```
# 1 Import Libraries


import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.graph_objs as gl


import nltk
from nltk.corpus import stopwords
from sklearn.metrics.pairwise import linear_kernel
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer


# 2 Read the Dataset:


zomato_data=pd.read_csv("zomato.csv")
zomato_df = zomato_data

zomato_df.head(2)
```

```
---------------------------------------------------------------------------
ParserError                               Traceback (most recent call last)
<ipython-input-12-cf9e1218e127> in <cell line: 1>()
----> 1 zomato_data=pd.read_csv("zomato.csv")
      2 zomato_df = zomato_data
      3
      4 zomato_df.head(2)
```

9 frames

```
/usr/local/lib/python3.10/dist-packages/pandas/_libs/parsers.pyx in
pandas._libs.parsers.raise_parser_error()

ParserError: Error tokenizing data. C error: EOF inside string starting at row
766
```

SEARCH STACK OVERFLOW

```
# 3 Analyze the dataset


zomato_df.shape
```

```
(391, 17)
```

```
zomato_df.columns
```

```
Index(['url', 'address', 'name', 'online_order', 'book_table', 'rate', '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')
```

[illegible]

```
zomato_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 391 entries, 0 to 390
Data columns (total 17 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   url                          391 non-null    object 
 1   address                      390 non-null    object 
 2   name                         390 non-null    object 
 3   online_order                 390 non-null    object 
 4   book_table                   390 non-null    object 
 5   rate                         355 non-null    object 
 6   votes                        390 non-null    float64
 7   phone                        386 non-null    object 
 8   location                     390 non-null    object 
 9   rest_type                    390 non-null    object 
 10  dish_liked                   182 non-null    object 
 11  cuisines                     390 non-null    object 
 12  approx_cost(for two people)  390 non-null    object 
 13  reviews_list                 390 non-null    object 
 14  menu_item                    390 non-null    object 
 15  listed_in(type)              390 non-null    object 
 16  listed_in(city)              390 non-null    object 
dtypes: float64(1), object(16)
memory usage: 52.1+ KB
```

```
zomato_df.isnull().sum()
```

```
url                              0
address                          1
name                             1
online_order                     1
book_table                       1
rate                            36
votes                            1
phone                            5
location                         1
rest_type                        1
dish_liked                     209
cuisines                         1
approx_cost(for two people)      1
reviews_list                     1
menu_item                        1
listed_in(type)                  1
listed_in(city)                  1
dtype: int64
```

Data Cleaning & Preprocessing

```
zomato_df=zomato_df.drop(['phone','dish_liked','url'],axis=1)
```

```
zomato_df=zomato_df.drop(['phone','dish_liked','url'],axis=1)
```

```
zomato_df.dropna(how='any',inplace=True)
```

```
zomato_df.duplicated().sum()
zomato_df.drop_duplicates(inplace=True)
```

```
zomato_df=zomato_df.rename(columns={'approx_cost(for two people)':'cost','listed_in(ci
```

```
zomato_df=zomato_df.loc[zomato_df.rate !='NEW']
zomato_df=zomato_df.loc[zomato_df.rate !='-'].reset_index(drop=True)
remove_slash=lambda x: x.replace('/5','') if type(x)==np.str else x
zomato_df.rate= zomato_df.rate.apply(remove_slash).str.strip().astype('float')
```

```
<ipython-input-22-291affcf1392>:3: DeprecationWarning: `np.str` is a deprecated al
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs
  remove_slash=lambda x: x.replace('/5','') if type(x)==np.str else x
```

```
#Changing the cost to string
zomato_df['cost'] = zomato_df['cost'].astype(str)
zomato_df['cost'] = zomato_df['cost'].apply(lambda x:x.replace( ',','.'))
zomato_df['cost']= zomato_df['cost'].astype(float)
```

```
# checking for null values after cleaning & data Processing
```

```
zomato_df.isnull().sum()
```

```
address            0
name               0
online_order       0
book_table         0
rate               0
votes              0
location           0
rest_type          0
cuisines           0
cost               0
reviews_list       0
menu_item          0
listed_in(type)    0
city               0
dtype: int64
```

Checking mean rating with restaurant name and rating for each restaurant using below line codes

```
# computing the mean
# computing the mean
restaurants = list(zomato_df['name'].unique())
zomato_df['Mean Rating']= 0
for i in range(len(restaurants)):
  zomato_df['Mean Rating'][zomato_df['name']== restaurants[i]] = zomato_df['rate'][zom

# Scaling the mean rating values
from sklearn.preprocessing import MinMaxScaler
scaler =MinMaxScaler(feature_range =(1,5))
zomato_df[['Mean Rating']]=scaler.fit_transform(zomato_df[['Mean Rating']]).round(2)
```

```
<ipython-input-30-7ba463f05ebf>:5: 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
  zomato_df['Mean Rating'][zomato_df['name']== restaurants[i]] = zomato_df['rate']
```

Checking the mean rating with restaurant name and rating

```
zomato_df[['name','rate','Mean Rating']].head()
```

	name	rate	Mean Rating
0	Jalsa	4.1	4.05
1	Spice Elephant	4.1	4.05
2	San Churro Cafe	3.8	3.48
3	Addhuri Udupi Bhojana	3.7	3.29
4	Grand Village	3.8	3.48

Text Preprocessing and Cleaning

```
# Lower Casing

zomato_df["reviews_list"]=zomato_df["reviews_list"].str.lower()

# removal of the Punctuations
import string
PUNCT_TO_REMOVE =string.punctuation
def remove_punctuation(text):
  """custom function to remove the punctuation"""
  return text.translate(str.maketrans('','', PUNCT_TO_REMOVE))
zomato_df["reviews_list"]=zomato_df["reviews_list"].apply(lambda text: remove_punctuat
```

```
zomato_df[['reviews_list', 'cuisines']].sample(5)
```

```
zomato_df[['reviews_list','cuisines']].sample(5)
```

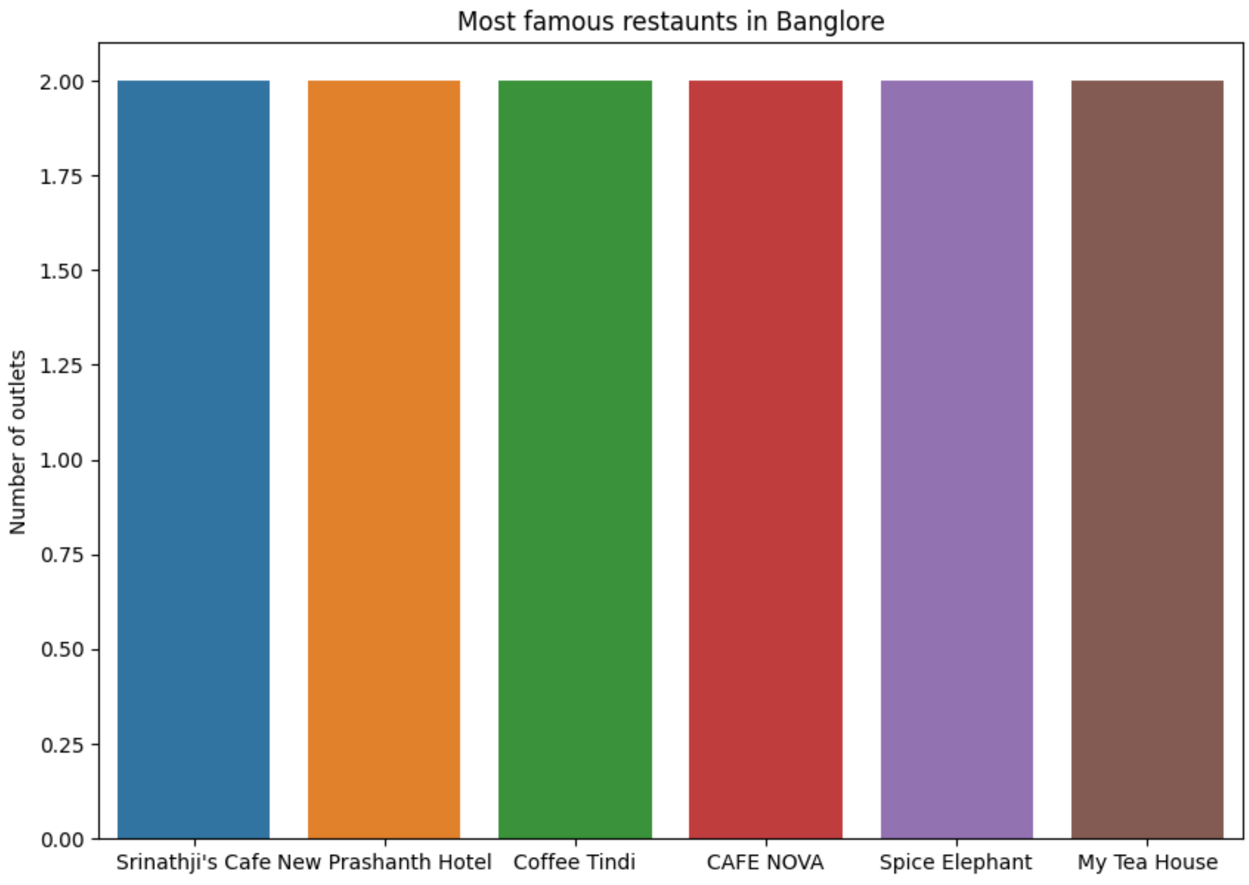
	reviews_list	cuisines
266	rated 10 ratedn this place is right opposite ...	Cafe, Fast Food, Continental, Chinese, Momos
66	rated 40 ratedn quite a cosy small joint but...	South Indian
290	rated 30 ratedn had been for reviewing herenw...	Cafe, Italian, Pizza
224		Fast Food

Data Vizualization

```
# most famous 6 restaunsts in banglore

plt.figure(figsize=(10,7))
chains=zomato_df['name'].value_counts()[:6]
sns.barplot(x=chains.index,y=chains,palette='tab10')
plt.title("Most famous restaunts in Banglore")
plt.ylabel("Number of outlets")
```

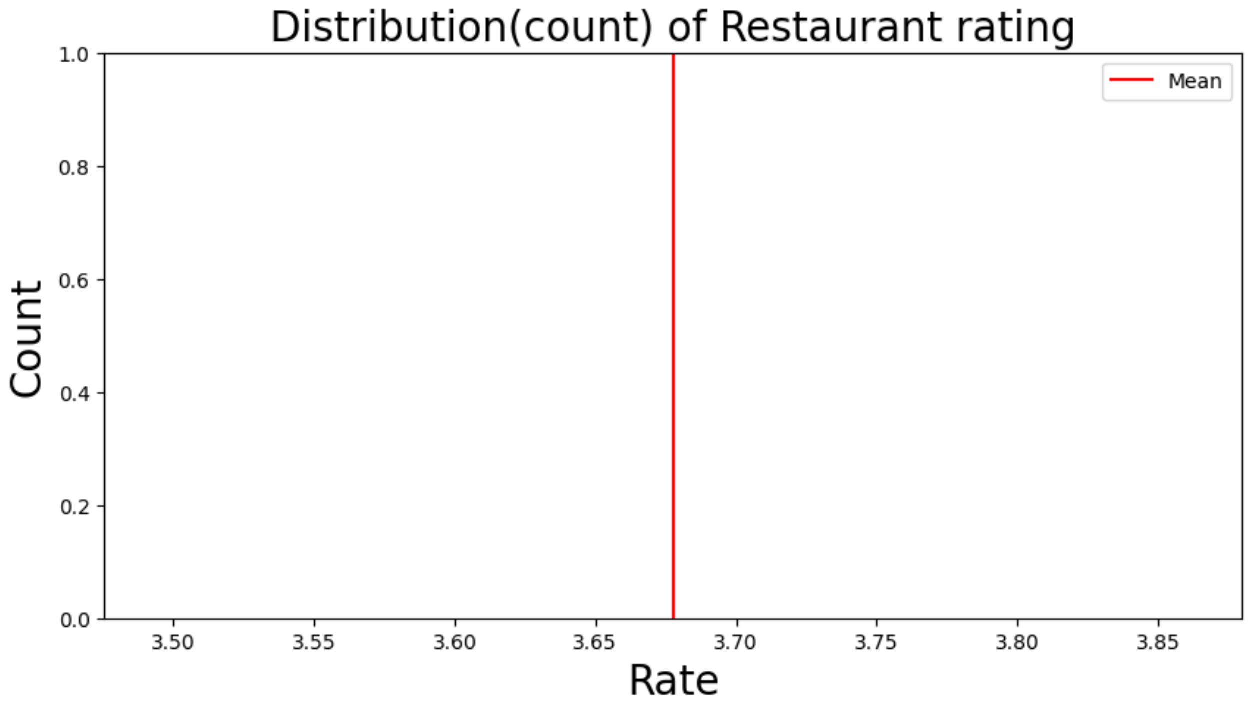
```
Text(0, 0.5, 'Number of outlets')
```

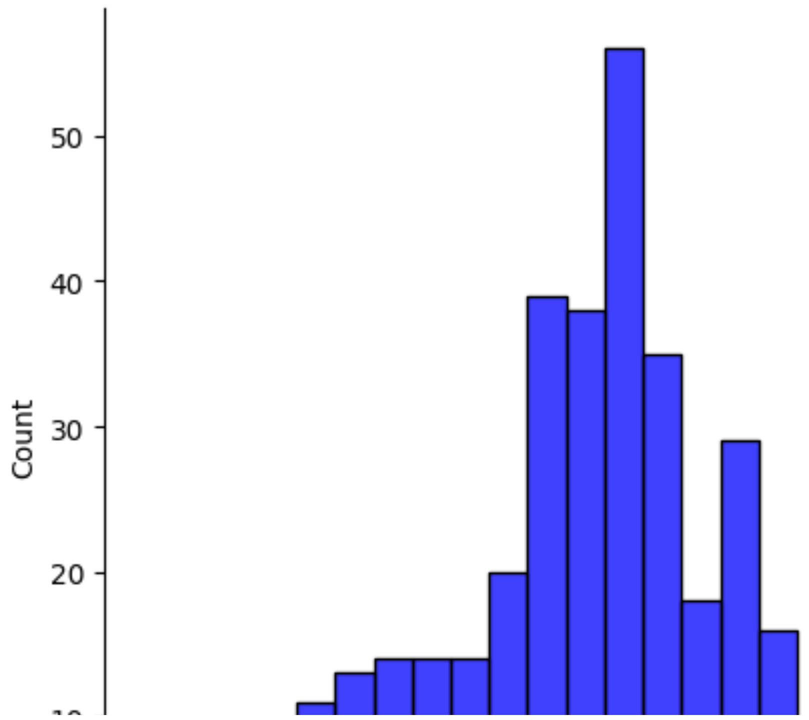


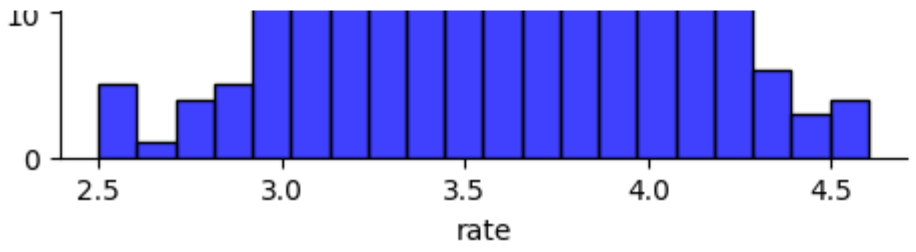
```
# Distribution of Restaurant Rating

fig, ax =plt.subplots(nrows=1,ncols=1, figsize=(10,5))
sns.displot(zomato_df.rate,kde=False,color='b',ax=ax,bins=20);
ax.axvline(zomato_df.rate.mean(),0,1,color='r',label='Mean')
ax.legend();
ax.set_ylabel('Count',size=20)
ax.set_xlabel('Rate',size=20)
ax.set_title('Distribution(count) of Restaurant rating',size=20);
```

```
    /usr/local/lib/python3.10/dist-packages/seaborn/distributions.py:2142: UserWarning
      warnings.warn(msg, UserWarning)
```



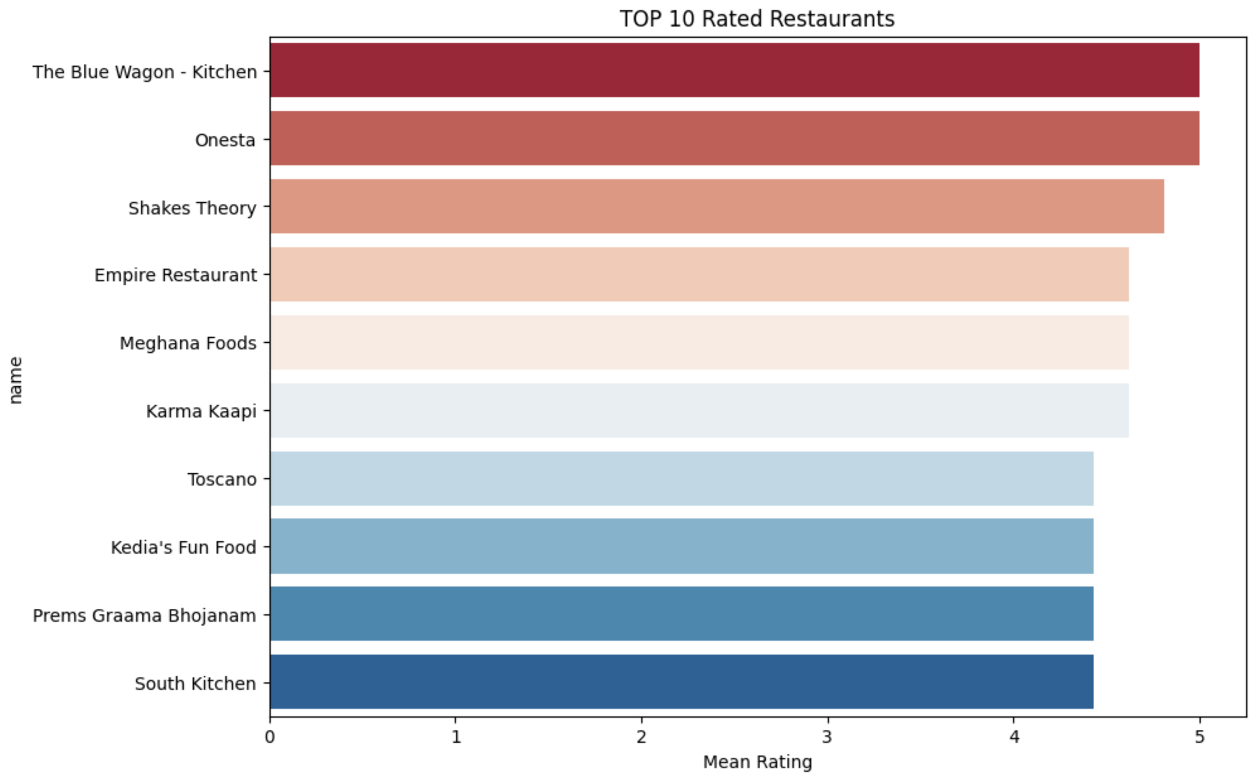




```
# TOP 10 Rated Restaurants

df_rating =zomato_df.drop_duplicates(subset='name')
df_rating =df_rating.sort_values(by='Mean Rating', ascending=False).head(10)
plt.figure(figsize=(10,7))
sns.barplot(data=df_rating, x='Mean Rating', y='name',palette='RdBu')
plt.title('TOP 10 Rated Restaurants')
```

```
     Text(0.5, 1.0, 'TOP 10 Rated Restaurants')
```

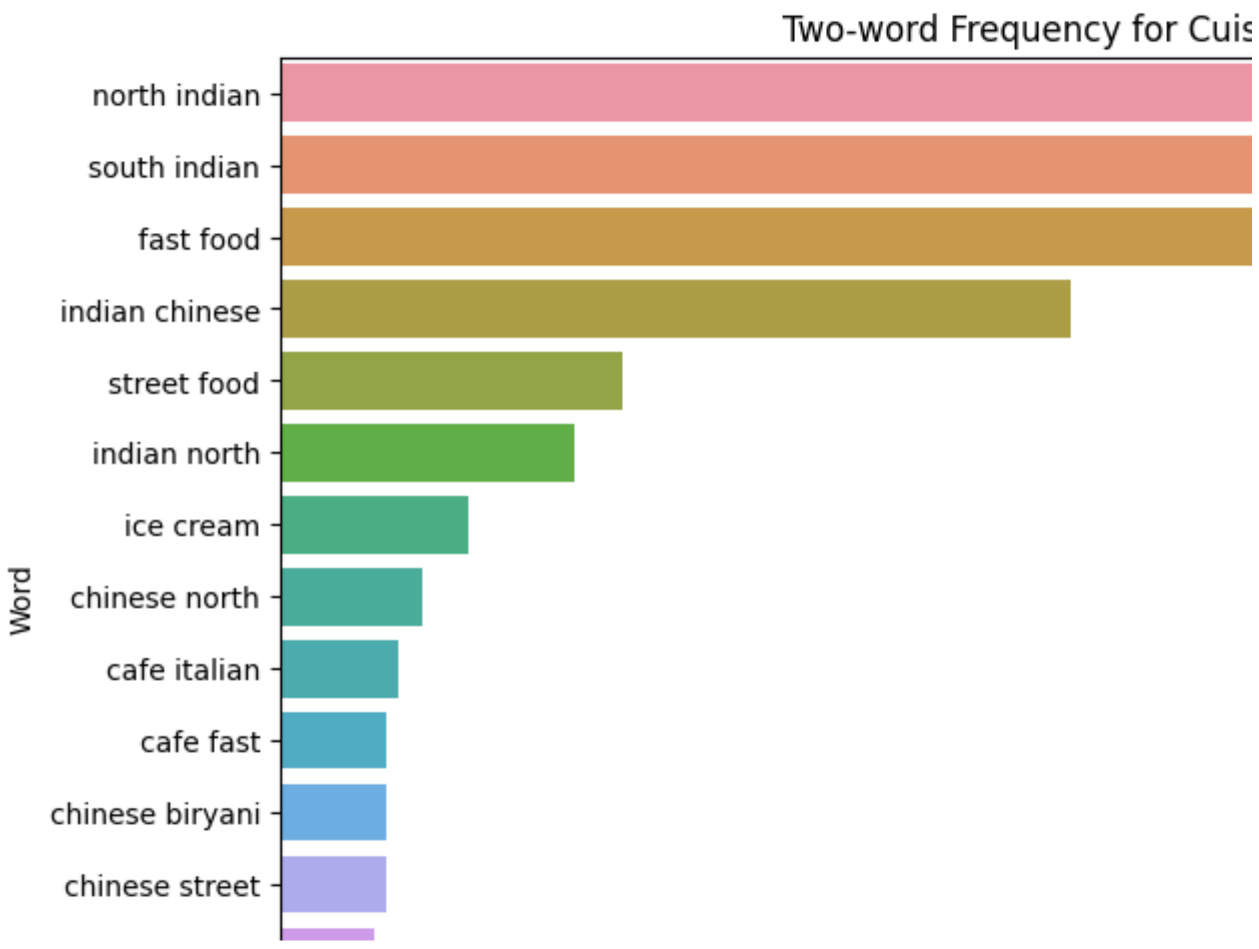


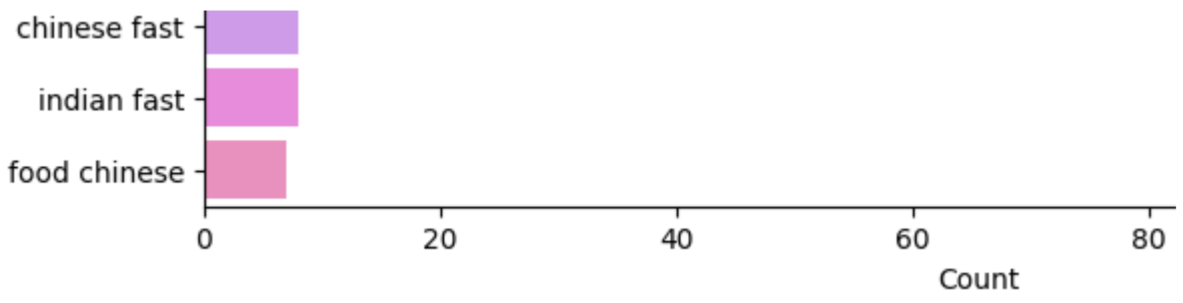
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import CountVectorizer

def get_top_words(column, n, ngram_range):
    vectorizer = CountVectorizer(ngram_range=ngram_range, stop_words=None)
    X = vectorizer.fit_transform(column)
    words = vectorizer.get_feature_names_out()
    word_counts = X.sum(axis=0).A1
    word_freq = dict(zip(words, word_counts))
    sorted_word_freq = sorted(word_freq.items(), key=lambda x: x[1], reverse=True)
    return sorted_word_freq[:n]

# Assuming that 'zomato_df' is your DataFrame and 'cuisines' is the column of interest
lst = get_top_words(zomato_df['cuisines'], 15, (2, 2))
df_words = pd.DataFrame(lst, columns=['Word', 'Count'])

plt.figure(figsize=(10, 7))
sns.barplot(data=df_words, x='Count', y='Word')
plt.title('Two-word Frequency for Cuisines')
plt.show()
```





CONTENT-BASE RECOMMENDER SYSTEM

```
df_percent = pd.DataFrame(zomato_df)
df_percent.set_index('name', inplace=True)
indices =pd.Series(df_percent.index)

# creating tf-idf matrix

tfidf =TfidfVectorizer(analyzer='word', ngram_range=(1,2),min_df=0,stop_words='english
tfidf_matrix= tfidf.fit_transform(df_percent['reviews_list'])
cosine_similarities =linear_kernel(tfidf_matrix,tfidf_matrix)
```

Creating Recommendation system

```
# Assuming df_percent is your DataFrame
available_restaurants = df_percent.index.unique()

# Print the available restaurants
print("Available Restaurants:")
for restaurant in available_restaurants:
    print(restaurant)
```

```
Available Restaurants:
Jalsa
Spice Elephant
San Churro Cafe
Addhuri Udupi Bhojana
Grand Village
Timepass Dinner
Rosewood International Hotel - Bar & Restaurant
Onesta
Penthouse Cafe
Smacznego
CafÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂ© Down The Alley
Cafe Shuffle
The Coffee Shack
Caf-Eleven
Cafe Vivacity
Catch-up-ino
Kirthi's Biryani
T3H Cafe
```

```
     360 Atoms Restaurant And Cafe
     The Vintage Cafe
     Woodee Pizza
     Cafe Coffee Day
     My Tea House
     Hide Out Cafe
     CAFE NOVA
     Coffee Tindi
     Sea Green Cafe
     Cuppa
     Srinathji's Cafe
     Redberrys
     Foodiction
     Sweet Truth
     Ovenstory Pizza
     Faasos
     Behrouz Biryani
     Fast And Fresh
     Szechuan Dragon
     Empire Restaurant
     Maruthi Davangere Benne Dosa
     Chaatimes
     Havyaka Mess
     McDonald's
     Domino's Pizza
     Hotboxit
     Kitchen Garden
     Recipe
     Beijing Bites
     Tasty Bytes
     Petoo
     Shree Cool Point
     Corner House Ice Cream
     Biryanis And More
     Roving Feast
     FreshMenu
     Banashankari Donne Biriyani
     Wamama
     Five Star Chicken
```

```
def recommend(name,cosine_similarities= cosine_similarities):
  # create a list to put top restaurants
  recommend_restaurant=[]

  # find the index of the hotel entered
  idx =indices[indices==name].index[0]

  # find the restaurant with a similar cosine-sin value
  score_series =pd.Series(cosine_similarities[idx]).sort_values(ascending=False)

  top30_indexes=list(score_series.iloc[0:31].index)

  for each in top30_indexes:
    recommend_restaurant.append(list(df_percent.index)[each])

  df_new=pd.DataFrame(columns=['cuisines','Mean Rating','cost'])
```

```
    for each in recommend_restaurant:
      df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']][d

    df_new=df_new.drop_duplicates(subset=['cuisines','Mean Rating','cost'],keep=False)
    df_new=df_new.sort_values(by='Mean Rating', ascending=False).head(10)
    print('TOP %s RESTAURANTS LIKE %s WITH SIMILAR REVIEWS:'%(str(len(df_new)),name))


    return df_new
```

```
recommend('Salut')
```

```
TOP 10 RESTAURANTS LIKE Salut WITH SIMILAR REVIEWS:
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
```

	cuisines	Mean Rating	cost
Onesta	Pizza, Cafe, Italian	5.00	600.0
The Blue Wagon - Kitchen	Cafe, Beverages	5.00	400.0
Empire Restaurant	North Indian, Mughlai, South Indian, Chinese	4.62	750.0
South Kitchen	South Indian	4.43	100.0
Szechuan Dragon	Chinese, Thai, Momos	4.24	600.0
Salut	Continental, Finger Food, Seafood, Pizza	4.05	1.2
Gufha - The President Hotel	North Indian, Afghani, Mughlai	4.05	1.2
Cafe Aira	Cafe, Continental, Beverages, Desserts	4.05	500.0

```
recommend('Onesta')
```

```
TOP 10 RESTAURANTS LIKE Onesta WITH SIMILAR REVIEWS:
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
```

```
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
```

	cuisines	Mean Rating	cost
The Blue Wagon - Kitchen	Cafe, Beverages	5.00	400.0
Toscano	Italian, Salad	4.43	1.3
South Kitchen	South Indian	4.43	100.0
Szechuan Dragon	Chinese, Thai, Momos	4.24	600.0

Mojo Pizza - 2X Toppings	Pizza	4.24	600.0
Gufha - The President Hotel	North Indian, Afghani, Mughlai	4.05	1.2
Cafe Aira	Cafe, Continental, Beverages, Desserts	4.05	500.0
Salut	Continental, Finger Food, Seafood, Pizza	4.05	1.2

```
recommend('Jalsa')
```

```
TOP 10 RESTAURANTS LIKE Jalsa WITH SIMILAR REVIEWS:
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
<ipython-input-72-78b30c4f06b2>:19: FutureWarning: The frame.append method is depr
  df_new =df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating','cost']]
```

	cuisines	Mean Rating	cost
The Blue Wagon - Kitchen	Cafe, Beverages	5.00	400.0
Empire Restaurant	North Indian, Mughlai, South Indian, Chinese	4.62	750.0
South Kitchen	South Indian	4.43	100.0
The Coffee Shack	Cafe, Chinese, Continental, Italian	4.24	500.0
Szechuan Dragon	Chinese, Thai, Momos	4.24	600.0
Jalsa	North Indian, Mughlai, Chinese	4.05	800.0
Salut	Continental, Finger Food, Seafood, Pizza	4.05	1.2
The Biryani Cafe	Biryani, Chinese, Kebab	4.05	300.0