	1212.5
3	Andheri West	Dessert Parlor	300.0
4	Andheri West	Lounge	1700.0

In [52]:

```
fig = px.scatter(highest_rated_price_data, x="REGION", y="PRICE", color="CUSINE TYPE", symb
                  title=' Avg Price Distibution of High rated restaurant for each Cuisine Ty
fig.show()
```



Q7) Which areas have a large number of Chinese Restaurant Market?

In [53]:

```
chinese_data = zomato_data[zomato_data['CUSINE_CATEGORY'].str.contains('Chinese')]
chinese_data
```

Out[53]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	TIMING	RATIN
0	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Bandra	Casual Dining	12noon to 130am	
2	Chin Chin Chu	1800	Asian,Chinese	Juhu	Casual Dining	12noon to 1am	V
4	BKC DIVE	1200	North Indian,Chinese,Continental	Bandra	Bar	1130am to 1am	V
5	Flea Bazaar Café	800	American,Asian,Street Food,North Indian,Luckno...	Lower Parel	Café	12noon to 1am	V
6	Persian Darbar	1300	Biryani,North Indian,Chinese,Mughlai	Marol	Casual Dining	10am to 3am	
...	...	...	...	...	...	...	
15071	Lucknow Zaika	500	North Indian,Chinese	Kurla	Quick Bites	12noon to 2am	
15072	Zuha's Kitchen	400	Chinese,North Indian,Mughlai	Mumbai Central	Quick Bites	12noon to 4pm,730pm to 430am	
15075	Tirupati Balaji	500	Chinese,Fast Food,North Indian	Andheri West	Casual Dining	8am to 11pm,12midnight to 115am	
15076	Hari Om Snack Bar	350	Fast Food,South Indian,Chinese	Kandivali West	Quick Bites	11am to 230am	
15079	Mandarin Panda	400	Desserts,Chinese,Thai	Malad West	none	12noon to 330pm,7pm to 1am	

5119 rows × 10 columns

In [54]:

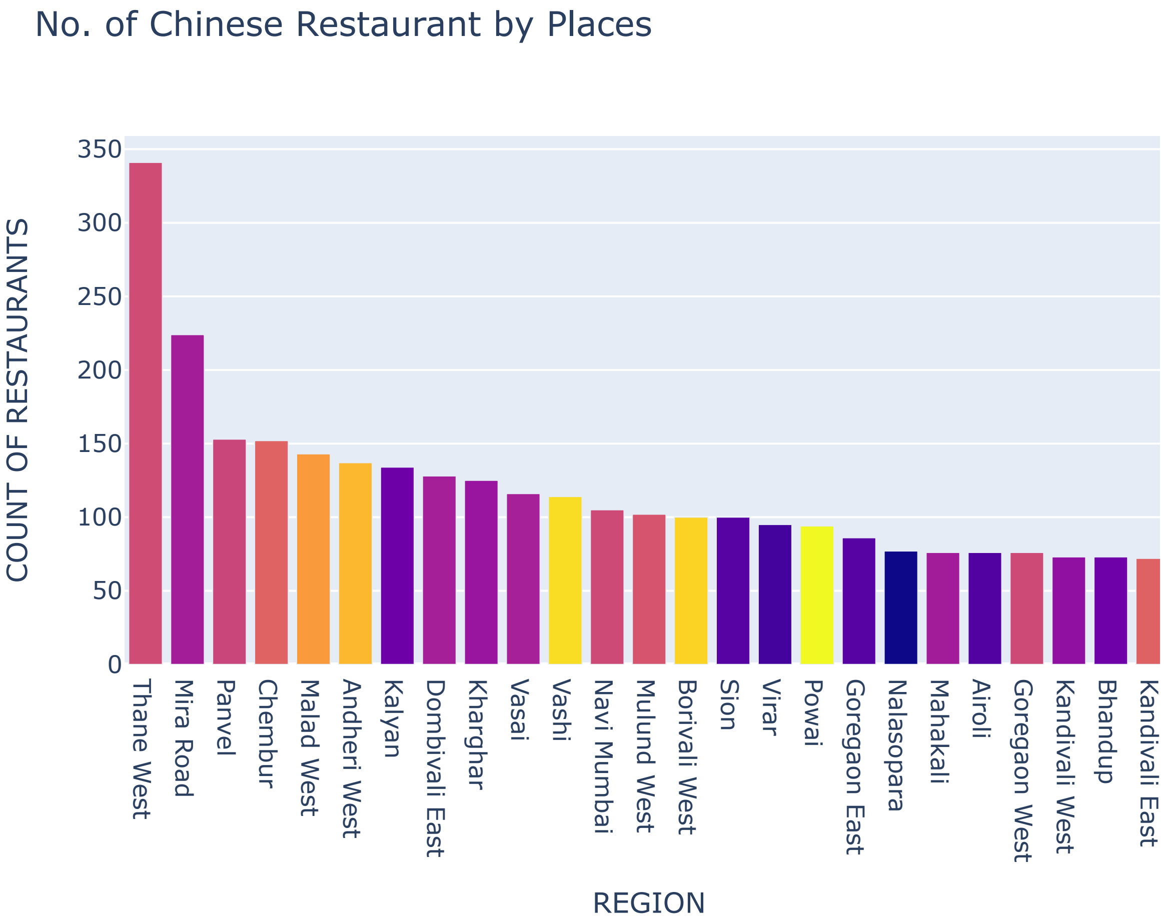
```
chinese_rest_data = chinese_data.groupby(by='REGION').agg({'NAME' : 'count', 'PRICE' : 'mea
chinese_rest_data = chinese_rest_data.sort_values('COUNT OF RESTAURANTS', ascending=False).
chinese_rest_data.head()
```

Out[54]:

	COUNT OF RESTAURANTS	PRICE
REGION		
Thane West	341	589.017595
Mira Road	224	553.348214
Panvel	153	583.660131
Chembur	152	604.934211
Malad West	143	641.608392

In [55]:

```
fig=px.bar(chinese_rest_data, y="COUNT OF RESTAURANTS", color="PRICE", title= "No. of Chine
fig.show()
```



Q8) Is there a relation between Price and Rating by

each Cuisine Type?

In [56]:

```
price_rating_data = zomato_data.groupby(['CUSINE TYPE', 'RATING'])['PRICE'].mean().reset_in
price_rating_data
```

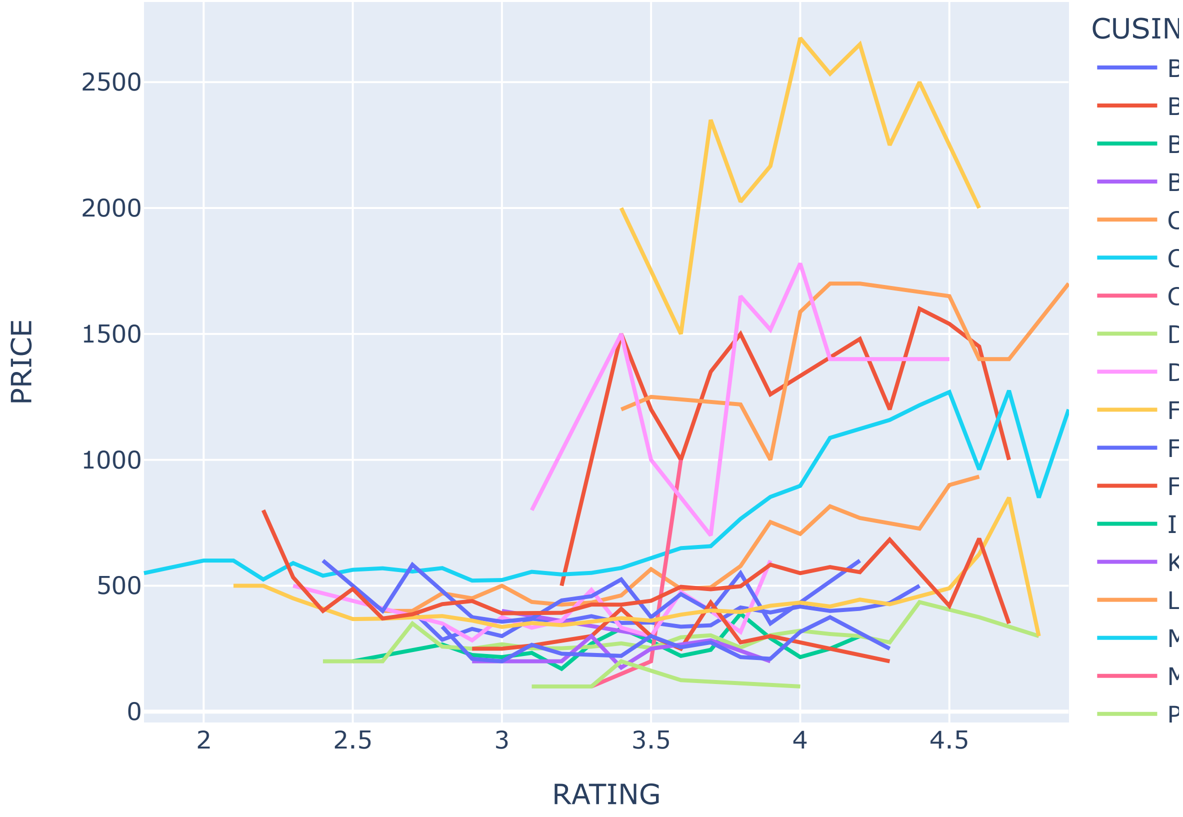
Out[56]:

	CUSINE TYPE	RATING	PRICE
0	Bakery	2.7	400.000000
1	Bakery	2.8	285.714286
2	Bakery	2.9	328.571429
3	Bakery	3.0	300.000000
4	Bakery	3.1	369.117647
...	...	...	...
278	none	4.3	683.333333
279	none	4.4	555.000000
280	none	4.5	420.000000
281	none	4.6	687.500000
282	none	4.7	350.000000

283 rows × 3 columns

In [57]:

```
fig = px.line(price_rating_data, y="PRICE", x="RATING",color='CUSINE TYPE')
fig.show()
```



Q9) Is there a relation between Region and Price?

In [58]:

```
region_price_data = zomato_data.groupby(['REGION'])['PRICE'].mean().reset_index()
region_price_data
```

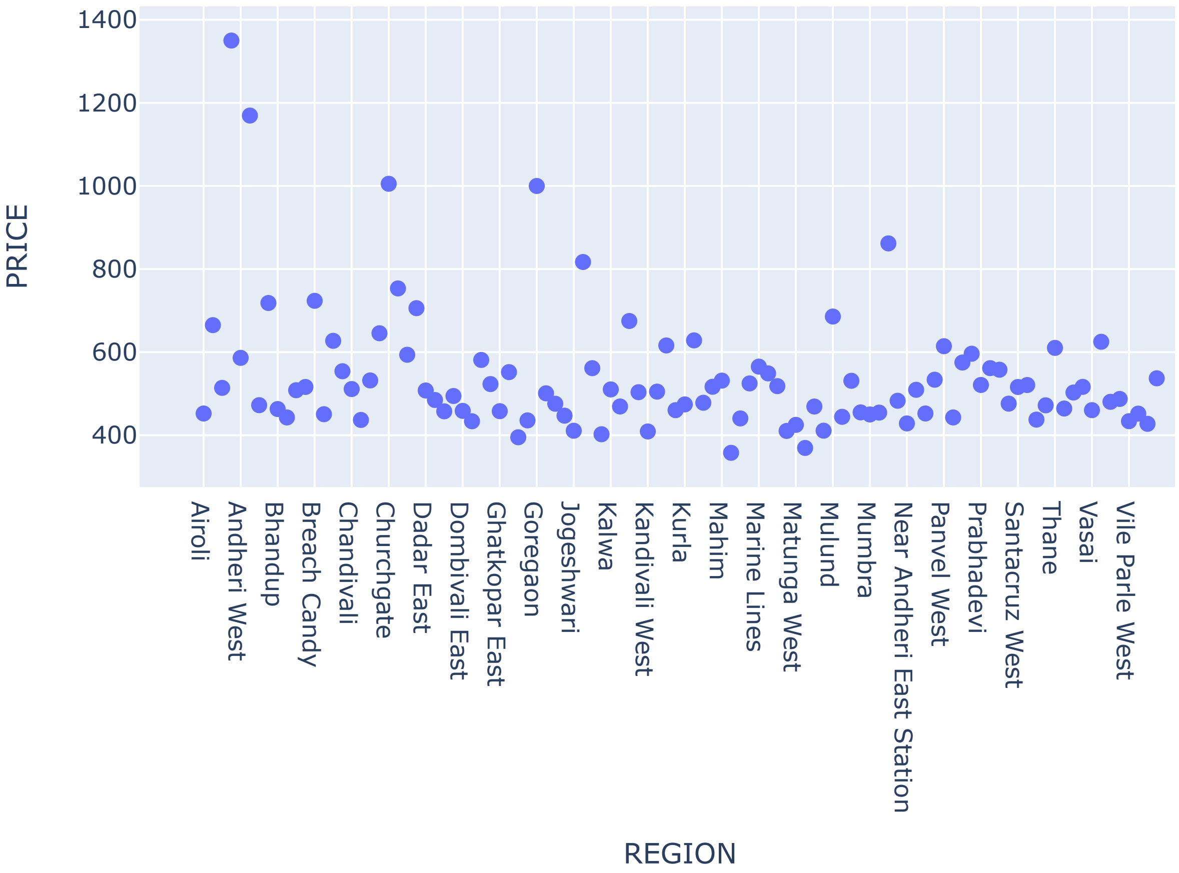
Out[58]:

	REGION	PRICE
0	Airoli	452.287582
1	Alibaug	665.000000
2	Ambernath	514.000000
3	Andheri East	1350.000000
4	Andheri West	586.277916
...	...	...
99	Vile Parle East	487.368421
100	Vile Parle West	433.802817
101	Virar	452.027027
102	Wadala	427.500000
103	Worli	537.012987

104 rows × 2 columns

In [59]:

```
fig = px.scatter(region_price_data, x="REGION", y="PRICE").update_traces(marker_size=8)
fig.show()
```



Q10) Find the list of Affordable Restaurants?

The criteria for Affordable Restaurants would be:

1) Low Price 2) High Rated First step will be to find the restaurants with average cost 1/4th the average cost of most expensive restaurant in our dataframe. Let me explain:-The most expensive restaurant has an average meal cost= 6000. We'll try to stay economical and only pick the restaurants that are 1/4th of 6000.

In [60]:

```
max_price = zomato_data['PRICE'].max()
one_fourth_price = max_price/4
one_fourth_price
```

Out[60]:

```
1250.0
```

In [61]:

```
# Finding list of restaurants that have price less than and equal to 1/4th of the max price
aff_rest_data = zomato_data[['NAME', 'PRICE', 'CUSINE_CATEGORY', 'REGION', 'CUSINE TYPE']]
aff_rest_data = aff_rest_data[aff_rest_data['PRICE'] <= 1250]
aff_rest_data.sort_values(by='PRICE', inplace=True)
aff_rest_data
```

Out[61]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE
6137	Sanjog Wine N Dine	5	North Indian,Chinese	Thane West	Casual Dining
2925	Jab We Eat	50	South Indian,North Indian,Maharashtrian,Fast Food	Girgaum	none
9598	Ho5 Store	50	Fast Food	Matunga West	none
9589	Golden Butterfly	100	Bakery,Desserts	Mira Road	Bakery
5916	Madhuri Puranpoli	100	Maharashtrian	Vile Parle East	none
...	...	...	...	...	...
2740	Peninsula Next	1200	North Indian,Mughlai,Chinese	Sion	Casual Dining
5528	The Thekka	1200	Finger Food,Continental,North Indian,Chinese	Vashi	Lounge
964	Bijoli Grill	1250	Bengali	Powai	Casual Dining
6045	Fabelle at The Chocolate Boutique - ITC Grand ...	1250	Desserts	Parel	Dessert Parlor
7301	SamBar Pub & Kitchen	1250	Finger Food,South Indian,North Indian	Khar	Pub

10190 rows × 5 columns

In [62]:

```
# Finding the highest rated list of restaurants

highrate_rest_data = zomato_data[['NAME', 'PRICE', 'CUSINE_CATEGORY', 'REGION', 'CUSINE TYP
highrate_rest_data = highrate_rest_data[highrate_rest_data['RATING'] >= 4.5]
highrate_rest_data.sort_values(by='PRICE', inplace=True)
highrate_rest_data
```

Out[62]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	RATING
1502	Cake Centre- The Dessert Maker	150	Desserts	Mohammad Ali Road	Dessert Parlor	4.6
763	Curry And Combos Twist	200	North Indian,Chinese	Andheri West	Quick Bites	4.5
807	Moussestruck	200	Desserts	Near Andheri West Station	none	4.5
14234	Cone Culture	250	European	Kharghar	Casual Dining	4.6
725	Belo Pops	300	Ice Cream,Desserts,Beverages	Andheri West	none	4.5
...	...	...	...	...	...	...
5335	Mia Cucina	2000	Italian	Bandra West	Casual Dining	4.5
1786	Global Fusion	2000	Chinese,Japanese,Asian,North Indian	Worli	Fine Dining	4.6
8887	Drifters Tap Station	2000	North Indian,Continental,European,American	Bandra	Casual Dining	4.5
12625	House of Mandarin	2100	Chinese,Sushi,Asian	Bandra West	Casual Dining	4.5
902	The Harbour Bay - SeaFood Kitchen & Bar	2400	Seafood,Beverages	Bandra West	Casual Dining	4.5

97 rows × 6 columns

Now, we'll merge the aff_rest_df with highrate_rest_df to obtain the intersection i.e the list of Affordable Restaurants !!

In [63]:

```
highrate_aff_data = pd.merge(aff_rest_data, highrate_rest_data, how='inner', on=['NAME', 'R
highrate_aff_data = highrate_aff_data[['NAME', 'PRICE_x', 'CUSINE_CATEGORY_x', 'REGION', 'C
highrate_aff_data.rename(columns={'NAME':'NAME', 'PRICE_x':'PRICE', 'CUSINE_CATEGORY_x':'CU
                                'REGION':'REGION', 'CUSINE TYPE_x':'CUSINE TYPE'},inplace=T
```

In [64]:

```
# Affordable Restaurants with low price and high rating
highrate_aff_data
```

Out[64]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE
0	Cake Centre-The Dessert Maker	150	Desserts	Mohammad Ali Road	Dessert Parlor
1	Curry And Combos Twist	200	North Indian,Chinese	Andheri West	Quick Bites
2	Moussestruck	200	Desserts	Near Andheri West Station	none
3	Cone Culture	250	European	Kharghar	Casual Dining
4	Smiley Pops	300	Desserts,Ice Cream,Beverages,Sandwich	Andheri West	Dessert Parlor
...	...	...	...	...	...
60	Wild Dining Restaurant	1200	North Indian,Continental,Mexican,Chinese	Andheri West	Casual Dining
61	Invento	1200	Chinese,Fast Food,North Indian,Italian,Mexican	Lower Parel	Casual Dining
62	Culinary Tales	1200	Chinese,European,Continental,Salad,Italian,Pizza	Panvel	Casual Dining
63	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Bandra	Casual Dining
64	The Joker Bistro	1200	North Indian,Chinese,Continental	CBD Belapur	Casual Dining

65 rows × 5 columns

Q10) Find the list of most Reliable Restaurants?

The criteria for most Reliable Restaurants would be:

1) Low Price 2) High Rated 3) Large No. of Votes First step will be to find the restaurants with Votes greater than Mean of Votes

In [65]:

```
mean_votes = zomato_data['VOTES'].mean()
mean_votes
```

Out[65]:

```
177.2656679624538
```

In [66]:

```
# Finding list of restaurants that have Votes greater than and equal to  Mean of Vote
mean_rest_data = zomato_data[['NAME', 'PRICE', 'CUSINE_CATEGORY', 'REGION', 'CUSINE TYPE',
mean_rest_data = mean_rest_data[mean_rest_data['VOTES'] > 177]
mean_rest_data.sort_values(by='VOTES', inplace=True)
mean_rest_data
```

Out[66]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE	VO
4194	Sai Sagar Veg Treat	500	North Indian,South Indian,Chinese,Fast Food,Be...	Kalyan	Casual Dining	
884	Rajmanya-Seafood family restaurant	800	Maharashtrian,Konkan,Seafood	Vashi	Casual Dining	
3914	Ice Cafe	500	Fast Food,Ice Cream,Beverages,Pizza	Borivali West	Quick Bites	
7897	Konkan Lajjatdar	500	Seafood,Biryani,Beverages,Chinese,Malwani,Konkan	Andheri West	Casual Dining	
3828	Frozen Delight - The Dessert Cafe	250	Desserts,Ice Cream	Airoli	Dessert Parlor	
...	...	...	...	...	...	
8539	Leopold Cafe & Bar	1600	American,Chinese,Mughlai,Italian	Colaba	Casual Dining	7
1251	Joey's Pizza	800	Pizza	Malad West	Quick Bites	7
5337	Chili's American Grill & Bar	1400	American,Mexican,Burger,Tex-Mex	Powai	Casual Dining	7
3751	Prithvi Cafe	700	Cafe,Fast Food	Juhu	Café	8
8897	Candies	700	Cafe,Italian,North Indian,Desserts	Bandra West	Café	10

2345 rows × 6 columns

These are the most reliable, highest rated and affordable restaurants:

We obtain this dataframe by simply taking the intersection of highrate_aff_df & mean_rest_data

This dataframe obtained below shows the restaurants whose:

Cost is below 1250

Rating is above 4.5

Votes are above 17

In [67]:

```
reliable_rest_data = pd.merge(mean_rest_data, highrate_aff_data, how='inner', on=['NAME', '
reliable_rest_data = reliable_rest_data[['NAME', 'PRICE_x', 'CUSINE_CATEGORY_x', 'REGION',
reliable_rest_data.rename(columns={'NAME':'NAME', 'PRICE_x':'PRICE', 'CUSINE_CATEGORY_x':'C
                                 'REGION':'REGION', 'CUSINE TYPE_x':'CUSINE TYPE'},inplace=
```

In [68]:

```
reliable_rest_data
```

Out[68]:

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE
0	Rajmanya-Seafood family restaurant	800	Maharashtrian,Konkan,Seafood	Vashi	Casual Dining
1	Fresh Food Co.	500	Continental,Healthy Food,Salad,Beverages,Desse...	Santacruz East	none
2	Dessertino	300	Desserts,Ice Cream	Kandivali West	Dessert Parlor
3	Invento	1200	Chinese,Fast Food,North Indian,Italian,Mexican	Lower Parel	Casual Dining
4	Maharashtra Lunch Home	600	Maharashtrian,Malwani,Konkan,Seafood	Kharghar	Casual Dining
5	Regano's	600	Continental,Fast Food,Italian,Desserts	Malad West	Casual Dining
6	Big Bang Cuurry	350	North Indian,Biryani,Rolls	Panvel	none
7	Sandy's Den	1000	Fast Food,Bar Food	Chembur	Casual Dining
8	Angrezi Patiyalaa	1200	North Indian,Finger Food,American,Mexican,Chinese	Andheri West	Casual Dining
9	Maezo	1000	Modern Indian	Thane West	Casual Dining
10	Tossin Pizza	900	Pizza,Italian,Fast Food	Chembur	Casual Dining
11	Little West Pizza	600	Pizza	Borivali West	Quick Bites
12	Ceremonial Kitchen & Co	1000	Seafood,Maharashtrian,North Indian,Chinese	Thane	Casual Dining
13	Moussestruck	200	Desserts	Near Andheri West Station	none
14	Daftar Goregaon	750	Pizza,Chinese,North Indian,Beverages	Goregaon East	Casual Dining
15	Poetry By Love & Cheesecake	1000	Cafe,Desserts	Juhu	Café
16	Makhan Singh	800	North Indian,Chinese,Biryani	Powai	none
17	The Joker Bistro	1200	North Indian,Chinese,Continental	CBD Belapur	Casual Dining
18	Cone Culture	250	European	Kharghar	Casual Dining
19	Peco Peco	700	Chinese,Seafood,Asian	Powai	none
20	Shuruwat- Veg Food Journey	600	Continental,Tea,South Indian,Fast Food,Pizza,N...	Ghatkopar West	Casual Dining
21	Justice Cafe and Dine	800	Cafe,Chinese,Italian,Continental,North Indian,...	Thane	Café

	NAME	PRICE	CUSINE_CATEGORY	REGION	CUSINE TYPE
22	Thangabali	1000	Seafood,South Indian,Mangalorean,Andhra,Kerala	Khar	Bar
23	Harsh's Bistro	800	Chinese,Continental	Malad West	Casual Dining
24	Quarter Canteen	1100	North Indian,Seafood,Chinese	Bandra West	Casual Dining
25	Culinary Tales	1200	Chinese,European,Continental,Salad,Italian,Pizza	Panvel	Casual Dining
26	Dum & Curry	700	Mughlai,North Indian,Chinese	Powai	Quick Bites
27	Curry Culture	800	North Indian,Biryani,Chinese,Kebab,Mughlai,Asian	Powai	none
28	Belo Pops	300	Ice Cream,Desserts,Beverages	Andheri West	none
29	Tick-eat	800	North Indian,Italian,Chinese,Mexican,Lebanese	Mulund West	Casual Dining
30	Coppetto Artisan Gelato	350	Ice Cream,Desserts	Bandra West	Dessert Parlor
31	Zaika Restaurant & Party Hall	1000	North Indian,Chinese,Beverages	Bhayandar	Casual Dining
32	Shaollin Temple	1000	Chinese,Thai	CBD Belapur	Casual Dining
33	Spice Republic	1200	Cafe,Continental,Mediterranean,Mexican,Italian...	Borivali West	Café
34	Aquafire Restaurant	1100	North Indian,Continental,Chinese,Italian	Vile Parle East	Casual Dining
35	Rajdhani	950	Gujarati,Rajasthani,North Indian	Ghatkopar West	Casual Dining
36	Wild Dining Restaurant	1200	North Indian,Continental,Mexican,Chinese	Andheri West	Casual Dining
37	Family Tree	800	Italian,Mexican,North Indian,Chinese,Salad	Thane West	Casual Dining
38	Bombay Salad Co.	900	Salad,Healthy Food,Juices	Bandra West	Casual Dining
39	Cafe Monza	1000	Italian,American,Salad,Mexican	Kharghar	Casual Dining
40	Hitchki	1200	Modern Indian,North Indian,Chinese,Momos,Birya...	Bandra	Casual Dining
41	Vedge	1000	Thai,Chinese,North Indian,Mexican,Italian,Asian	Panvel	Casual Dining
42	Joey's Pizza	800	Pizza	Malad West	Quick Bites