In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
#looding the data.

df = pd.read_excel('C:\\Users\\Sakshi\\Downloads\\resturant.xlsx'
```

In [3]:
```
df.head()
```

Out[3]:

	url	address	name	online_order	bo
0	https://www.zomato.com/bangalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	
1	https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	
3	https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	

In [4]:
```
df.columns
```

Out[4]:
```
Index(['url', 'address', 'name', 'online_order', 'book_table', 'rat
e', 'votes',
       'phone', 'location', 'rest_type', 'dish_liked', 'cuisines',
       'approx_cost(for two people)', 'reviews_list', 'menu_item',
       'listed_in(type)', 'listed_in(city)'],
      dtype='object')
```

In [5]:
```
1 len(df.columns)
```

Out[5]: 17

In [6]:
```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51717 entries, 0 to 51716
Data columns (total 17 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   url                          51717 non-null  object 
 1   address                      51717 non-null  object 
 2   name                         51717 non-null  object 
 3   online_order                 51717 non-null  object 
 4   book_table                   51717 non-null  object 
 5   rate                         43942 non-null  object 
 6   votes                        51717 non-null  int64  
 7   phone                        50509 non-null  object 
 8   location                     51696 non-null  object 
 9   rest_type                    51490 non-null  object 
 10  dish_liked                   23639 non-null  object 
 11  cuisines                     51672 non-null  object 
 12  approx_cost(for two people)  51371 non-null  float64
 13  reviews_list                 51717 non-null  object 
 14  menu_item                    51717 non-null  object 
 15  listed_in(type)              51717 non-null  object 
 16  listed_in(city)              51717 non-null  object 
dtypes: float64(1), int64(1), object(15)
memory usage: 6.7+ MB
```

In [7]:
```
1 df.drop(['url','address','phone','reviews_list','menu_item','list
```

In [8]:
```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51717 entries, 0 to 51716
Data columns (total 11 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   name                         51717 non-null  object 
 1   online_order                 51717 non-null  object 
 2   book_table                   51717 non-null  object 
 3   rate                         43942 non-null  object 
 4   votes                        51717 non-null  int64  
 5   location                     51696 non-null  object 
 6   rest_type                    51490 non-null  object 
 7   dish_liked                   23639 non-null  object 
 8   cuisines                     51672 non-null  object 
 9   approx_cost(for two people)  51371 non-null  float64
 10  listed_in(type)              51717 non-null  object 
dtypes: float64(1), int64(1), object(9)
memory usage: 4.3+ MB
```

In [9]:
```
pd.set_option('display.max_rows',None)
```

In [10]:
```
df['name']
```

Out[10]:
```
0                                                    Jalsa
1                                           Spice Elephant
2                                          San Churro Cafe
3                                  Addhuri Udupi Bhojana
4                                           Grand Village
5                                          Timepass Dinner
6          Rosewood International Hotel - Bar & Restaurant
7                                                  Onesta
8                                          Penthouse Cafe
9                                              Smacznego
10      CafÃ‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â‚Ã‚Â© Down The A...
11                                         Cafe Shuffle
12                                     The Coffee Shack
13                                           Caf-Eleven
14                                         San Churro Cafe
15                                         Cafe Vivacity
16                                         Catch-up-ino
17                                     Kirthi's Biryani
18                                              T3H Cafe
```

In [11]:
```
df['name'].iloc[10]
```

Out[11]:
```
'CafÃ\x83Â\x83Ã\x82Â\x83Ã\x83Â\x82Ã\x82Â\x83Ã\x83Â\x83Ã\x82Â\x82Ã\x83Â\x82Ã\x82Â© Down The Alley'
```

In [12]:
```
"CafÃ\x83Â\x83Ã\x82Â\x83Ã\x83Â\x82Ã\x82Â\x83Ã\x83Â\x83Ã\x82Â\x82Ã
"E2 - EntrÃ•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â© Envoy"
"Cocoa 'n' CrÃ•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â¨me"



"Cafe Down the Allaey"
```

Out[12]:
```
'Cafe Down the Allaey'
```

In [13]:
```
#type('a')
isinstance(123456,str)
```

Out[13]:
```
False
```

In [14]:
```
import re
def clean_name(inp_name):
    if isinstance(inp_name,str):
        list_name = re.findall(r'[a-zA-z0-9]',inp_name)
        string_name = ''.join(list_name)
        return string_name
    else:
        return inp_name
```

In [15]:
```
clean_name("E2 - EntrÃ•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â•Ã•Â© Envoy")
```

Out[15]: 'E2EntrEnvoy'

In [16]:
```
df['name'] = df['name'].apply(clean_name)
```

In [17]:
```
df.head()
```

Out[17]:

	name	online_order	book_table	rate	votes	location	rest_type
0	Jalsa	Yes	Yes	4.1/5	775	Banashankari	Casual Dining
1	SpiceElephant	Yes	No	4.1/5	787	Banashankari	Casual Dining
2	SanChurroCafe	Yes	No	3.8/5	918	Banashankari	Cafe, Casual Dining
3	AddhuriUdupiBhojana	No	No	3.7/5	88	Banashankari	Quick Bites
4	GrandVillage	No	No	3.8/5	166	Basavanagudi	Casual Dining

In [18]:
```
df['online_order'].unique()
```

Out[18]: array(['Yes', 'No'], dtype=object)

In [19]:
```
1 df['online_order'].value_counts()
```

Out[19]:
```
online_order
Yes    30444
No     21273
Name: count, dtype: int64
```

According to the above obsevation **60%** of the resturants provide the Online order sevice. While **40%** of the resturants do not provide online order service.

In [20]:
```
1 df['book_table'].value_counts()
```

Out[20]:
```
book_table
No     45268
Yes     6449
Name: count, dtype: int64
```

In [21]:
```
1 df['book_table'].unique()
```

Out[21]:
```
array(['Yes', 'No'], dtype=object)
```

In [22]:
```
1 df['rate']
```

Out[22]:
```
0        4.1/5
1        4.1/5
2        3.8/5
3        3.7/5
4        3.8/5
5        3.8/5
6        3.6/5
7        4.6/5
8        4.0/5
9        4.2/5
10       4.1/5
11       4.2/5
12       4.2/5
13       4.0/5
14       3.8/5
15       3.8/5
16       3.9/5
17       3.8/5
18       3.9/5
```

In [23]:
```
1 df['rate'].iloc[0]
```

Out[23]:
```
'4.1/5'
```

In [24]:
```
1 float('4.1/5'.split('/')[0])
```

Out[24]:
```
4.1
```

In [25]:
```
def rate_clean(rates):
    # print(type(rates))
    if rates == "NEW" or rates == "-":
        return np.nan
    else:
        new_rate = float(str(rates).split('/')[0])
        return new_rate
```

In [26]:
```
rate_clean('4.1/5')
```

Out[26]:
```
4.1
```

In [27]:
```
df['rate'].unique()
```

Out[27]:
```
array(['4.1/5', '3.8/5', '3.7/5', '3.6/5', '4.6/5', '4.0/5', '4.2/
5',
       '3.9/5', '3.1/5', '3.0/5', '3.2/5', '3.3/5', '2.8/5', '4.4/
5',
       '4.3/5', 'NEW', '2.9/5', '3.5/5', nan, '2.6/5', '3.8 /5', '3.
4/5',
       '4.5/5', '2.5/5', '2.7/5', '4.7/5', '2.4/5', '2.2/5', '2.3/
5',
       '3.4 /5', '-', '3.6 /5', '4.8/5', '3.9 /5', '4.2 /5', '4.0 /
5',
       '4.1 /5', '3.7 /5', '3.1 /5', '2.9 /5', '3.3 /5', '2.8 /5',
       '3.5 /5', '2.7 /5', '2.5 /5', '3.2 /5', '2.6 /5', '4.5 /5',
       '4.3 /5', '4.4 /5', '4.9/5', '2.1/5', '2.0/5', '1.8/5', '4.6
/5',
       '4.9 /5', '3.0 /5', '4.8 /5', '2.3 /5', '4.7 /5', '2.4 /5',
       '2.1 /5', '2.2 /5', '2.0 /5', '1.8 /5'], dtype=object)
```

In [28]:
```
df['rate'] = df['rate'].apply(rate_clean)
```

In [29]:
```
1 df['rate']
```

Out[29]:
```
0       4.1
1       4.1
2       3.8
3       3.7
4       3.8
5       3.8
6       3.6
7       4.6
8       4.0
9       4.2
10      4.1
11      4.2
12      4.2
13      4.0
14      3.8
15      3.8
16      3.9
17      3.8
18      3.9
```

In [30]:
```
1 df['rate'].mean()
```

Out[30]:
```
3.700448817952718
```

In [31]:

```
1 df['rest_type'].value_counts()
```

Out[31]:

rest_type	
Quick Bites	19132
Casual Dining	10330
Cafe	3732
Delivery	2604
Dessert Parlor	2263
Takeaway, Delivery	2037
Casual Dining, Bar	1154
Bakery	1141
Beverage Shop	867
Bar	697
Food Court	624
Sweet Shop	468
Bar, Casual Dining	425
Lounge	396
Pub	357
Fine Dining	346
Casual Dining, Cafe	319
Beverage Shop, Quick Bites	298
Bakery, Quick Bites	289
Mess	267
Pub, Casual Dining	255
Sweet Shop, Quick Bites	178
Kiosk	176
Dessert Parlor, Cafe	175
Cafe, Casual Dining	173
Cafe, Bakery	164
Cafe, Dessert Parlor	148
Bakery, Dessert Parlor	147
Microbrewery, Casual Dining	132
Casual Dining, Pub	127
Dessert Parlor, Quick Bites	125
Takeaway	120
Cafe, Quick Bites	111
Food Court, Quick Bites	101
Quick Bites, Beverage Shop	101
Dessert Parlor, Bakery	85
Quick Bites, Dessert Parlor	84
Food Truck	84
Pub, Microbrewery	84
Dessert Parlor, Beverage Shop	80
Beverage Shop, Dessert Parlor	79
Bakery, Cafe	78
Quick Bites, Sweet Shop	75
Quick Bites, Bakery	60
Casual Dining, Microbrewery	59
Confectionery	47
Microbrewery, Pub	42
Quick Bites, Cafe	41
Pub, Cafe	40
Pub, Bar	40
Fine Dining, Bar	40
Dhaba	40
Club	37
Lounge, Casual Dining	37
Lounge, Bar	36
Microbrewery	30
Beverage Shop, Cafe	28
Quick Bites, Food Court	19
Cafe, Bar	19
Bakery, Beverage Shop	16

```
Lounge, Cafe                    16
Lounge, Microbrewery            16
Cafe, Lounge                    15
Casual Dining, Irani Cafee      15
Fine Dining, Lounge             14
Bar, Cafe                       13
Bakery, Kiosk                   13
Microbrewery, Bar               13
Dessert Parlor, Sweet Shop      12
Casual Dining, Lounge           10
Bar, Quick Bites                 9
Microbrewery, Lounge             9
Food Court, Dessert Parlor       8
Casual Dining, Sweet Shop        8
Bar, Lounge                      6
Fine Dining, Microbrewery        5
Food Court, Casual Dining        5
Bar, Pub                         5
Casual Dining, Quick Bites       4
Club, Casual Dining              4
Quick Bites, Mess                4
Quick Bites, Meat Shop           4
Bakery, Sweet Shop               3
Bhojanalya                       3
Mess, Quick Bites                3
Pop Up                           2
Bakery, Food Court               2
Cafe, Food Court                 2
Dessert Parlor, Kiosk            2
Food Court, Beverage Shop        2
Dessert Parlor, Food Court       2
Sweet Shop, Dessert Parlor       1
Quick Bites, Kiosk               1
Name: count, dtype: int64
```

In [32]:

```
1 len(df['rest_type'].value_counts())
```

Out[32]: 93

In [33]:

```
1 df['location'].value_counts()
```

Out[33]:

location	
BTM	5124
HSR	2523
Koramangala 5th Block	2504
JP Nagar	2235
Whitefield	2144
Indiranagar	2083
Jayanagar	1926
Marathahalli	1846
Bannerghatta Road	1630
Bellandur	1286
Electronic City	1258
Koramangala 1st Block	1238
Brigade Road	1218
Koramangala 7th Block	1181
Koramangala 6th Block	1156
Sarjapur Road	1065
Ulsoor	1023
Koramangala 4th Block	1017
MG Road	918
Banashankari	906
Kalyan Nagar	853
Richmond Road	812
Frazer Town	727
Malleshwaram	725
Basavanagudi	684
Residency Road	675
Banaswadi	664
Brookefield	658
New BEL Road	649
Kammanahalli	648
Rajajinagar	591
Church Street	569
Lavelle Road	529
Shanti Nagar	511
Shivajinagar	499
Domlur	496
Cunningham Road	491
Old Airport Road	446
Ejipura	439
Commercial Street	370
St. Marks Road	352
Koramangala 8th Block	320
Vasanth Nagar	295
Jeevan Bhima Nagar	272
Wilson Garden	246
Bommanahalli	238
Koramangala 3rd Block	216
Kumaraswamy Layout	195
Thippasandra	194
Basaveshwara Nagar	191
Nagawara	187
Seshadripuram	165
Hennur	161
Majestic	155
HBR Layout	153
Infantry Road	151
Race Course Road	139
City Market	126
Yeshwantpur	119
Varthur Main Road, Whitefield	117

```
ITPL Main Road, Whitefield        113
South Bangalore                   107
Koramangala 2nd Block             102
Hosur Road                        102
Kaggadasapura                     101
CV Raman Nagar                     90
Vijay Nagar                        80
RT Nagar                           80
Sanjay Nagar                       76
Sadashiv Nagar                     63
Sahakara Nagar                     53
Koramangala                        48
East Bangalore                     44
Jalahalli                          38
Magadi Road                        34
Rammurthy Nagar                    32
Langford Town                      27
Sankey Road                        27
Old Madras Road                    22
Mysore Road                        22
Kanakapura Road                    19
KR Puram                           18
Uttarahalli                        17
Hebbal                             14
North Bangalore                    14
Kengeri                             9
Nagarbhavi                          9
Central Bangalore                   8
West Bangalore                      6
Yelahanka                           6
Jakkur                              3
Rajarajeshwari Nagar                2
Peenya                              1
Name: count, dtype: int64
```

In [34]:
```
df[df['rate'] == 0]
```

Out[34]:

name	online_order	book_table	rate	votes	location	rest_type	dish_liked	cuisines

In [35]:
```
df.shape
```

Out[35]: (51717, 11)

Which restaurants offer online delivery services

In [36]:
```
print("Restaurants that offer online delivery services:")
print(df[df['online_order'] == 'Yes']['name'].unique())
```

```
Restaurants that offer online delivery services:
['Jalsa' 'SpiceElephant' 'SanChurroCafe' ... 'BaoGrills' 'QuickBites'
 'ShrawanRestaurant']
```

Which locations have the highest-rated restaurants?

In [37]:
```
location_counts = df['location'].value_counts()
```

In [38]:
```
print("Locations with the Most Restaurants:")
print(location_counts.head(10))
```

```
Locations with the Most Restaurants:
location
BTM                    5124
HSR                    2523
Koramangala 5th Block  2504
JP Nagar               2235
Whitefield             2144
Indiranagar            2083
Jayanagar              1926
Marathahalli           1846
Bannerghatta Road      1630
Bellandur              1286
Name: count, dtype: int64
```

What are top 10 highest-rated restaurantsz?

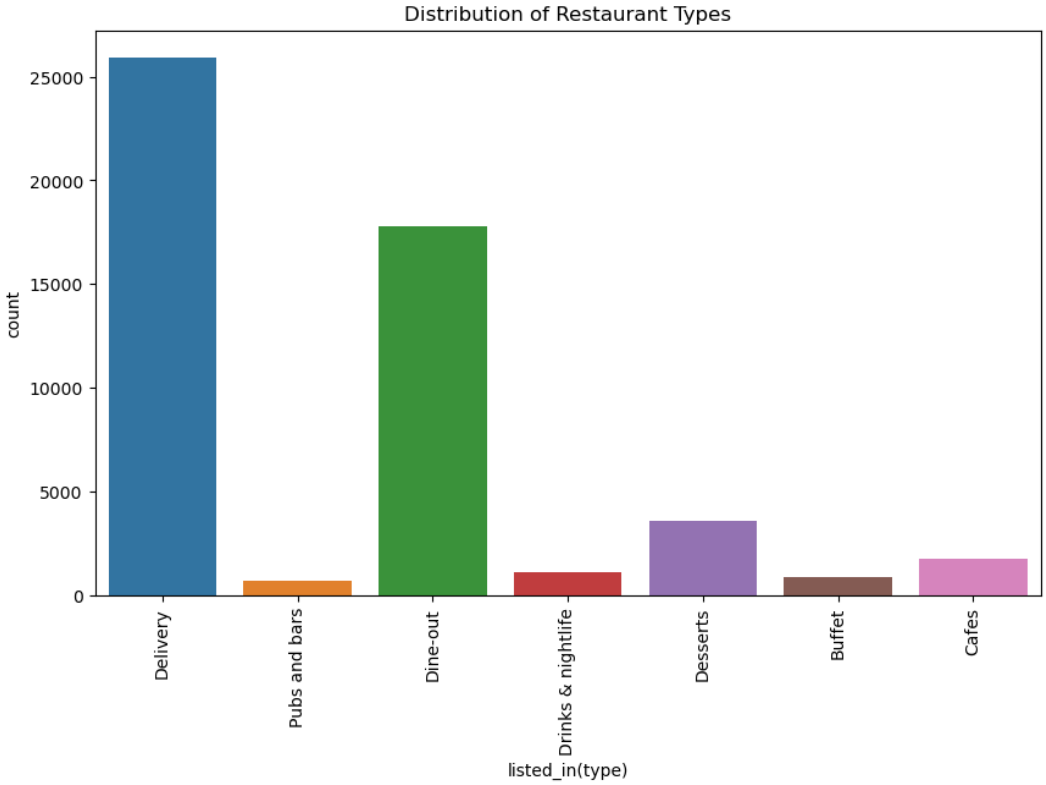
In [39]:
```
df = df.sort_values(by='rate', ascending=False)
top_10_rated = df.head(10)
print(top_10_rated[['name', 'rate']])
```

```
                            name  rate
35082  AsiaKitchenByMainlandChina   4.9
37613  AsiaKitchenByMainlandChina   4.9
10879  AsiaKitchenByMainlandChina   4.9
17877       BelgianWaffleFactory    4.9
4944     BygBrewskiBrewingCompany   4.9
47987       BelgianWaffleFactory    4.9
51042                   Flechazo   4.9
39559                PunjabGrill   4.9
43055       BelgianWaffleFactory    4.9
37099       ABsAbsoluteBarbecues    4.9
```

What is type of restaurant in listed_in(type)?

In [40]:

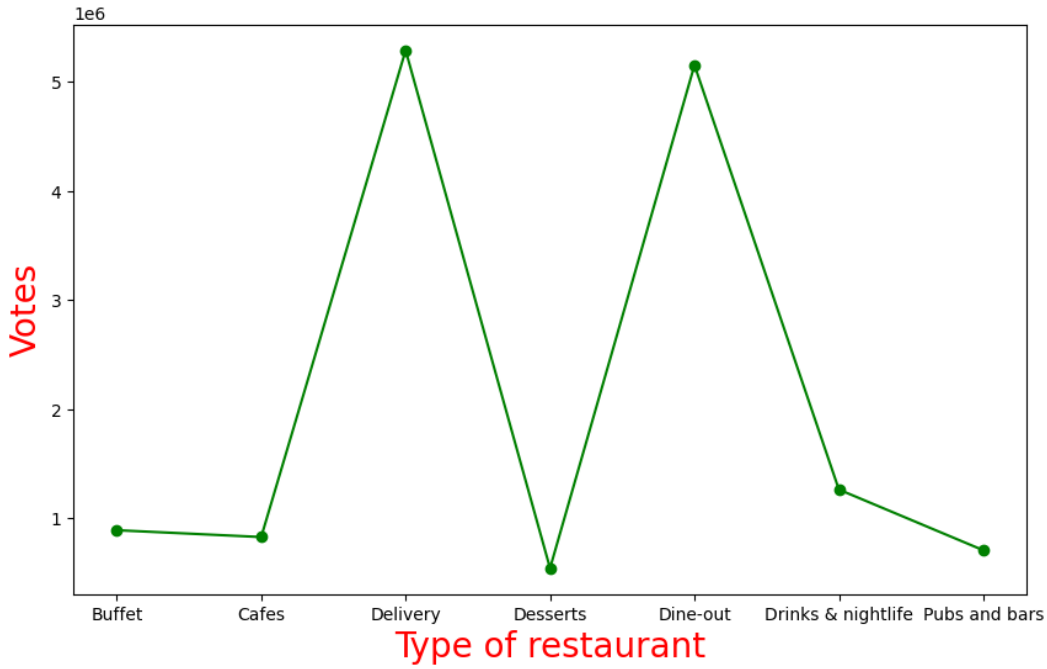
```
plt.figure(figsize=(10, 6))
sns.countplot(x='listed_in(type)',data=df)
plt.xticks(rotation=90)
plt.title('Distribution of Restaurant Types')
plt.show()
```



Delivery are majority fall into type of restaurants category.

In [41]:
```
grouped_data = df.groupby('listed_in(type)')['votes'].sum()
result = pd.DataFrame({'votes': grouped_data})
plt.figure(figsize=(10, 6))
plt.plot(result, c="green", marker="o")
plt.xlabel("Type of restaurant", c="red", size=20)
plt.ylabel("Votes", c="red", size=20)
```

Out[41]: Text(0, 0.5, 'Votes')

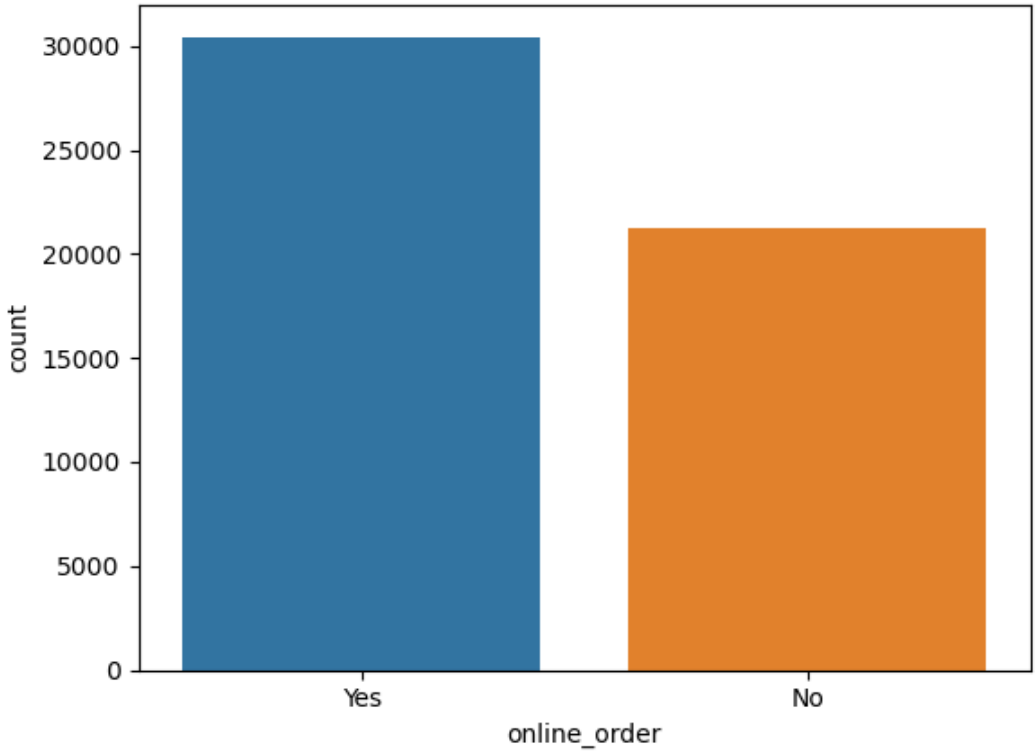


Delivery are preferred by a larger number of individuals persons.

Does restaurant accept online_order?

In [42]:
```
1 sns.countplot(x=df['online_order'])
```

Out[42]: <Axes: xlabel='online_order', ylabel='count'>



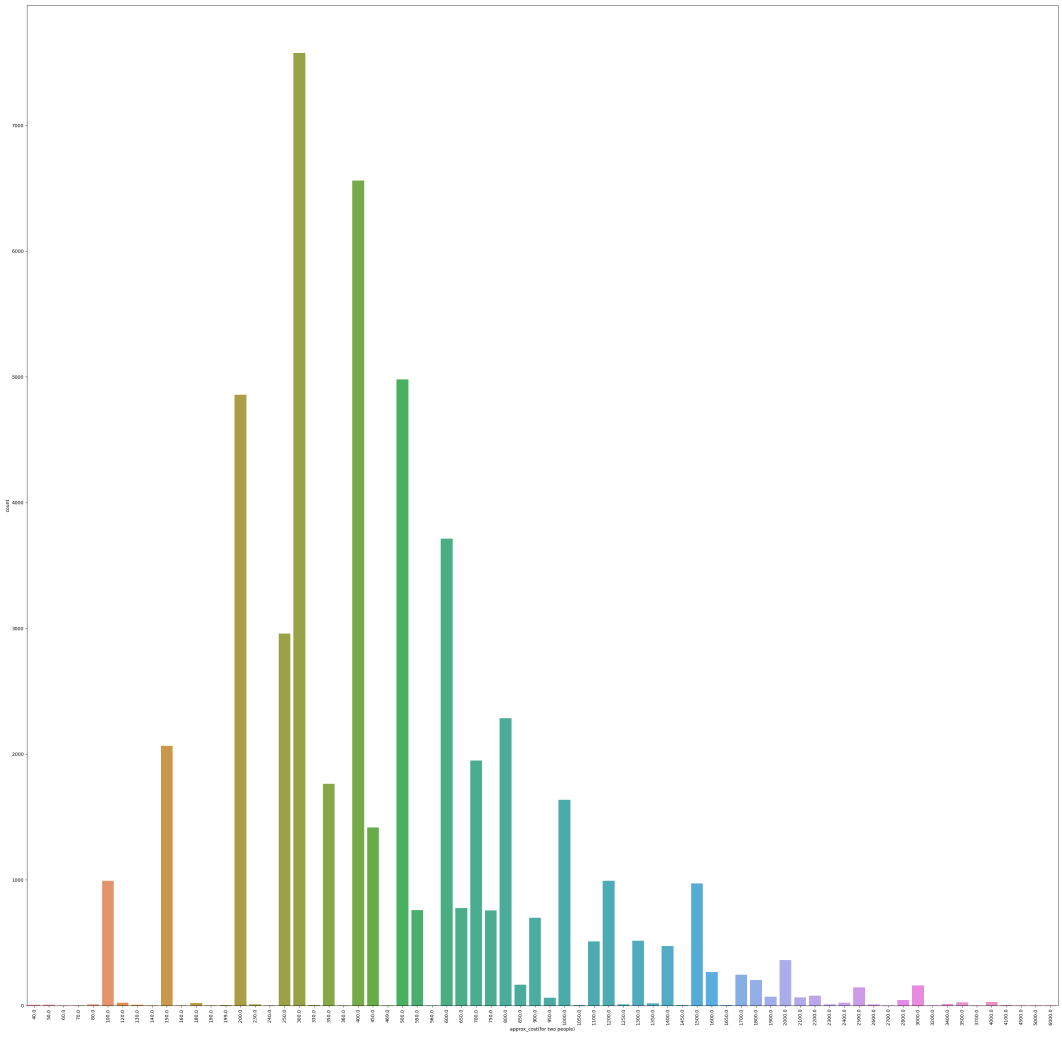
The most of the restaurants accept online orders.

Which restaurant customer prefer most?

In [43]:
```
customer_data=df['approx_cost(for two people)']
plt.figure(figsize=(40, 39))
plt.xticks(rotation=90)
sns.countplot(x=customer_data)

```

Out[43]: <Axes: xlabel='approx_cost(for two people)', ylabel='count'>



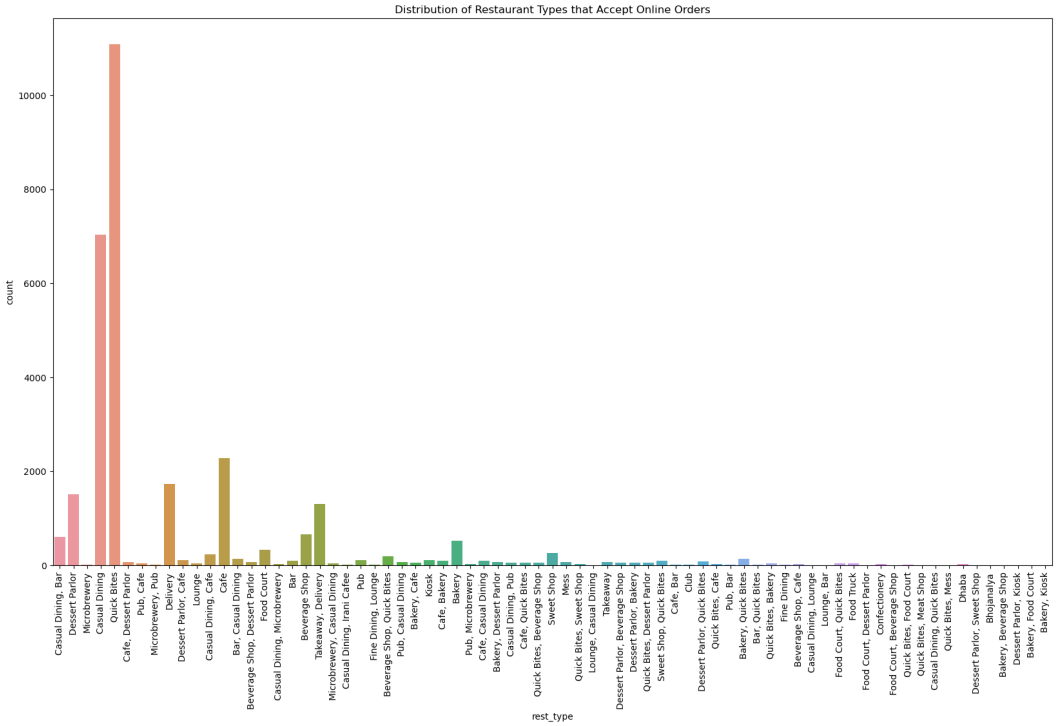
The most of customer prefer restaurants with an approximate cost of **300 rupees.**

Which is restaurant accept online_order?

In [44]:

```
online_order =df[df['online_order'] == 'Yes']

plt.figure(figsize=(20,11))
sns.countplot(x='rest_type', data=online_order)
plt.xticks(rotation=90)
plt.title('Distribution of Restaurant Types that Accept Online Or
plt.show()
```



Quick Bites Restaurants mostly accept online_order

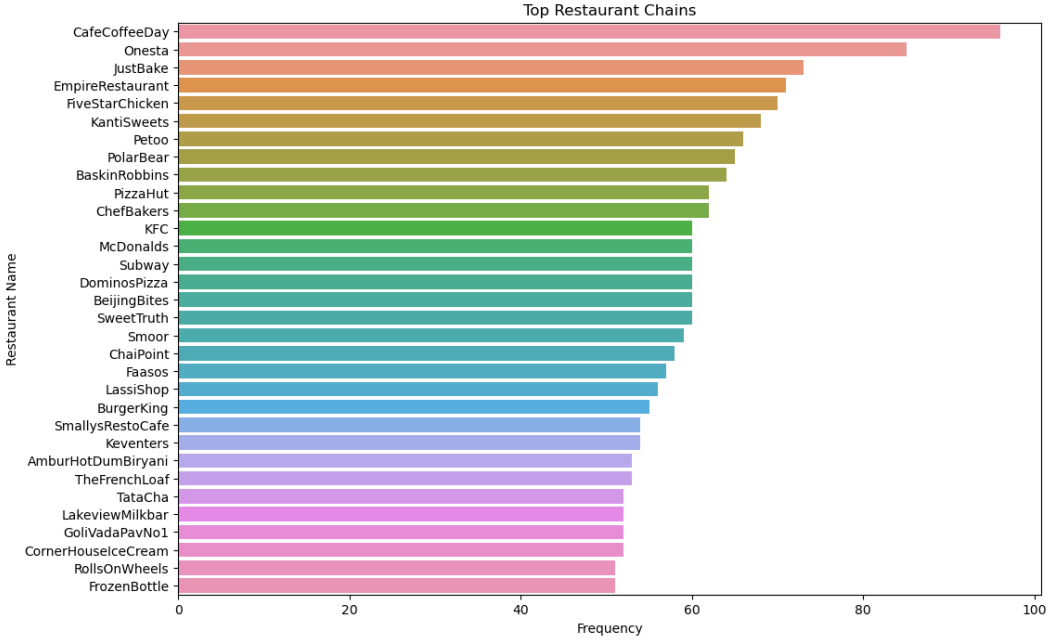
Top restaurants chains?

In [45]:

```
name_counts = df['name'].value_counts()
name_counts = name_counts[name_counts >50]

plt.figure(figsize=(12,8))
sns.barplot(x=name_counts.values,y=name_counts.index)
plt.xlabel('Frequency')
plt.ylabel('Restaurant Name')
plt.title('Top Restaurant Chains')

plt.show()
```



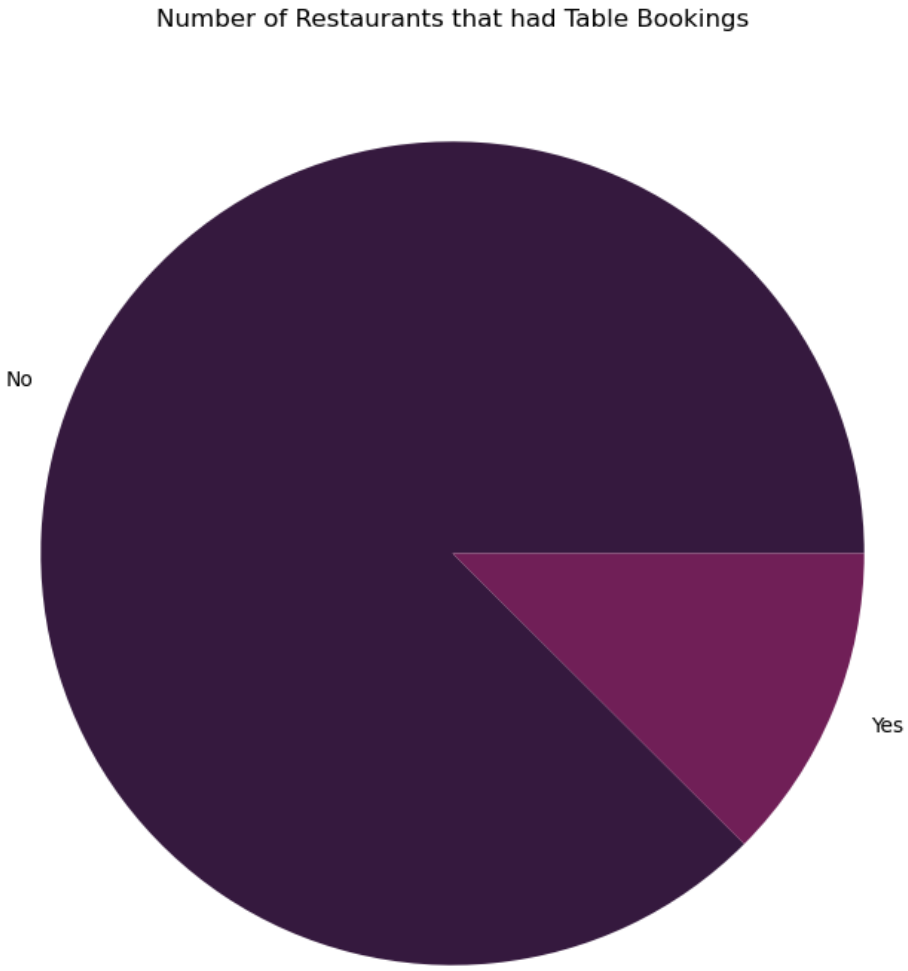
Cafe Coffee Day is top restaurants chains

How many restaurants had table bookings?

In [46]:

```
table_counts = df['book_table'].value_counts()
plt.figure(figsize=(10,9))
plt.pie(table_counts.values,labels=table_counts.index,colors=sns.
plt.title('Number of Restaurants that had Table Bookings')

plt.show()
```



Most restaurants do not have table booking facility.

Which restaurants had the highest voting?(means counts of rating given)

In [47]:
```
1 df.head()
```

Out[47]:

	name	online_order	book_table	rate	votes	location
35082	AsiaKitchenByMainlandChina	Yes	Yes	4.9	2249	Koramangala 5th Block
37613	AsiaKitchenByMainlandChina	Yes	Yes	4.9	2256	Koramangala 5th Block
10879	AsiaKitchenByMainlandChina	Yes	Yes	4.9	2178	Koramangala 5th Block
17877	BelgianWaffleFactory	Yes	No	4.9	1735	Brigade Road
4944	BygBrewskiBrewingCompany	Yes	Yes	4.9	16345	Sarjapur Road

In [48]:
```
1 voting = df.groupby('name')[['votes']].mean()
2 voting
```

Out[48]:

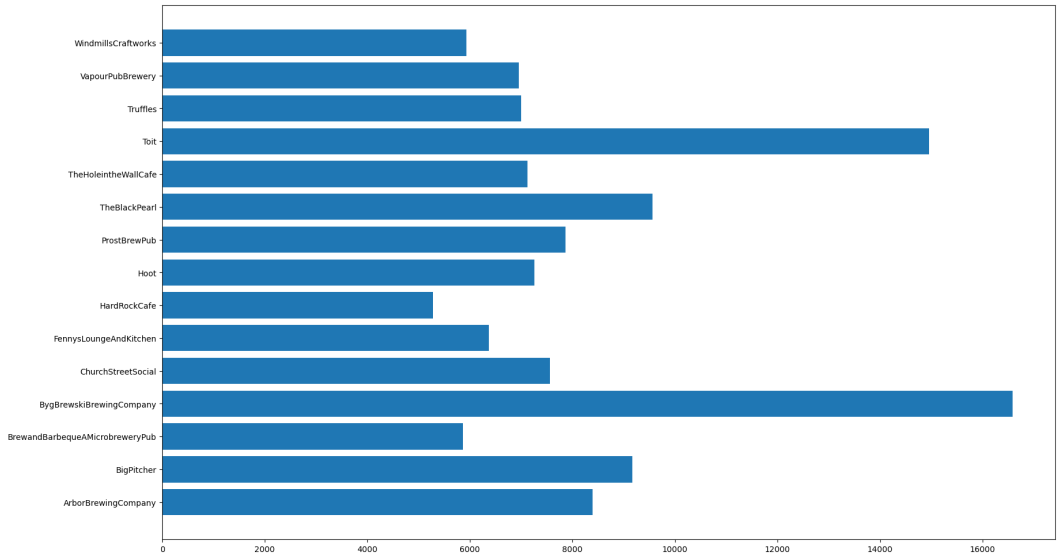
	votes
name	
1947	1470.714286
1000BC	49.000000
100C	41.000000
1131BarKitchen	2823.000000
11to11ExpressBiriyanis	22.000000
12thMainGrandMercure	354.100000
1441Pizzeria	135.500000
1522ThePub	1646.880000
154BreakfastClub	1514.125000
1722UrbanBistro	222.090909
18IceCafe	24.166667

In [49]:
```
1 voting['votes'].describe()
```

Out[49]:
```
count     8723.000000
mean       162.866843
std        551.374445
min          0.000000
25%          1.581169
50%         21.000000
75%        103.879310
max      16588.500000
Name: votes, dtype: float64
```

In [50]:
```
1 high_vot = voting[voting['votes'] >5000]
```

In [51]:
```
1 plt.figure(figsize=(20,12),dpi=100)
2 plt.barh(high_vot.index,high_vot['votes'])
3
4 plt.show()
```



Byg Brewski Brewing Company restaurant had the highest voting.

In [52]:
```
1 df.shape
```

Out[52]:
```
(51717, 11)
```

In [ ]:
```
1
```