In [2]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```
df=pd.read_csv('Zomato-data-.csv')
df
```

Out[3]:

| | name | online_order | book_table | rate | votes | approx_cost(for two people) | listed_in(typ |
|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1/5 | 775 | 800 | Buff |
| **1** | Spice Elephant | Yes | No | 4.1/5 | 787 | 800 | Buff |
| **2** | San Churro Cafe | Yes | No | 3.8/5 | 918 | 800 | Buff |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7/5 | 88 | 300 | Buff |
| **4** | Grand Village | No | No | 3.8/5 | 166 | 600 | Buff |
| **...** | ... | ... | ... | ... | ... | ... | |
| **143** | Melting Melodies | No | No | 3.3/5 | 0 | 100 | Dinir |
| **144** | New Indraprasta | No | No | 3.3/5 | 0 | 150 | Dinir |
| **145** | Anna Kuteera | Yes | No | 4.0/5 | 771 | 450 | Dinir |
| **146** | Darbar | No | No | 3.0/5 | 98 | 800 | Dinir |
| **147** | Vijayalakshmi | Yes | No | 3.9/5 | 47 | 200 | Dinir |

148 rows × 7 columns

In [4]:
```
(r,c)=df.shape
print("rows:-",r)
print("columns:-",c)
```

```
rows:- 148
columns:- 7
```

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148 entries, 0 to 147
Data columns (total 7 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   name                         148 non-null    object
 1   online_order                 148 non-null    object
 2   book_table                   148 non-null    object
 3   rate                         148 non-null    object
 4   votes                        148 non-null    int64 
 5   approx_cost(for two people)  148 non-null    int64 
 6   listed_in(type)              148 non-null    object
dtypes: int64(2), object(5)
memory usage: 8.2+ KB
```

In [6]:
```
df.columns
```

Out[6]:
```
Index(['name', 'online_order', 'book_table', 'rate', 'votes',
       'approx_cost(for two people)', 'listed_in(type)'],
      dtype='object')
```

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

Out[7]:

| | **votes** | **approx_cost(for two people)** |
|---|---|---|
| **count** | 148.000000 | 148.000000 |
| **mean** | 264.810811 | 418.243243 |
| **std** | 653.676951 | 223.085098 |
| **min** | 0.000000 | 100.000000 |
| **25%** | 6.750000 | 200.000000 |
| **50%** | 43.500000 | 400.000000 |
| **75%** | 221.750000 | 600.000000 |
| **max** | 4884.000000 | 950.000000 |

In [8]:
```
df.dtypes
```

Out[8]:
```
name                           object
online_order                   object
book_table                     object
rate                           object
votes                           int64
approx_cost(for two people)     int64
listed_in(type)                object
dtype: object
```

In [9]:
```
# chcvk for nul values
df.isnull().sum()
```

```
Out[9]:  name                           0
         online_order                   0
         book_table                     0
         rate                           0
         votes                          0
         approx_cost(for two people)    0
         listed_in(type)                0
         dtype: int64
```

```
In [10]: #check a duplicate value
         duplicated_rows=df[df.duplicated()]
         print("duplicate rows based on all columns:")
         print(duplicated_rows)
```

```
duplicate rows based on all columns:
Empty DataFrame
Columns: [name, online_order, book_table, rate, votes, approx_cost(for two peopl
e), listed_in(type)]
Index: []
```

```
In [11]: total_sum_duplicated_valuse=df.duplicated().sum()
         print(total_sum_duplicated_valuse)
```

```
0
```

```
In [12]: df['rate'] = df['rate'].astype(str).str.replace('/5', '', regex=False).str.strip
         df['rate'] = pd.to_numeric(df['rate'], errors='coerce')
         df
         df = df.rename(columns={
             "approx_cost(for two people)": "cost_for_two",
             "listed_in(type)": "restaurant_type"
         })
         df
```

Out[12]:

|  | name | online_order | book_table | rate | votes | cost_for_two | restaurant_type |
|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | 800 | Buffet |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | 800 | Buffet |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | 800 | Buffet |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | 300 | Buffet |
| **4** | Grand Village | No | No | 3.8 | 166 | 600 | Buffet |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **143** | Melting Melodies | No | No | 3.3 | 0 | 100 | Dining |
| **144** | New Indraprasta | No | No | 3.3 | 0 | 150 | Dining |
| **145** | Anna Kuteera | Yes | No | 4.0 | 771 | 450 | Dining |
| **146** | Darbar | No | No | 3.0 | 98 | 800 | Dining |
| **147** | Vijayalakshmi | Yes | No | 3.9 | 47 | 200 | Dining |

148 rows × 7 columns

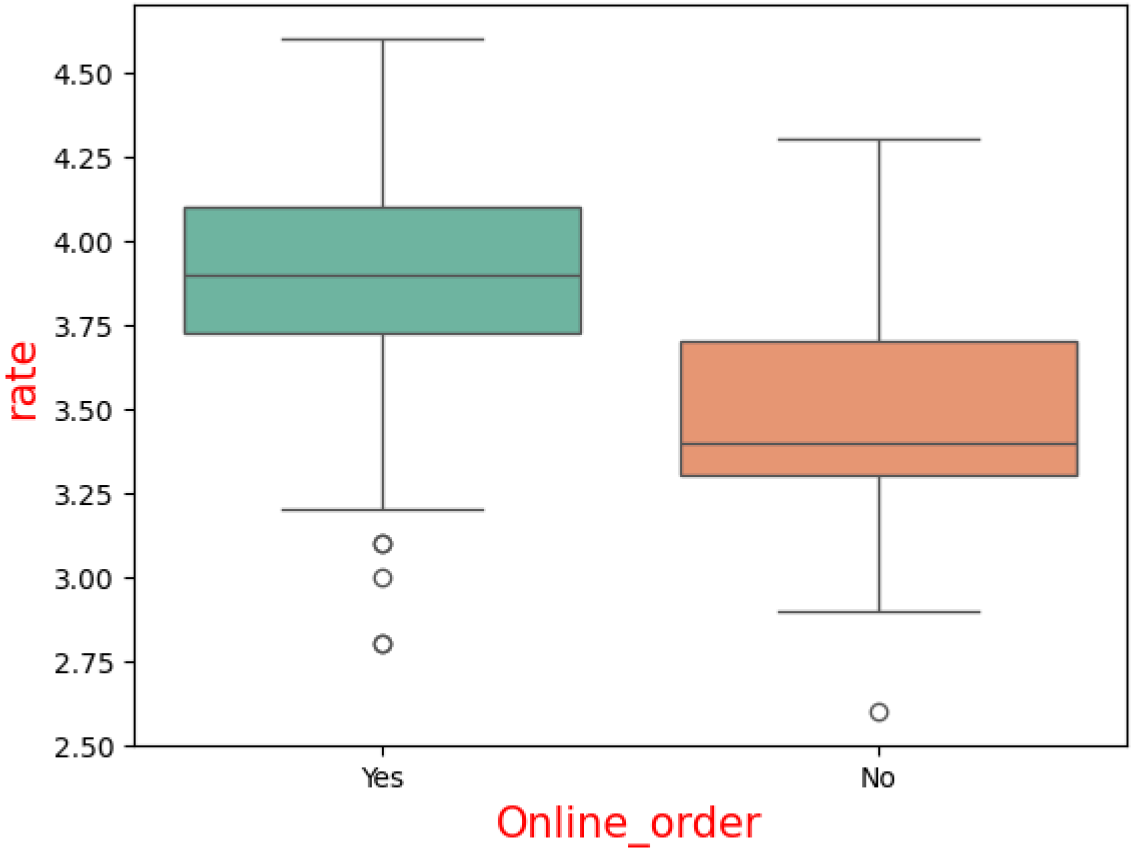
In [13]:

```
sns.boxplot(x='online_order',y='rate',data=df,palette="Set2")
plt.xlabel('Online_order',c='r',size=15)
plt.ylabel('rate',c='r',size=15)
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_12212\2374956612.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.boxplot(x='online_order',y='rate',data=df,palette="Set2")
```

Out[13]:

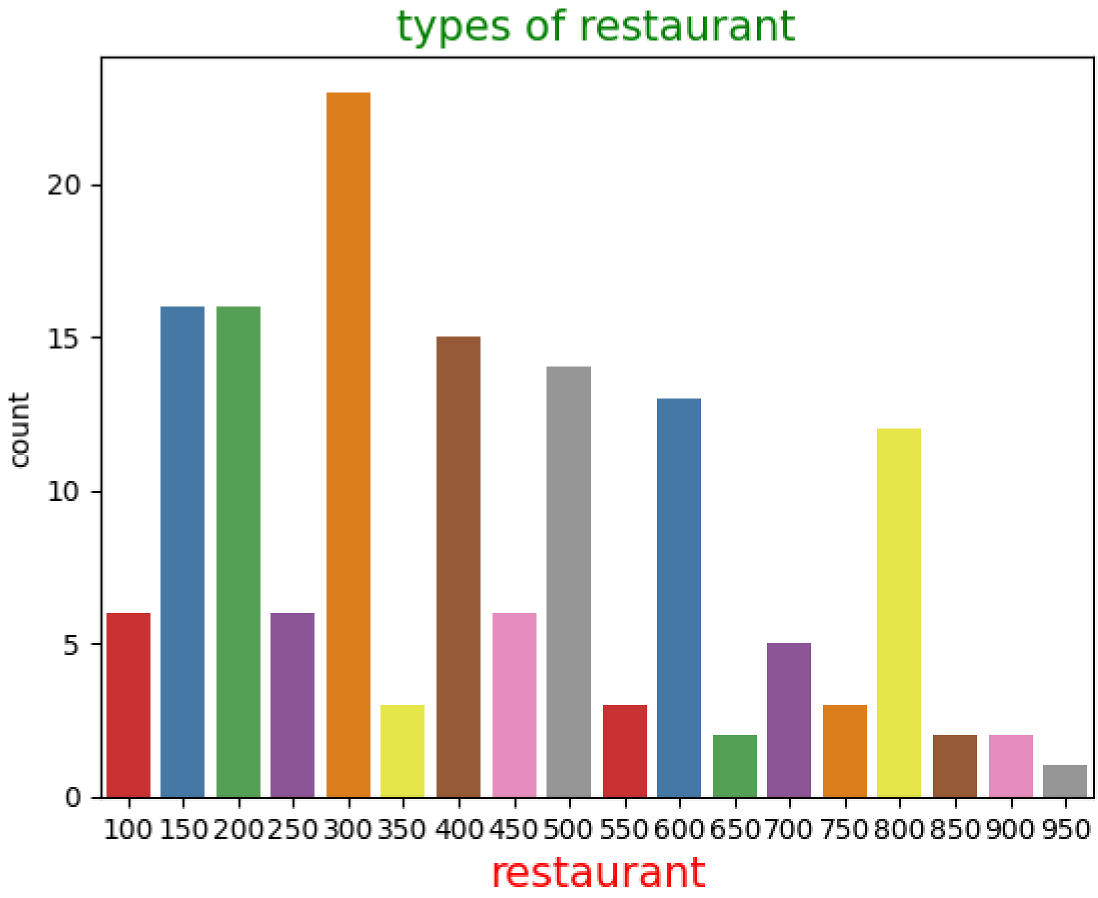
```
Text(0, 0.5, 'rate')
```



In [14]:
```
sns.countplot(x='cost_for_two', data=df, palette="Set1")
plt.title('types of restaurant',c='g',size='15')
plt.xlabel('restaurant',c='r',size='15')
```

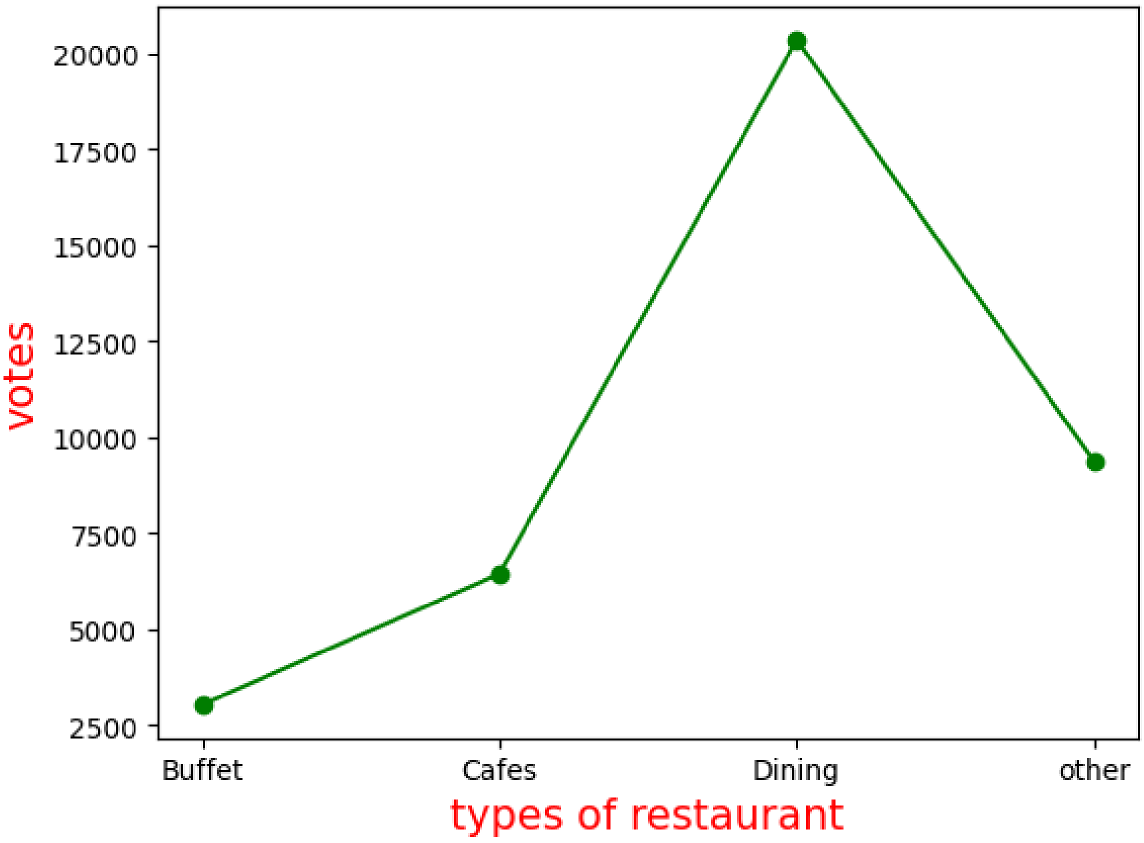
```
C:\Users\hp\AppData\Local\Temp\ipykernel_12212\646652665.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.countplot(x='cost_for_two', data=df, palette="Set1")
```

Out[14]:
```
Text(0.5, 0, 'restaurant')
```

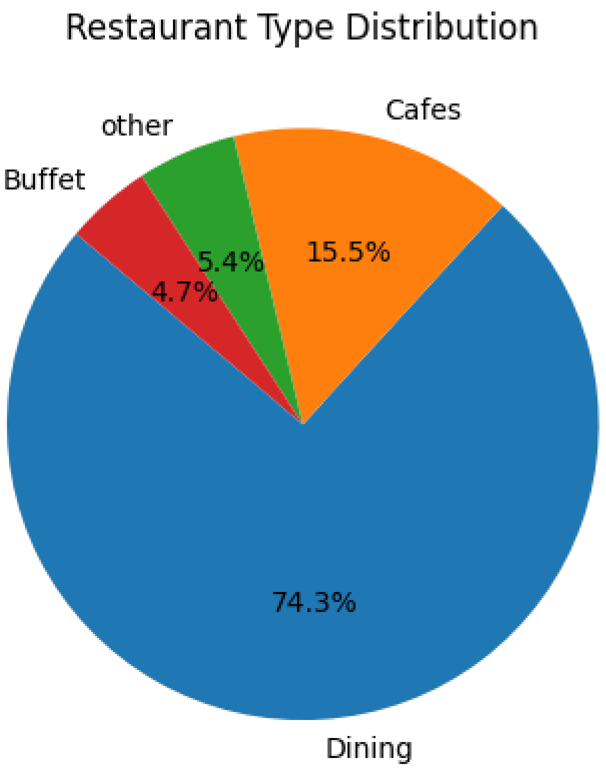


```
In [15]: df2=df.groupby('restaurant_type')['votes'].sum()
         df2
         plt.plot(df2,c='g',marker='o')
         plt.xlabel('types of restaurant',c='r',size='15')
         plt.ylabel('votes',c='r',size='15')
```

```
Out[15]: Text(0, 0.5, 'votes')
```



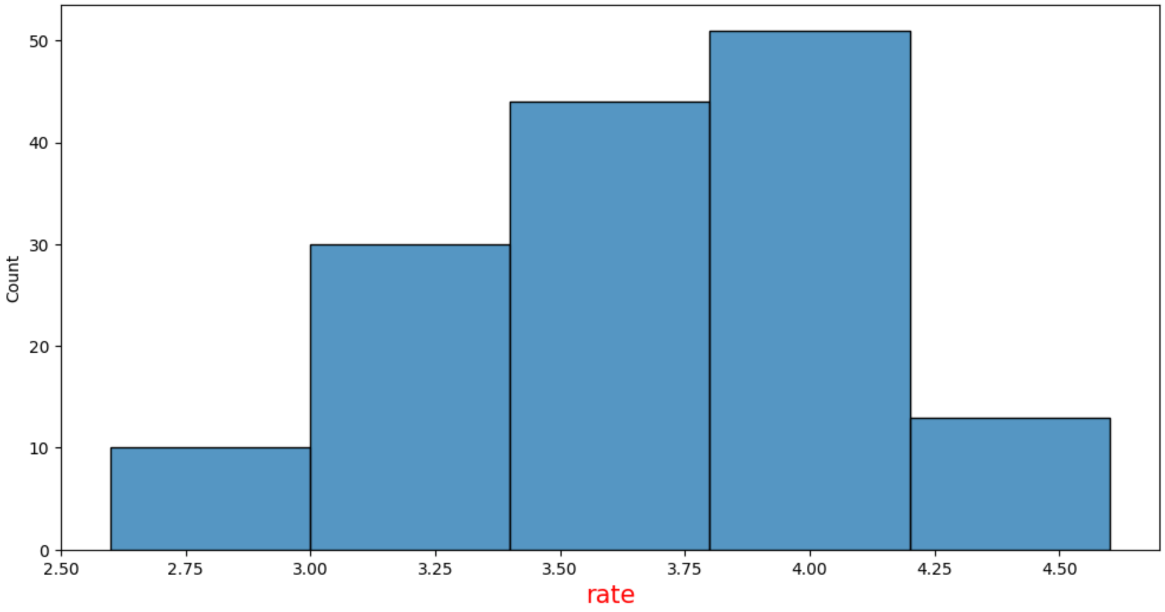
```
In [16]: type_counts = df['restaurant_type'].value_counts()
         plt.pie(type_counts, labels=type_counts.index, autopct='%1.1f%%', startangle=140
         plt.title("Restaurant Type Distribution")
         plt.show()
```



```
In [26]: plt.figure(figsize=(12,6))
         sns.histplot(x='rate', data=df, palette="Set1",bins=5)
         plt.xlabel('rate',c='r',size='15')
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_12212\2322393813.py:2: UserWarning: Igno
ring `palette` because no `hue` variable has been assigned.
  sns.histplot(x='rate', data=df, palette="Set1",bins=5)
```

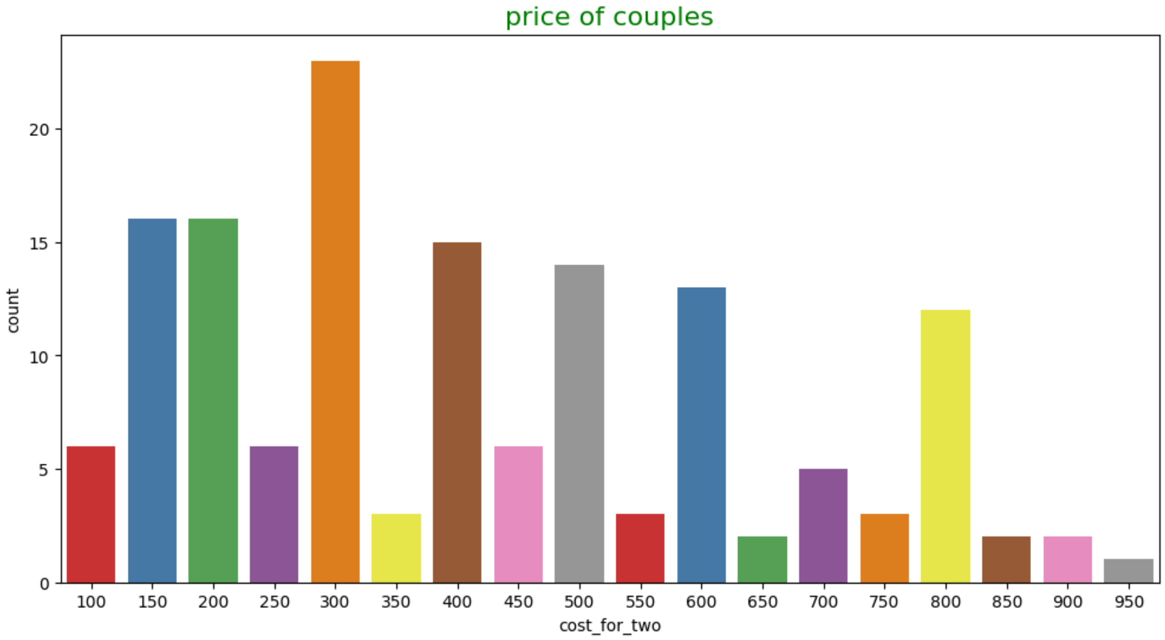
Out[26]: `Text(0.5, 0, 'rate')`



In [25]:
```
plt.figure(figsize=(12,6))
sns.countplot(x='cost_for_two', data=df, palette="Set1")
plt.title('price of couples',c='g',size=16)
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_12212\1686631152.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.countplot(x='cost_for_two', data=df, palette="Set1")
```

Out[25]: `Text(0.5, 1.0, 'price of couples')`



In [19]:
```
def price_category(cost):
    if cost <= 300:
        return "Low"
    elif cost <= 600:
        return "Medium"
    else:
        return "High"

df['price_category'] = df['cost_for_two'].apply(price_category)
```

In [20]:

```
df
```

Out[20]:

| | name | online_order | book_table | rate | votes | cost_for_two | restaurant_type |
|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | 800 | Buffet |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | 800 | Buffet |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | 800 | Buffet |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | 300 | Buffet |
| **4** | Grand Village | No | No | 3.8 | 166 | 600 | Buffet |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **143** | Melting Melodies | No | No | 3.3 | 0 | 100 | Dining |
| **144** | New Indraprasta | No | No | 3.3 | 0 | 150 | Dining |
| **145** | Anna Kuteera | Yes | No | 4.0 | 771 | 450 | Dining |
| **146** | Darbar | No | No | 3.0 | 98 | 800 | Dining |
| **147** | Vijayalakshmi | Yes | No | 3.9 | 47 | 200 | Dining |

148 rows × 8 columns

In [21]:

```
def rating_category(rating):
    if rating <= 2.5:
        return "Poor"
    elif rating <= 3.5:
        return "Average"
    elif rating <= 4.0:
        return "Good"
    else:
        return "Excellent"

df['rating_category'] = df['rate'].apply(rating_category)
```

In [22]:

```
df
```

Out[22]:

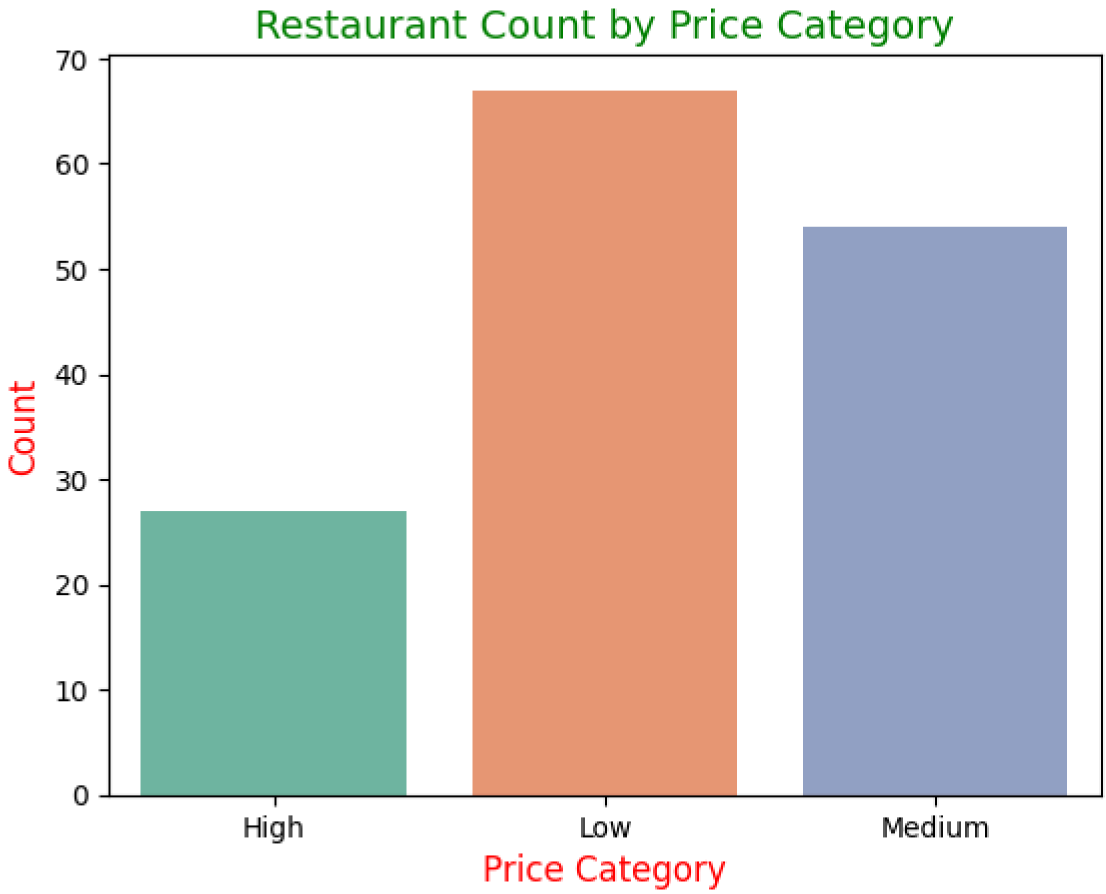
| | name | online_order | book_table | rate | votes | cost_for_two | restaurant_type |
|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | 800 | Buffet |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | 800 | Buffet |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | 800 | Buffet |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | 300 | Buffet |
| **4** | Grand Village | No | No | 3.8 | 166 | 600 | Buffet |
| **...** | ... | ... | ... | ... | ... | ... | ... |
| **143** | Melting Melodies | No | No | 3.3 | 0 | 100 | Dining |
| **144** | New Indraprasta | No | No | 3.3 | 0 | 150 | Dining |
| **145** | Anna Kuteera | Yes | No | 4.0 | 771 | 450 | Dining |
| **146** | Darbar | No | No | 3.0 | 98 | 800 | Dining |
| **147** | Vijayalakshmi | Yes | No | 3.9 | 47 | 200 | Dining |

148 rows × 9 columns

In [23]:

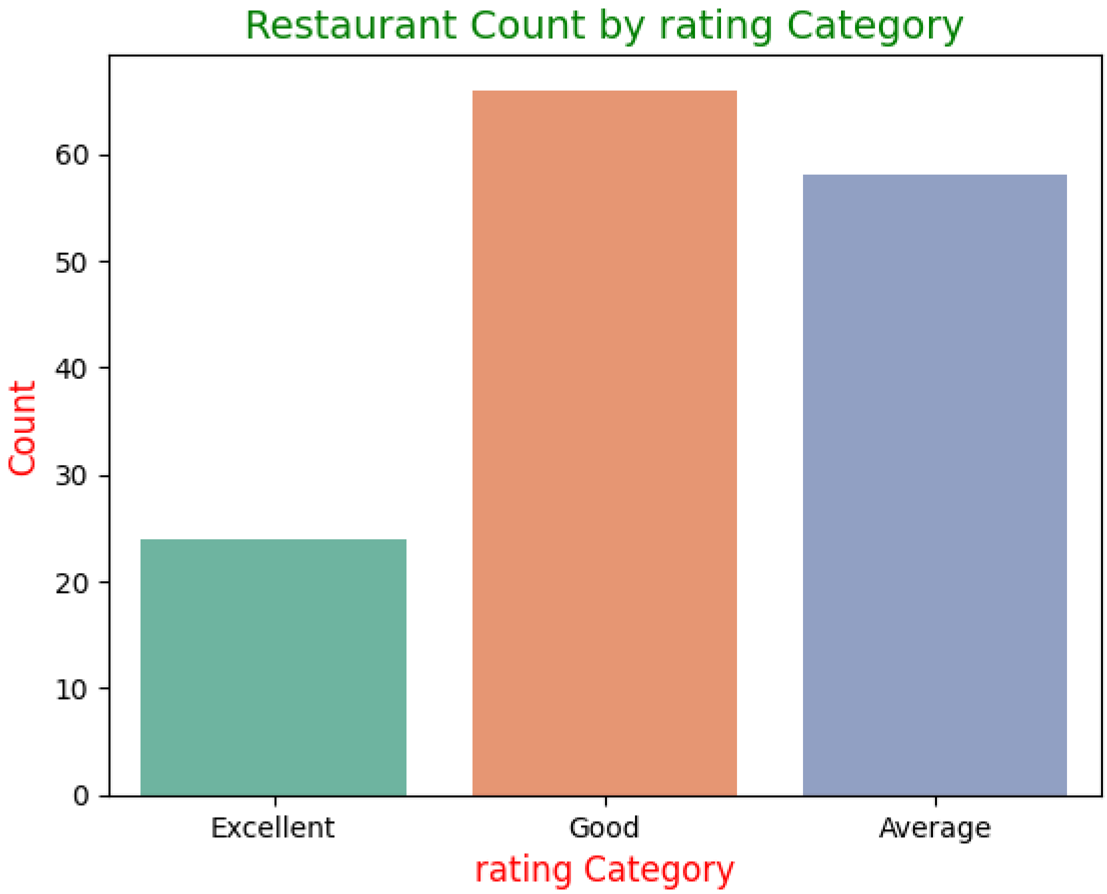
```
sns.countplot(x='price_category', data=df, palette="Set2")
plt.title("Restaurant Count by Price Category", fontsize=14, color="g")
plt.xlabel("Price Category", fontsize=12, color="r")
plt.ylabel("Count", fontsize=12, color="r")
plt.show()
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_13796\810351994.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.countplot(x='price_category', data=df, palette="Set2")
```



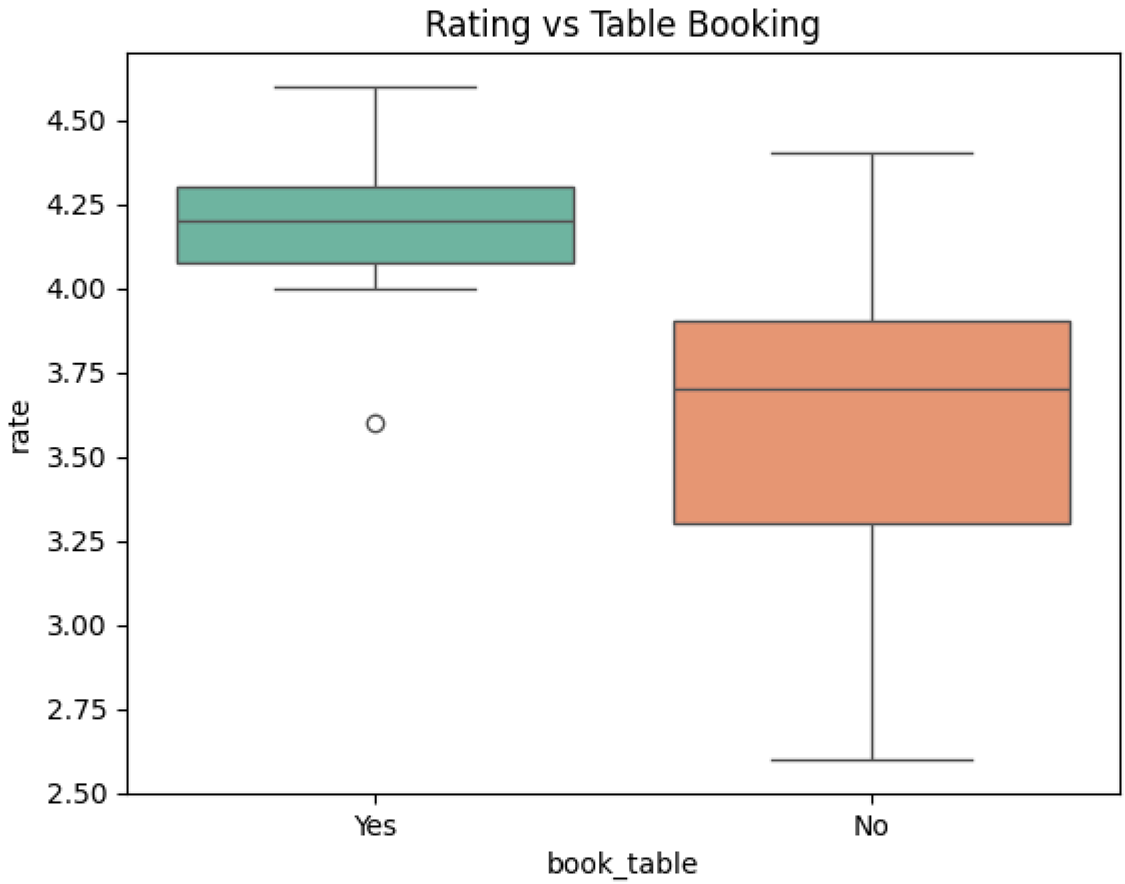
In [24]:
```
sns.countplot(x='rating_category', data=df, palette="Set2")
plt.title("Restaurant Count by rating Category", fontsize=14, color="g")
plt.xlabel("rating Category", fontsize=12, color="r")
plt.ylabel("Count", fontsize=12, color="r")
plt.show()
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_13796\2051898981.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.countplot(x='rating_category', data=df, palette="Set2")
```



```
In [25]: sns.boxplot(x='book_table', y='rate', data=df,palette="Set2")
         plt.title("Rating vs Table Booking")
         plt.show()
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_13796\2992761269.py:1: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.boxplot(x='book_table', y='rate', data=df,palette="Set2")
```



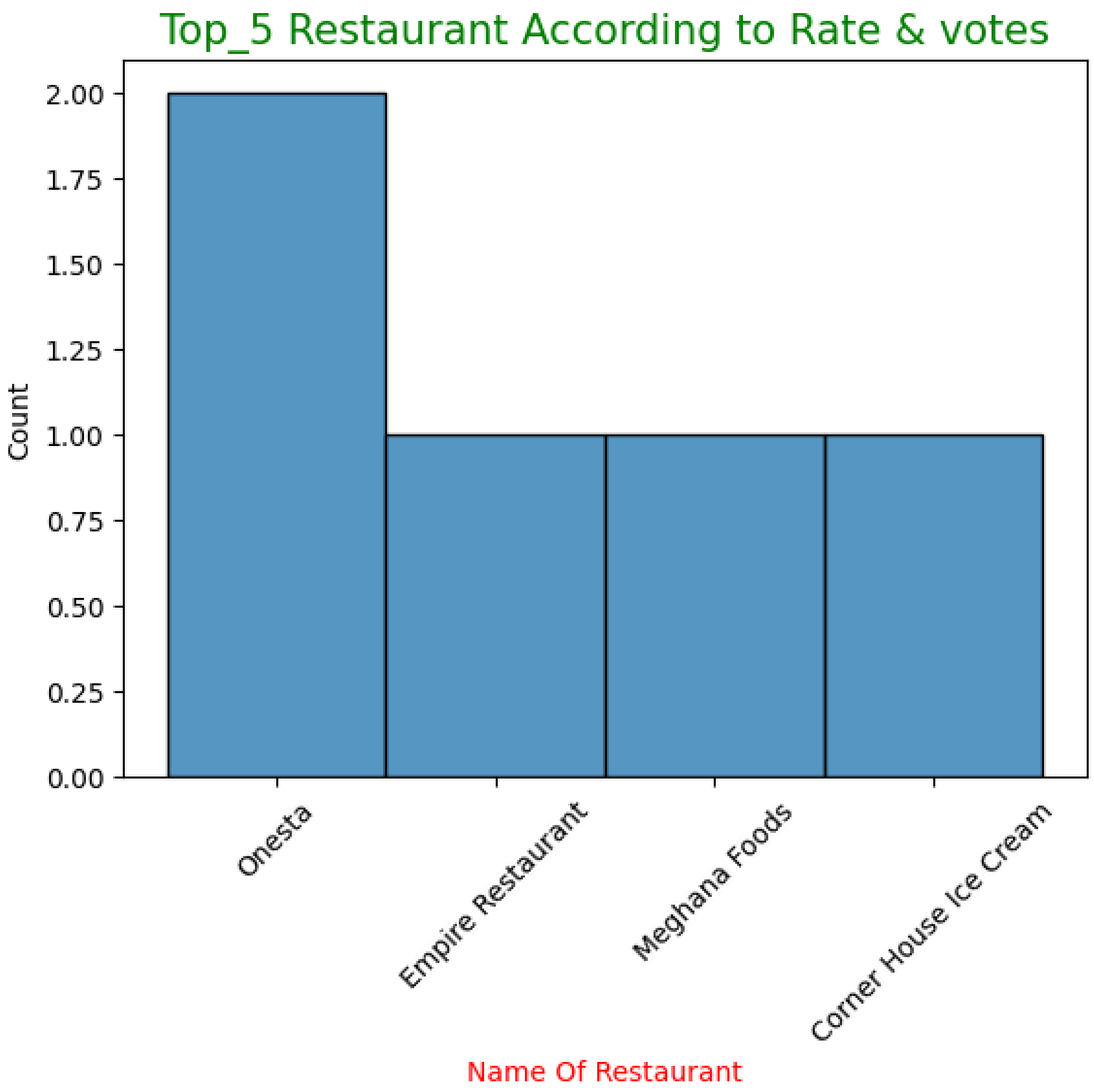
In [28]:
```
top_5 = df.sort_values(by=['rate', 'votes'], ascending=[False, False]).head(5)

# Sirf relevant columns dikhana
print(top_5[['name', 'rate', 'votes', 'cost_for_two']])
```

```
                     name  rate  votes  cost_for_two
7                  Onesta   4.6   2556           600
44                 Onesta   4.6   2556           600
38      Empire Restaurant   4.4   4884           750
86          Meghana Foods   4.4   4401           600
52  Corner House Ice Cream   4.3    345           400
```

In [31]:
```
sns.histplot(x='name', data=top_5,palette="Set2")
plt.title('Top_5 Restaurant According to Rate & votes',c='g',size=15)
plt.xlabel('Name Of Restaurant',c='r')
plt.xticks(rotation=45)
```

```
C:\Users\hp\AppData\Local\Temp\ipykernel_13796\1130827841.py:1: UserWarning: Igno
ring `palette` because no `hue` variable has been assigned.
  sns.histplot(x='name', data=top_5,palette="Set2")
```

Out[31]:
```
([0, 1, 2, 3],
 [Text(0, 0, 'Onesta'),
  Text(1, 0, 'Empire Restaurant'),
  Text(2, 0, 'Meghana Foods'),
  Text(3, 0, 'Corner House Ice Cream')])
```


Top_5 Restaurant According to Rate & votes
Count
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Onesta
Empire Restaurant
Meghana Foods
Corner House Ice Cream
Name Of Restaurant

```
In [ ]:
```

```
In [ ]:
```