In [47]:

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

In [62]:

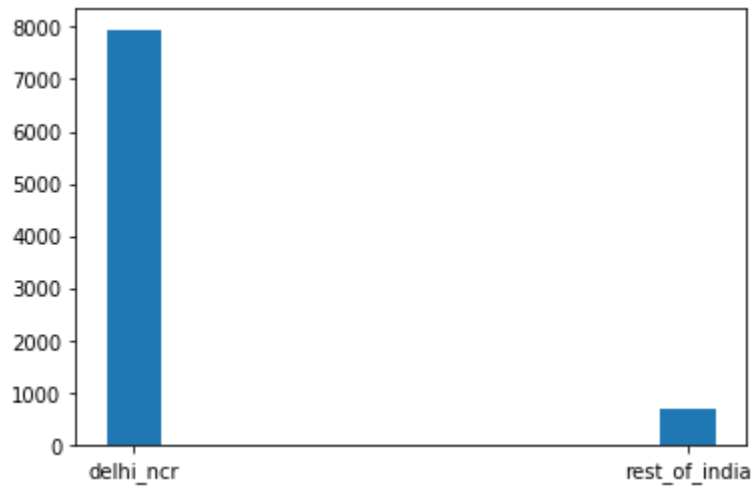
```
data=pd.read_csv('f:/data/zomato.csv',skipinitialspace=True,encoding = "ISO-8859-1")
```

Plot the bar graph of number of restaurants present in Delhi NCR vs Rest of India

In [50]:

```
#as we know that country code of india is 1
df=data[data['Country Code']==1]
df.head()
df.City.unique()
NCR_cities=['New Delhi','Ghaziabad','Faridabad','Gurgaon',"Noida"]
def cities(city):
    y=0
    if city in NCR_cities:
        y=1
    else:
        y=0
    return y
df["ncr/rest_india"]=df['City'].apply(cities)
delhi_ncr=df["ncr/rest_india"].value_counts().iloc[0]
rest_india=df["ncr/rest_india"].value_counts().iloc[1]
print("restuarant in Delhi NCR :",delhi_ncr)
print("restuarant in rest of india :",rest_india)

plt.bar(["delhi_ncr",'rest_of_india'],[delhi_ncr,rest_india],width=0.1,align='center')
plt.show()
```

```
restuarant in Delhi NCR : 7947
restuarant in rest of india : 705

C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
  del sys.path[0]
```



Find the cuisines which are not present in restaurant of Delhi NCR but present in rest of India.Check using Zomato API whether this cuisines are actually not served in restaurants of Delhi-NCR or just it due to incomplete dataset

In [51]:

```
df=df[['Cuisines','ncr/rest_india']]
df.dropna(inplace=True)
Cuisines_rest=[]
Cuisines_ncr=[]
for i in df.values:
    if i[1]==0:
        for j in i[0].split(','):
            if j.strip() not in Cuisines_rest:
                Cuisines_rest.append(j.strip())
    else:
        for j in i[0].split(','):
            if j.strip() not in Cuisines_ncr:
                Cuisines_ncr.append(j.strip())
print("total number of Cuisines in Delhi NCR : ",len(Cuisines_ncr))
print("total number of Cuisines in Rest od India : ",len(Cuisines_rest))
print()
diff_Cuisines=[]
for i in Cuisines_rest:
    if i not in Cuisines_ncr:
        diff_Cuisines.append(i)
print("List of Cusines not served in Delhi NcR :")
for i in diff_Cuisines:
    print(i,end=',')
print()
print()
print("As we can see BBQ is present in these list of cuisine which are not served in delhi
print("Name of restaurant : ","Barbeque Nation")
print("Aderess : ","2nd Floor, Munshilal Building, Block N, Outer Circle, Connaught Place,
print("So we can say that BBQ cuisine is served in Delhi NCR but data Set was incomplete")
```

```
total number of Cuisines in Delhi NCR :  86
total number of Cuisines in Rest od India :  70

List of Cusines not served in Delhi NcR :
German,Malwani,BBQ,Cajun,

As we can see BBQ is present in these list of cuisine which are not served i
n delhi But Cuisine of BBQ is present in delhi,we can verify it through zoma
to apis
Name of restaurant :  Barbeque Nation
Aderess :  2nd Floor, Munshilal Building, Block N, Outer Circle, Connaught P
lace, New Delhi
So we can say that BBQ cuisine is served in Delhi NCR but data Set was incom
plete
```

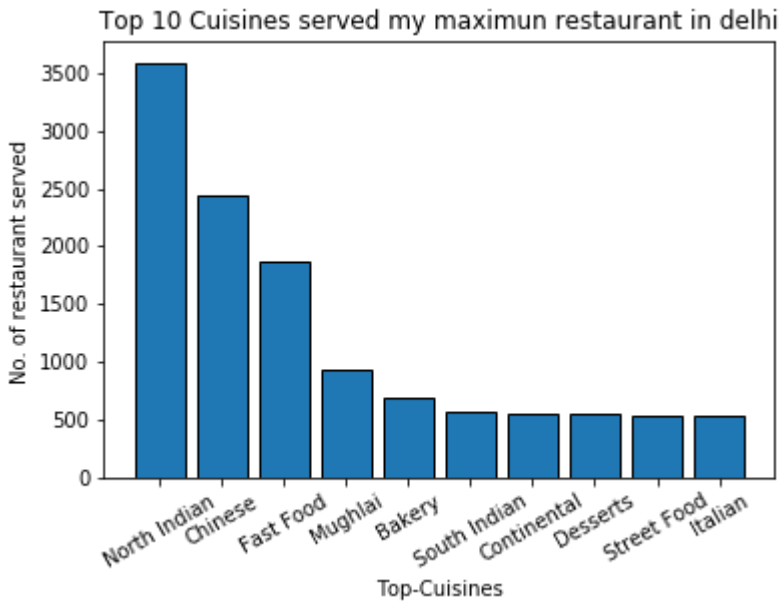
Find the top 10 cuisines served by maximum number of restaurants in Delhi NCR and rest of India

In [52]:

```
d_ncr={}
d_rest={}
all_={}
for i in df.values:
    if i[1]==0:
        for j in i[0].split(','):
            d_rest[j.strip()]=d_rest.get(j.strip(),0)+1
            all_[j.strip()]=all_.get(j.strip(),0)+1
    else:
        for j in i[0].split(','):
            d_ncr[j.strip()]=d_ncr.get(j.strip(),0)+1
            all_[j.strip()]=all_.get(j.strip(),0)+1

l=sorted(d_ncr.values())
l.reverse()
values=[]
cuisine=[]
for i in range(10):
    values.append(l[i])
    for j in d_ncr:
        if d_ncr[j]==l[i]:
            cuisine.append(j)
print("IN DELHI NCR")
for i in range(10):
    print(cuisine[i],values[i])
plt.bar(cuisine,values,edgecolor="black")
plt.title("Top 10 Cuisines served my maximun restaurant in delhi")
plt.xlabel("Top-Cuisines")
plt.ylabel("No. of restaurant served ")
plt.xticks(rotation=30)
plt.show()


l=sorted(d_rest.values())
l.reverse()
values=[]
cuisine=[]
for i in range(10):
    values.append(l[i])
    for j in d_rest:
        if d_rest[j]==l[i]:
            cuisine.append(j)
print("IN REST OF INDIA")
for i in range(10):
    print(cuisine[i],values[i])

plt.bar(cuisine,values,edgecolor="black")
plt.title("Top 10 Cuisines served my maximun restaurant in Rest of India")
plt.xlabel("Top-Cuisines")
plt.ylabel("No. of restaurant served ")
plt.xticks(rotation=30)
plt.show()

l=sorted(all_.values())
l.reverse()
values=[]
cuisine=[]
for i in range(10):
```

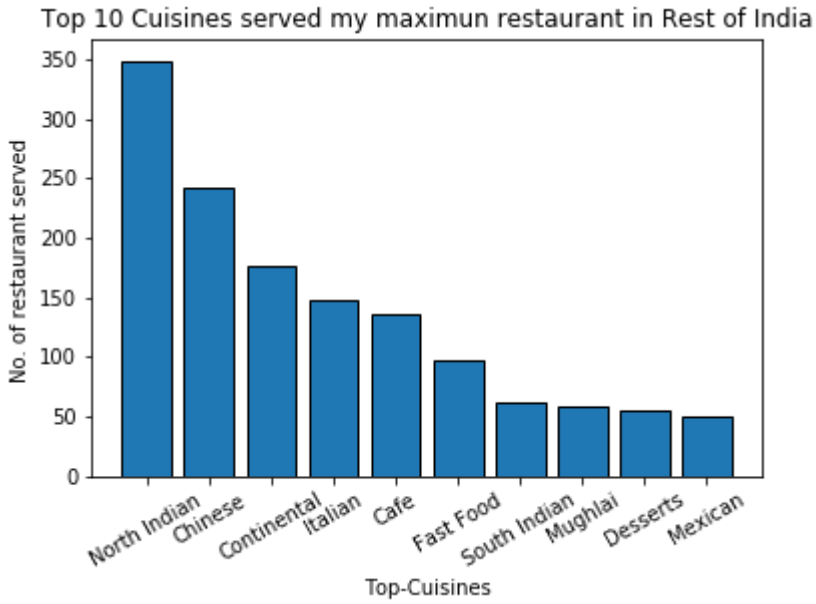
```
    values.append(l[i])
    for j in all_:
        if all_[j]==l[i]:
            cuisine.append(j)
print("IN INDIA")
for i in range(10):
    print(cuisine[i],values[i])

plt.bar(cuisine,values,edgecolor="black")
plt.title("Top 10 Cuisines served my maximun restaurant in India")
plt.xlabel("Top-Cuisines")
plt.ylabel("No. of restaurant served ")
plt.xticks(rotation=30)
plt.show()
```

```
IN DELHI NCR
North Indian 3597
Chinese 2448
Fast Food 1866
Mughlai 933
Bakery 697
South Indian 569
Continental 547
Desserts 542
Street Food 538
Italian 535
```

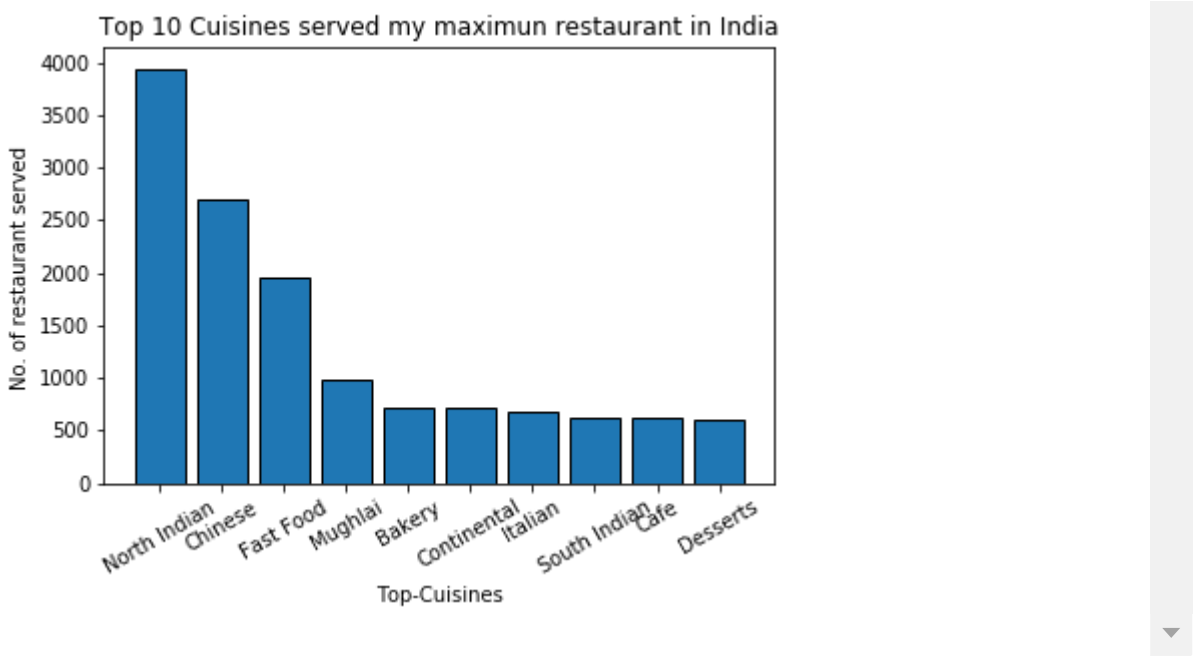



```
IN REST OF INDIA
North Indian 349
Chinese 242
Continental 177
Italian 147
```

```
Cafe 136
Fast Food 97
South Indian 62
Mughlai 59
Desserts 55
Mexican 50
```




```
IN INDIA
North Indian 3946
Chinese 2690
Fast Food 1963
Mughlai 992
Bakery 726
Continental 724
Italian 682
South Indian 631
Cafe 627
Desserts 597
```

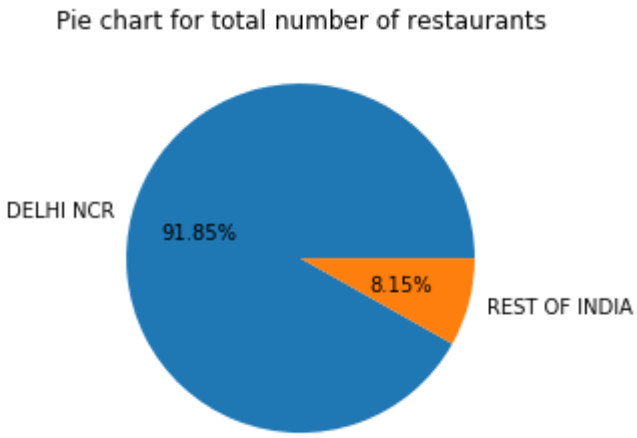


Write a short detailed analysis of how cuisine served is different from Delhi NCR to Rest of India. Plot the suitable graph to explain your inference.

In [56]:

```
print('from the given data set we can see that maximum no. of restaurant are present at del
plt.pie([7947,705],labels=["DELHI NCR",'REST OF INDIA'],autopct="%.2f%%")
plt.title("Pie chart for total number of restaurants")
plt.show()

print("In Indian different variety of cuisines are made In delhi ncr we have 86 different v
print("we can say that we can find all type of cuisines in delhi ncr which are present in i
print("famaous cuisines in india are : ")
for i in cuisine:
    print(i,end=',')
```

```
from the given data set we can see that maximum no. of restaurant are presen
t at delhi ncr i.e 7947 retaurants and rest are from differents parts.we can
plot pie chat between them to show percentage of restaurant presents in diff
erent parts
```

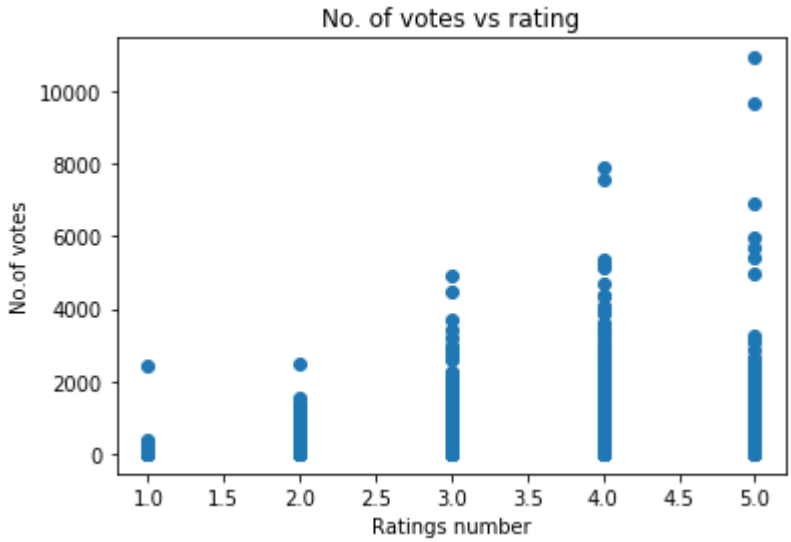



```
In Indian different variety of cuisines are made In delhi ncr we have 86 dif
ferent variety of cuisines
we can say that we can find all type of cuisines in delhi ncr which are pres
ent in india ,but due to incomplete data set we get 4 cuisines which are not
present in delhi
famaous cuisines in india are :
North Indian,Chinese,Fast Food,Mughlai,Bakery,Continental,Italian,South Indi
an,Cafe,Desserts,
```

Number of Votes given Restaurant

In [57]:

```
df=data[["Votes","Aggregate rating","Rating color"]]

#let white Represent 1 and rating below 1.5
#let Red represent 2 and rating between 1.5 to 2.5
#let Orange represent 3 and rating between 2.5 to 3.5
#let Yellow represent 4 and rating between 3.5 to 4.0
#let Green represent 5 and rating between 4.0 to 4.5
#let Dark Green represent 6 and rating between above 4.5

x=[1,2,3,4,5]
color=['Red', 'Orange', 'Yellow', 'Green', 'Dark Green']
x_axis=[]
y_axis=[]
for i in df.values:
    if i[2] !="White":
        x_axis.append(x[color.index(i[2])])
        y_axis.append(i[0])
plt.scatter(x_axis,y_axis)
plt.xlabel("Ratings number")
plt.ylabel("No.of votes")
plt.title("No. of votes vs rating")
plt.show()

print("After seeing these graph we can see that as no. of votes increases ratings also incr
print("For Rating Color equal to red i.e 1 we can see that maximum votes are less than 2000
print("For Rating Color equal to Orange i.e 2 we can see that maximum votes are near about
print("For Rating Color equal to Yellow i.e 3 we can see that maximum votes are near about
print("And so on")
```



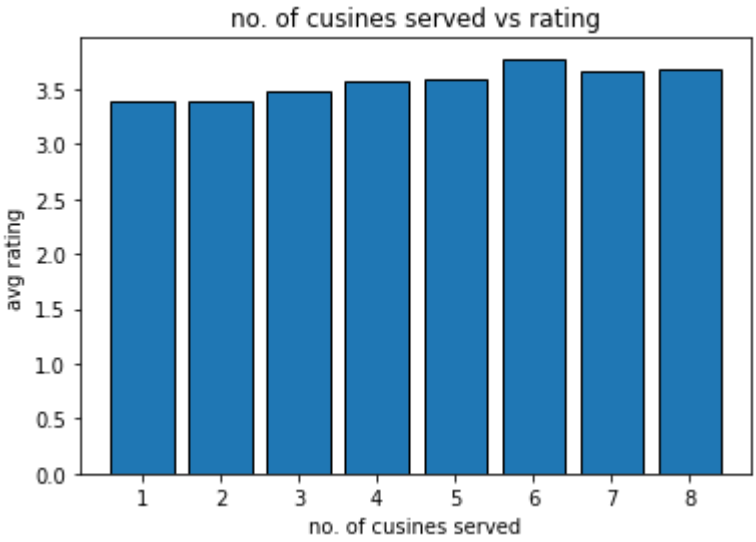
```
After seeing these graph we can see that as no. of votes increases ratings a
lso increases
For Rating Color equal to red i.e 1 we can see that maximum votes are less t
han 2000
For Rating Color equal to Orange i.e 2 we can see that maximum votes are nea
r about 2000
For Rating Color equal to Yellow i.e 3 we can see that maximum votes are nea
r about 4000
And so on
```

Restaurant serving more number of cuisines.

In [58]:

```
df=data[["Cuisines","Aggregate rating","Rating color"]]
df.dropna(inplace=True)
x=[1,2,3,4,5]
color=['Red', 'Orange', 'Yellow', 'Green', 'Dark Green']
x_axis=[]
y_axis=[]
d={}
d1={}
for i in df.values:
    if i[2] !="White":
        d[len(i[0].split(","))]=d.get(len(i[0].split(",")),0)+1
        d1[len(i[0].split(","))]=d1.get(len(i[0].split(",")),0)+i[1]
l1=np.array(list(d.values()))
l2=np.array(list(d1.values()))
a=l2/l1
y_axis=[float(format(i,".2f")) for i in a]
x_axis=list(d.keys())
plt.bar(x_axis,y_axis,edgecolor="Black")
plt.xlabel("no. of cusines served")
plt.ylabel("avg rating")
plt.title("no. of cusines served vs rating")

print("After seeing graph we can say that as no. of cuisines increases than rating also inc
```

```
After seeing graph we can say that as no. of cuisines increases than rating
also increases

C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
```

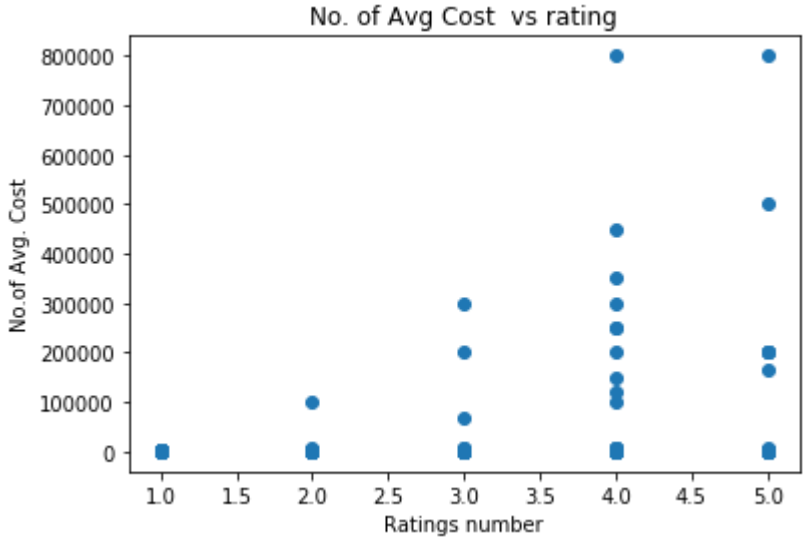


Average Cost of Restaurant

In [59]:

```
df=data[["Average Cost for two","Aggregate rating","Rating color"]]
df.dropna(inplace=True)
x=[1,2,3,4,5]
color=['Red', 'Orange', 'Yellow', 'Green', 'Dark Green']
x_axis=[]
y_axis=[]
for i in df.values:
    if i[2] !="White":
        x_axis.append(x[color.index(i[2])])
        y_axis.append(i[0])
plt.scatter(x_axis,y_axis)
plt.xlabel("Ratings number")
plt.ylabel("No.of Avg. Cost")
plt.title("No. of Avg Cost  vs rating")
plt.show()
```

```
C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
```

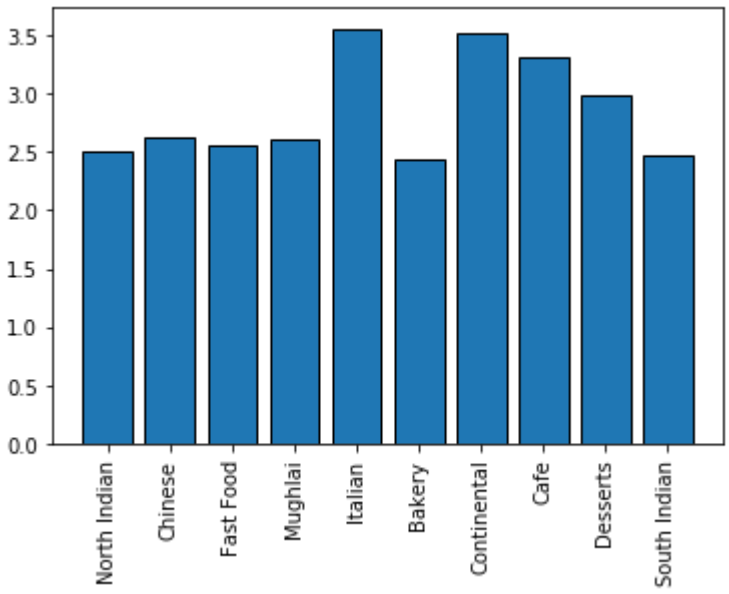


Restaurant serving some specific cuisines.

In [60]:

```
df=data[["Cuisines","Aggregate rating"]]
df.dropna(inplace=True)
d={}
d1={}
x_axis=[]
y_axis=[]
for i in df.values:
    for j in i[0].split(","):
        d[j.strip()]=d.get(j.strip(),0)+1
        d1[j.strip()]=d1.get(j.strip(),0)+i[1]

l=sorted(d.values())
l.reverse()
for i in l[:10]:
    for j in d:
        if d[j]==i:
            x_axis.append(j)
            y_axis.append(float(format(d1[j]/d[j],".2f")))
plt.bar(x_axis,y_axis,edgecolor="black")
plt.xticks(rotation=90)
plt.show()
print("from the bar grahp we can see that Italian ,Continental food are liked most")
```

```
C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
```



from the bar grahp we can see that Italian ,Continental food are liked most

Weighted Restaurant Rating=Σ (number of votes * rating) / Σ (number of votes) .

In [61]:

```
df=data
df=df[["Locality",'Aggregate rating','Votes']]
df.dropna(inplace=True)
df1=df.groupby('Locality')
localities={}
for i ,j in df1:
    t_votes=j['Votes'].sum()
    a=0
    if t_votes>0:
        for k in j.values:
            a+=int(int(k[2]))*float(k[1])
        localities[i]=format((a)/t_votes,".5f")
l=sorted(localities.values())
l.reverse()
for i in range(10):
    for j in localities:
        if localities[j]==l[i]:
            print(j," : ",l[i])
            localities[j]=-1
            break
print()
print("Only for INDIA")
print()
df=data[data["Country Code"]==1]
df=df[["Locality",'Aggregate rating','Votes']]
df.dropna(inplace=True)
df1=df.groupby('Locality')
localities={}
for i ,j in df1:
    t_votes=j['Votes'].sum()
    a=0
    if t_votes>0:
        for k in j.values:
            a+=int(int(k[2]))*float(k[1])
        localities[i]=format((a)/t_votes,".5f")
l=sorted(localities.values())
l.reverse()
for i in range(10):
    for j in localities:
        if localities[j]==l[i]:
            print(j," : ",l[i])
            localities[j]=-1
            break
```

```
C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
  This is separate from the ipykernel package so we can avoid doing imports
 until

Aminabad  :  4.90000
Barwa Towers, Al Sadd  :  4.90000
```

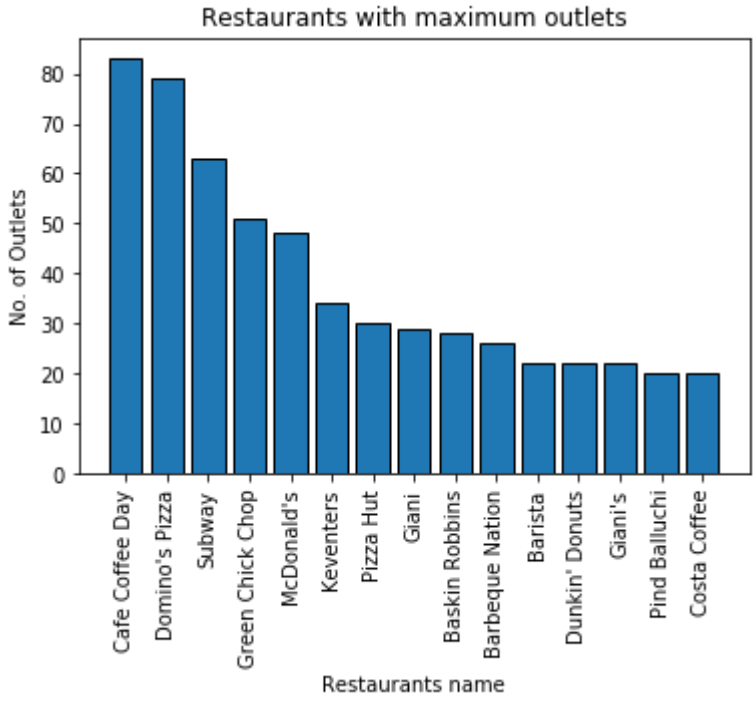
```
Beak Street, Soho  :  4.90000
Bebek  :  4.90000
Bishopsgate, City Of London  :  4.90000
Caddebostan  :  4.90000
Cengkareng  :  4.90000
City and Suburban  :  4.90000
DIFC  :  4.90000
Deira City Centre Area  :  4.90000

Only for INDIA

Aminabad  :  4.90000
Hotel Clarks Amer, Malviya Nagar  :  4.90000
Friends Colony  :  4.88692
Powai  :  4.84187
Kirlampudi Layout  :  4.82016
Deccan Gymkhana  :  4.80000
Express Avenue Mall,  Royapettah  :  4.80000
Banjara Hills  :  4.71876
Sector 5, Salt Lake  :  4.70702
Riverside Mall, Gomti Nagar  :  4.70000
```

Visualization

Plot the bar graph top 15 restaurants have a maximum number of outlets.

In [327]:

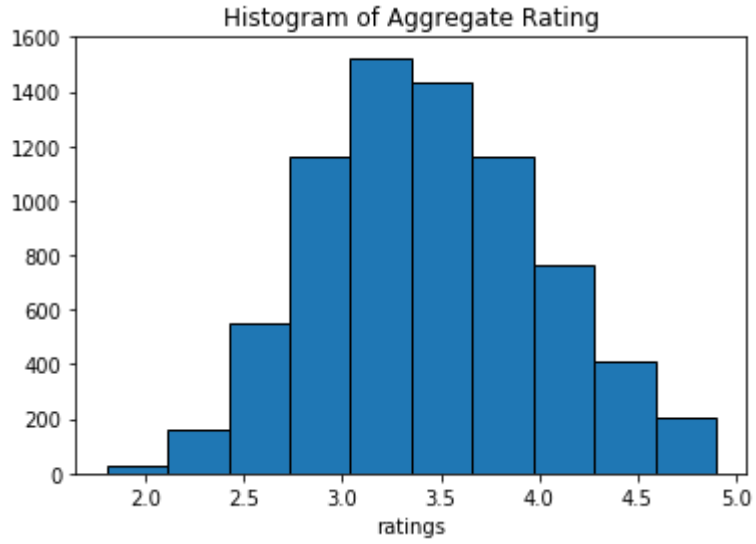
```
df=data[["Restaurant Name"]]
l=[]
l1=[]
for i in df["Restaurant Name"].value_counts().head(15).index:
    l.append(i)
for i in df["Restaurant Name"].value_counts().head(15).values:
    l1.append(i)
plt.bar(l,l1,edgecolor="black")
plt.xticks(rotation=90)
plt.title("Restaurants with maximum outlets")
plt.xlabel("Restaurants name")
plt.ylabel("No. of Outlets")
plt.show()
```



Plot the histogram of aggregate rating of restaurant(drop the unrated restaurant)

In [43]:

```
df=data["Aggregate rating"]
df.replace(0,np.nan,inplace=True)
df.dropna(inplace=True)
a=df.values
plt.title("Histogram of Aggregate Rating")
plt.xlabel("ratings")
plt.hist(a,edgecolor='black',bins=10)
plt.show()
```

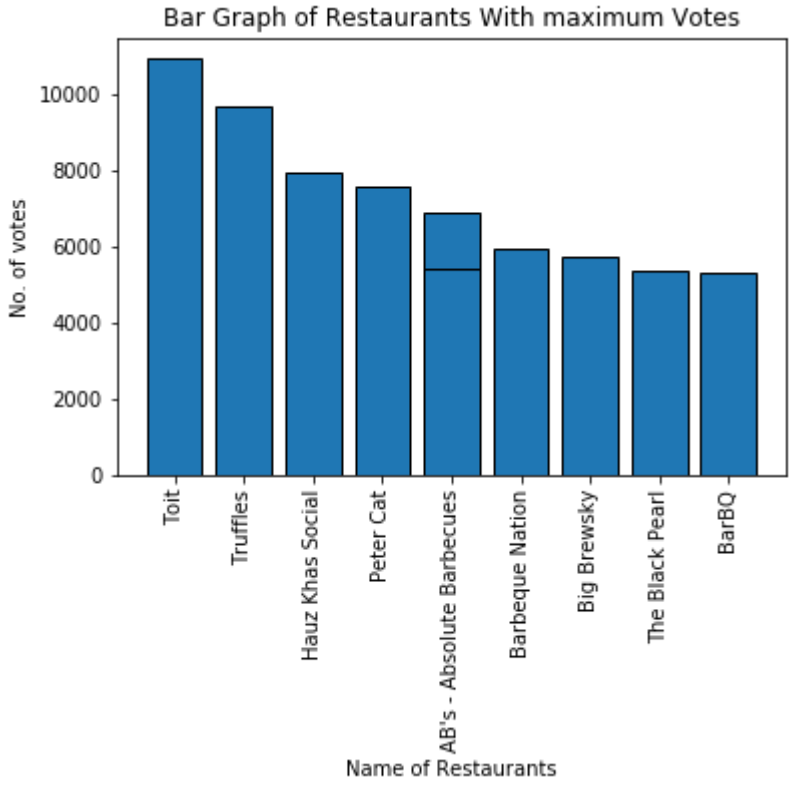


Plot the bar graph top 10 restaurants in the data with the highest number of votes

In [56]:

```
df=data[["Restaurant Name",'Votes']]
df.sort_values("Votes",inplace=True,ascending=False)
l=[]
l1=[]
for i in df.head(10).values:
    l.append(i[0])
    l1.append(i[1])
    print(i[0]," : ",i[1])
plt.title("Bar Graph of Restaurants With maximum Votes")
plt.xlabel("Name of Restaurants")
plt.ylabel("No. of votes")
plt.bar(l,l1,edgecolor='black')
plt.xticks(rotation=90)
plt.show()
```

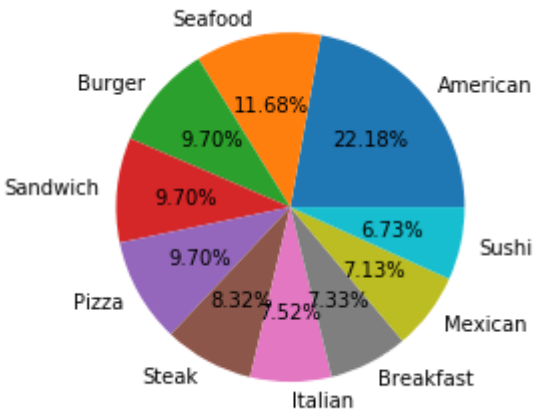
```
Toit  :  10934
Truffles  :  9667
Hauz Khas Social  :  7931
Peter Cat  :  7574
AB's - Absolute Barbecues  :  6907
Barbeque Nation  :  5966
Big Brewsky  :  5705
AB's - Absolute Barbecues  :  5434
The Black Pearl  :  5385
BarBQ  :  5288

C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
```



In [29]:

```
df=data[data['Country Code']==216]
d={}
df.dropna(inplace=True)
for i in df.values:
    for j in i[9].split(','):
        d[j.strip()]=d.get(j.strip(),0)+1
l=sorted(d.values())
l.reverse()
l1=[]
l2=[]
for i in range(10):
    for j in d:
        if d[j]==l[i]:
            print(j," : ",l[i])
            d[j]=-1
            l2.append(j)
            l1.append(l[i])
plt.pie(l1,labels=l2,autopct="%.2f%%")
plt.show()
```

```
American  :  112
Seafood  :  59
Burger  :  49
Sandwich  :  49
Pizza  :  49
Steak  :  42
Italian  :  38
Breakfast  :  37
Mexican  :  36
Sushi  :  34

C:\Users\RAVI KUMAR RUNGTA\anaconda3\lib\site-packages\ipykernel_launcher.p
y:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pand
as.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-v
ersus-a-copy)
  This is separate from the ipykernel package so we can avoid doing imports
 until
```



Plot the bubble graph of a number of Restaurants present in the city of India and keeping the weighted

restaurant rating of the city in a bubble.

In [64]:

```
india=data[data["Country Code"]==1]
count=india["City"].value_counts()
city_rating_votes=india.loc[:,("City","Aggregate rating","Votes")]

city_rating_votes["Aggregate rating"]=city_rating_votes["Aggregate rating"]*city_rating_vot



city_votes=city_rating_votes.groupby("City")["Votes"].agg("sum")#
city_rating=city_rating_votes.groupby("City")["Aggregate rating"].agg("sum")
city_rating_votes=pd.DataFrame(city_rating)
city_rating_votes["Votes"]=city_votes

city_rating_votes["Weighted Rating"]=city_rating_votes["Aggregate rating"]/city_rating_vote
city_weighted=city_rating_votes[["Weighted Rating"]]#.sort_values(ascending=False)

a=pd.DataFrame(city_weighted)
a["counts"]=count
print(a)
a
plt.figure(figsize=(15, 10), dpi=70)
plt.scatter(a.index,a.counts,s=a['Weighted Rating'] ,c='b',edgecolor='r',marker="*")
plt.xlabel("City")

plt.ylabel("Number of Restaurant")
plt.xticks(rotation=60)
plt.show()
```

City	Weighted Rating	counts
Agra	4.006984	20
Ahmedabad	4.163215	21
Allahabad	3.409626	20
Amritsar	3.764584	21
Aurangabad	3.427315	20
Bangalore	4.497423	20
Bhopal	4.125930	20
Bhubaneshwar	3.967759	21
Chandigarh	4.106691	18
Chennai	4.319682	20
Coimbatore	4.174994	20
Dehradun	4.016173	20
Faridabad	3.482979	251
Ghaziabad	3.037870	25
Goa	4.158438	20
Gurgaon	3.743958	1118
Guwahati	4.265733	21
Hyderabad	4.487773	18
Indore	3.998346	20
Jaipur	4.281508	20
Kanpur	3.853793	20
Kochi	4.137380	20
Kolkata	4.295152	20
Lucknow	4.323527	21
Ludhiana	4.083607	20
Mangalore	3.761117	20
Mohali	4.300000	1
Mumbai	4.216608	20
Mysore	3.761202	20

Nagpur	4.108652	20
Nashik	3.587255	20
New Delhi	3.765872	5473
Noida	3.472631	1080
Panchkula	4.200000	1
Patna	3.455066	20
Puducherry	3.737062	20
Pune	4.283996	20
Ranchi	3.573603	20
Secunderabad	4.548055	2
Surat	4.017698	20
Vadodara	4.131989	20
Varanasi	3.563852	20
Vizag	4.131187	20