```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('dark_background')
```

In [4]:

```
df = pd.read_csv(r'C:\Users\HOSHANGI\Downloads\zomato.csv')
df.head()
```

Out[4]:

| | 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) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | https://www.zomato.com/bangalore/jalsa-banasha... | 942, 21st Main Road, 2nd Stage, Banashankari, ... | Jalsa | Yes | Yes | 4.1/5 | 775 | 080 42297555\r\n+91 9743 772233 | Banashankari | Casual Dining | Pasta, Lunch Buffet, Masala Papad, Paneer Laja... | North Indian, Mughlai, Chinese | 800 | [('Rated 4.0', 'RATED\n A beautiful place to ... | [] | Buffet | Banashankari |
| **1** | https://www.zomato.com/bangalore/spice-elephan... | 2nd Floor, 80 Feet Road, Near Big Bazaar, | Spice Elephant | Yes | No | 4.1/5 | 787 | 080 41714161 | Banashankari | Casual Dining | Momos, Lunch Buffet, Chocolate Nirva | Chinese, North Indian, Thai | 800 | [('Rated 4.0', 'RATED\n Had been here | [] | Buffet | Banashankari |

| | 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) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 6th ... | | | | | | | | | na, Thai G... | | | for din ... | | | |
| **2** | https://www.zomato.com/SanchurroBangalore?cont... | 1112, Next to KIMS Medical College, 17th Cross... | SanChurro Cafe | Yes | No | 3.8/5 | 918 | +919663487993 | Banashankari | Cafe, Casual Dining | Churros, Cannelloni, Minestrone Soup, Hot Choc... | Cafe, Mexican, Italian | 800 | [('Rated 3.0', "RATED \n Ambience is not that ... | [] | Buffet | Banashankari |
| **3** | https://www.zomato.com/bangalore/addhuri-udupi... | 1st Floor, Annakutera, 3rd Stage, Banashankar... | Addhuri Udupi Bhojana | No | No | 3.7/5 | 88 | +919620009302 | Banashankari | Quick Bites | Masala Dosa | South Indian, North Indian | 300 | [('Rated 4.0', "RATED \n Great food and proper ... | [] | Buffet | Banashankari |

| | 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) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **4** | https://www.zomato.com/bangalore/grand-village... | 10, 3rd Floor, Lakshmi Associates, Gandhi Baza... | Grand Village | No | No | 3.8/5 | 166 | +918026612447\r\n+919901210005 | Basavanagudi | Casual Dining | Panipuri, Gol Gappe | North Indian, Rajasthani | 600 | [('Rated 4.0', 'RATED\n Very good restaurant ... | [] | Buffet | Banashankari |

In [7]:

```
df =
df.drop(['url','address','phone','menu_item','dish_liked','reviews_list'],
axis = 1)
df.head()
```

Out[7]:

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | approx_cost(for two people) | listed_in(type) | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1/5 | 775 | Banashankari | Casual Dining | North Indian, Mughlai, Chinese | 800 | Buffet | Banashankari |
| **1** | Spice Elephant | Yes | No | 4.1/5 | 787 | Banashankari | Casual Dining | Chinese, North Indian, Thai | 800 | Buffet | Banashankari |

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | approx_cost(for two people) | listed_in(type) | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **2** | San Churro Cafe | Yes | No | 3.8/5 | 918 | Banashankari | Cafe, Casual Dining | Cafe, Mexican, Italian | 800 | Buffet | Banashankari |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7/5 | 88 | Banashankari | Quick Bites | South Indian, North Indian | 300 | Buffet | Banashankari |
| **4** | Grand Village | No | No | 3.8/5 | 166 | Basavanagudi | Casual Dining | North Indian, Rajasthani | 600 | Buffet | Banashankari |

In [8]:

```
df.drop_duplicates(inplace = True)
df.shape
```

Out[8]:

```
(51609, 11)
```

In [9]:

```
def handlerate(value):
    if(value=='NEW' or value=='-'):
        return np.nan
    else:
        value = str(value).split('/')
        value = value[0]
        return float(value)

df['rate'] = df['rate'].apply(handlerate)
df['rate'].head()
```

Out[9]:

```
0    4.1
1    4.1
2    3.8
3    3.7
4    3.8
Name: rate, dtype: float64
```

In [10]:

```
df.rate.isnull().sum()
```

Out[10]:

```
10019
```

In [11]:

```
df['rate'].fillna(df['rate'].mean(),inplace = True)
```

```
df['rate'].isnull().sum()
```

Out[11]:

```
0
```

In [12]:

```
df.dropna(inplace = True)
df.head()
```

Out[12]:

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | approx_cost(for two people) | listed_in(type) | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | Banashankari | Casual Dining | North Indian, Mughlai, Chinese | 800 | Buffet | Banashankari |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | Banashankari | Casual Dining | Chinese, North Indian, Thai | 800 | Buffet | Banashankari |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | Banashankari | Cafe, Casual Dining | Cafe, Mexican, Italian | 800 | Buffet | Banashankari |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | Banashankari | Quick Bites | South Indian, North Indian | 300 | Buffet | Banashankari |
| **4** | Grand Village | No | No | 3.8 | 166 | Basavanagudi | Casual Dining | North Indian, Rajasthani | 600 | Buffet | Banashankari |

In [13]:

```
df.dropna(inplace = True)
df.head()
```

Out[13]:

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | approx_cost(for two people) | listed_in(type) | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | Banashankari | Casual Dining | North Indian, Mughlai, Chinese | 800 | Buffet | Banashankari |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | Banashankari | Casual Dining | Chinese, North Indian, Thai | 800 | Buffet | Banashankari |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | Banashankari | Cafe, Casual Dining | Cafe, Mexican, Italian | 800 | Buffet | Banashankari |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | Banashankari | Quick Bites | South Indian, North Indian | 300 | Buffet | Banashankari |
| **4** | Grand Village | No | No | 3.8 | 166 | Basavanagudi | Casual Dining | North Indian, Rajasthani | 600 | Buffet | Banashankari |

In [14]:

```
df.rename(columns={'approx_cost(for two people)': 'Cost2plate',
'listed_in(type)':'Type'}, inplace = True)
df.head(3)
```

Out[14]:

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | Cost2plate | Type | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | Banashankari | Casual Dining | North Indian, Mughlai, Chinese | 800 | Buffet | Banashankari |

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | Cost2plate | Type | listed_in(city) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Spice Elephant | Yes | No | 4.1 | 787 | Banashankari | Casual Dining | Chinese, North Indian, Thai | 800 | Buffet | Banashankari |
| 2 | San Churro Cafe | Yes | No | 3.8 | 918 | Banashankari | Cafe, Casual Dining | Cafe, Mexican, Italian | 800 | Buffet | Banashankari |

In [15]:

```
def handlecomma(value):
    value = str(value)
    if ',' in value:
        value = value.replace(',','')
        return float(value)
    else:
        return float(value)

df['Cost2plate'] = df['Cost2plate'].apply(handlecomma)
df['Cost2plate'].unique()
```

Out[15]:

```
array([ 800.,  300.,  600.,  700.,  550.,  500.,  450.,  650.,  400.,
        900.,  200.,  750.,  150.,  850.,  100., 1200.,  350.,  250.,
        950., 1000., 1500., 1300.,  199.,   80., 1100.,  160., 1600.,
        230.,  130.,   50.,  190., 1700., 1400.,  180., 1350., 2200.,
       2000., 1800., 1900.,  330., 2500., 2100., 3000., 2800., 3400.,
         40., 1250., 3500., 4000., 2400., 2600.,  120., 1450.,  469.,
         70., 3200.,   60.,  560.,  240.,  360., 6000., 1050., 2300.,
       4100., 5000., 3700., 1650., 2700., 4500.,  140.])
```

In [16]:

```
rest_types = df['rest_type'].value_counts(ascending = False)
rest_types
```

Out[16]:

```
Quick Bites                   19010
Casual Dining                 10253
Cafe                           3682
Delivery                       2574
Dessert Parlor                 2242
                              ...
Dessert Parlor, Kiosk             2
Food Court, Beverage Shop         2
Dessert Parlor, Food Court        2
Quick Bites, Kiosk                1
Sweet Shop, Dessert Parlor        1
Name: rest_type, Length: 93, dtype: int64
```

In [17]:

```
rest_types_lessthan1000 = rest_types[rest_types<1000]
rest_types_lessthan1000
```

Out[17]:

```
Beverage Shop                 863
Bar                           686
Food Court                    616
Sweet Shop                    468
Bar, Casual Dining            411
                             ...
Dessert Parlor, Kiosk           2
Food Court, Beverage Shop       2
Dessert Parlor, Food Court      2
Quick Bites, Kiosk              1
Sweet Shop, Dessert Parlor      1
Name: rest_type, Length: 85, dtype: int64
```

In [18]:

```
def handle_rest_type(value):
    if(value in rest_types_lessthan1000):
        return 'others'
    else:
        return value

df['rset_type'] = df['rest_type'].apply(handle_rest_type)
df['rest_type'].value_counts()
```

Out[18]:

```
Quick Bites                   19010
Casual Dining                 10253
Cafe                           3682
Delivery                       2574
Dessert Parlor                 2242
                              ...
Dessert Parlor, Kiosk             2
Food Court, Beverage Shop         2
Dessert Parlor, Food Court        2
Quick Bites, Kiosk                1
Sweet Shop, Dessert Parlor        1
Name: rest_type, Length: 93, dtype: int64
```

In [20]:

```
location = df['location'].value_counts(ascending = False)
location_lessthan300 = location[location<300]

def handle_location(value):
    if(value in location_lessthan300):
        return 'others'
    else:
        return value

df['location'] = df['location'].apply(handle_location)
df['location'].value_counts()
```

Out[20]:

```
BTM                    5056
others                 4954
HSR                    2494
Koramangala 5th Block  2479
JP Nagar               2218
Whitefield             2105
Indiranagar            2026
Jayanagar              1916
Marathahalli           1805
Bannerghatta Road      1609
Bellandur              1268
Electronic City        1246
Koramangala 1st Block  1236
Brigade Road           1210
Koramangala 7th Block  1174
Koramangala 6th Block  1127
Sarjapur Road          1047
Koramangala 4th Block  1017
Ulsoor                 1011
Banashankari            902
MG Road                 893
Kalyan Nagar            841
Richmond Road           803
Malleshwaram            721
Frazer Town             714
Basavanagudi            684
Residency Road          671
Brookefield             656
New BEL Road            644
Banaswadi               640
Kammanahalli            639
Rajajinagar             591
Church Street           566
Lavelle Road            518
Shanti Nagar            508
Shivajinagar            498
Cunningham Road         490
Domlur                  482
Old Airport Road        437
Ejipura                 433
Commercial Street       370
St. Marks Road          343
Name: location, dtype: int64
```

In [21]:

```
cuisines = df['cuisines'].value_counts(ascending = False)
cuisines_lessthan100 = cuisines[cuisines<100]

def handle_cuisines(value):
    if(value in cuisines_lessthan100):
        return 'others'
    else:
```

```
        return value

df['cuisines'] = df['cuisines'].apply(handle_cuisines)
df['cuisines'].value_counts()
```

Out[21]:

```
others                                  26159
North Indian                             2852
North Indian, Chinese                    2351
South Indian                             1820
Biryani                                   903
                                         ...
South Indian, Chinese, North Indian       105
North Indian, Mughlai, Chinese            104
South Indian, Fast Food                   104
Italian, Pizza                            102
North Indian, Chinese, Seafood            102
Name: cuisines, Length: 70, dtype: int64
```

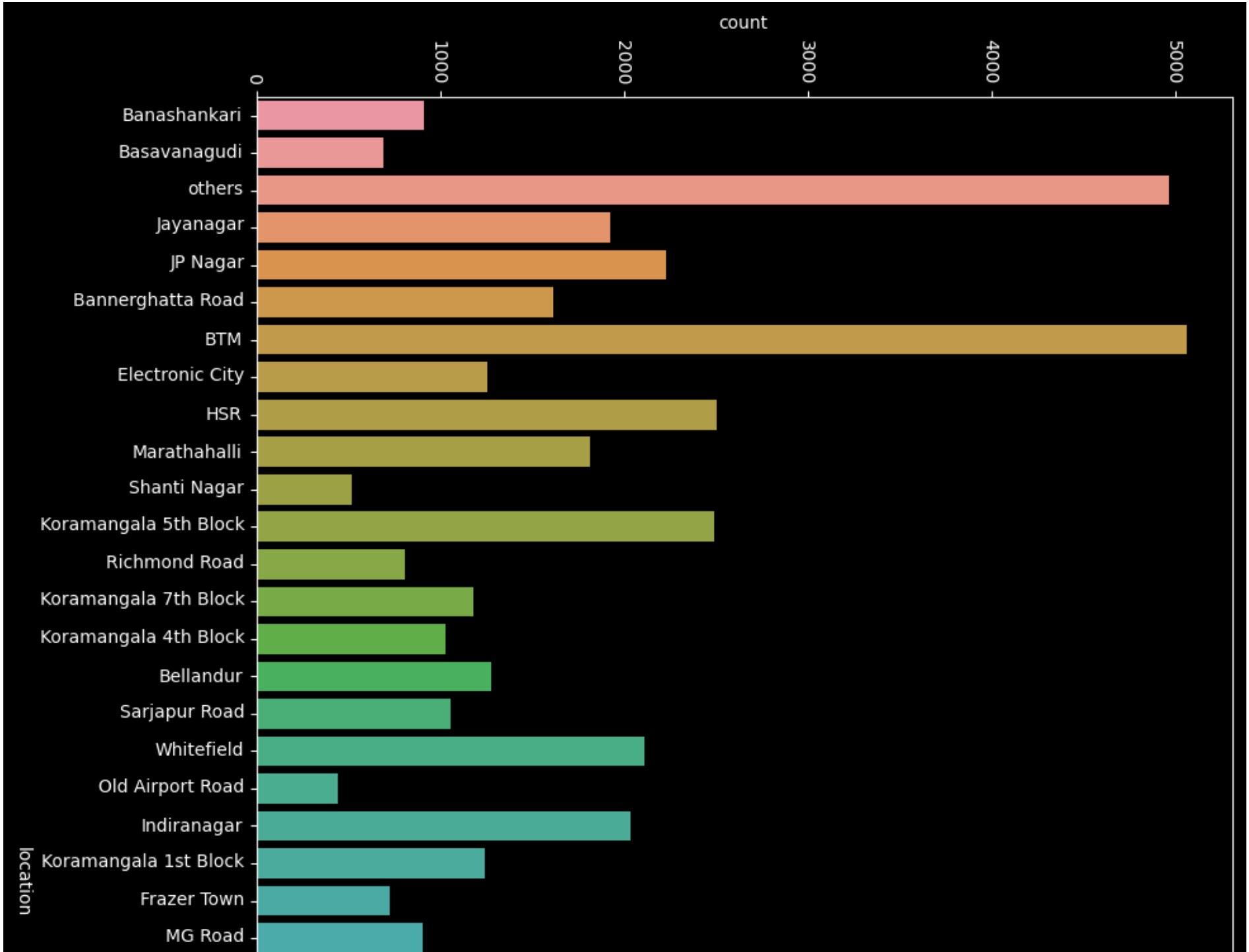
In [22]:

```
plt.figure(figsize = (16,10))
ax = sns.countplot(data=df, x='location')
plt.xticks(rotation = 90)
```

Out[22]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41]),
 [Text(0, 0, 'Banashankari'),
  Text(1, 0, 'Basavanagudi'),
  Text(2, 0, 'others'),
  Text(3, 0, 'Jayanagar'),
  Text(4, 0, 'JP Nagar'),
  Text(5, 0, 'Bannerghatta Road'),
  Text(6, 0, 'BTM'),
  Text(7, 0, 'Electronic City'),
  Text(8, 0, 'HSR'),
  Text(9, 0, 'Marathahalli'),
  Text(10, 0, 'Shanti Nagar'),
  Text(11, 0, 'Koramangala 5th Block'),
  Text(12, 0, 'Richmond Road'),
  Text(13, 0, 'Koramangala 7th Block'),
  Text(14, 0, 'Koramangala 4th Block'),
  Text(15, 0, 'Bellandur'),
  Text(16, 0, 'Sarjapur Road'),
  Text(17, 0, 'Whitefield'),
  Text(18, 0, 'Old Airport Road'),
  Text(19, 0, 'Indiranagar'),
  Text(20, 0, 'Koramangala 1st Block'),
  Text(21, 0, 'Frazer Town'),
  Text(22, 0, 'MG Road'),
  Text(23, 0, 'Brigade Road'),
  Text(24, 0, 'Lavelle Road'),
```

```
Text(25, 0, 'Church Street'),
Text(26, 0, 'Ulsoor'),
Text(27, 0, 'Residency Road'),
Text(28, 0, 'Shivajinagar'),
Text(29, 0, 'St. Marks Road'),
Text(30, 0, 'Cunningham Road'),
Text(31, 0, 'Commercial Street'),
Text(32, 0, 'Domlur'),
Text(33, 0, 'Ejipura'),
Text(34, 0, 'Malleshwaram'),
Text(35, 0, 'Kammanahalli'),
Text(36, 0, 'Koramangala 6th Block'),
Text(37, 0, 'Brookefield'),
Text(38, 0, 'Rajajinagar'),
Text(39, 0, 'Banaswadi'),
Text(40, 0, 'Kalyan Nagar'),
Text(41, 0, 'New BEL Road')])
```

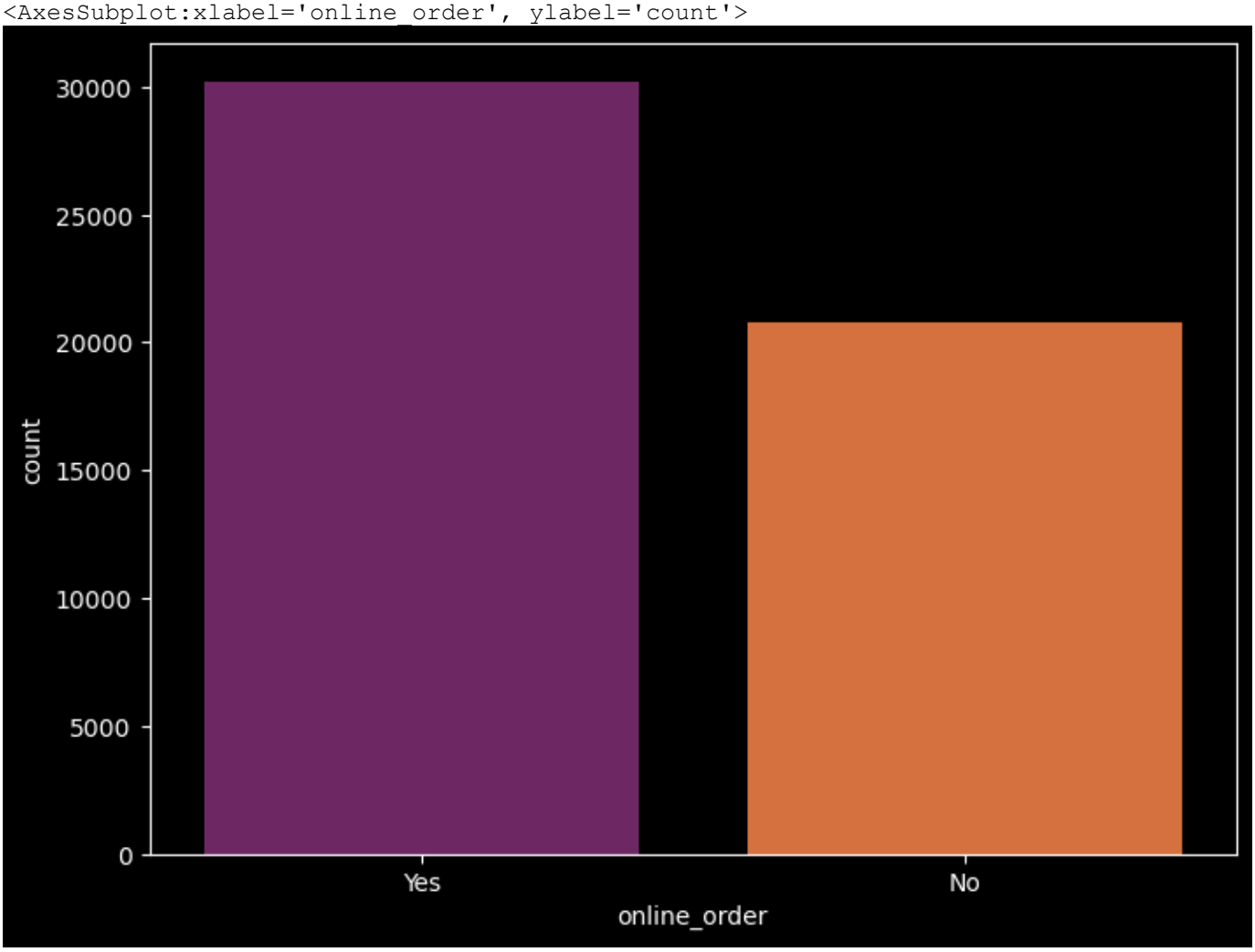

count
0
1000
2000
3000
4000
5000
location
Banashankari
Basavanagudi
others
Jayanagar
JP Nagar
Bannerghatta Road
BTM
Electronic City
HSR
Marathahalli
Shanti Nagar
Koramangala 5th Block
Richmond Road
Koramangala 7th Block
Koramangala 4th Block
Bellandur
Sarjapur Road
Whitefield
Old Airport Road
Indiranagar
Koramangala 1st Block
Frazer Town
MG Road

In [23]:

```
plt.figure(figsize=(8, 6))
sns.countplot(data=df, x='online_order', palette='inferno')
```

Out[23]:

```
<AxesSubplot:xlabel='online_order', ylabel='count'>
```

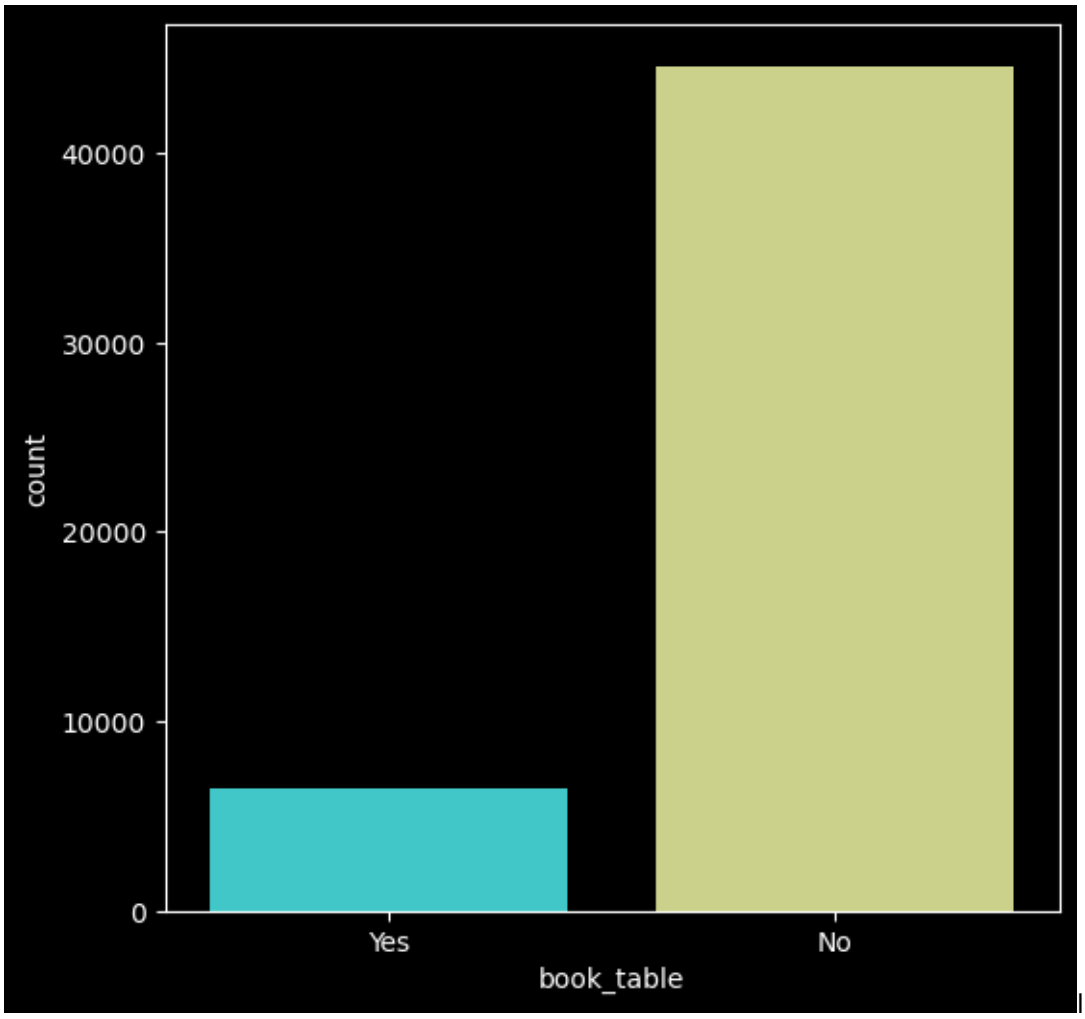



In [24]:

```
plt.figure(figsize = (6,6))
sns.countplot(data = df, x ='book_table', palette = 'rainbow')
```

Out[24]:

```
<AxesSubplot:xlabel='book_table', ylabel='count'>
```

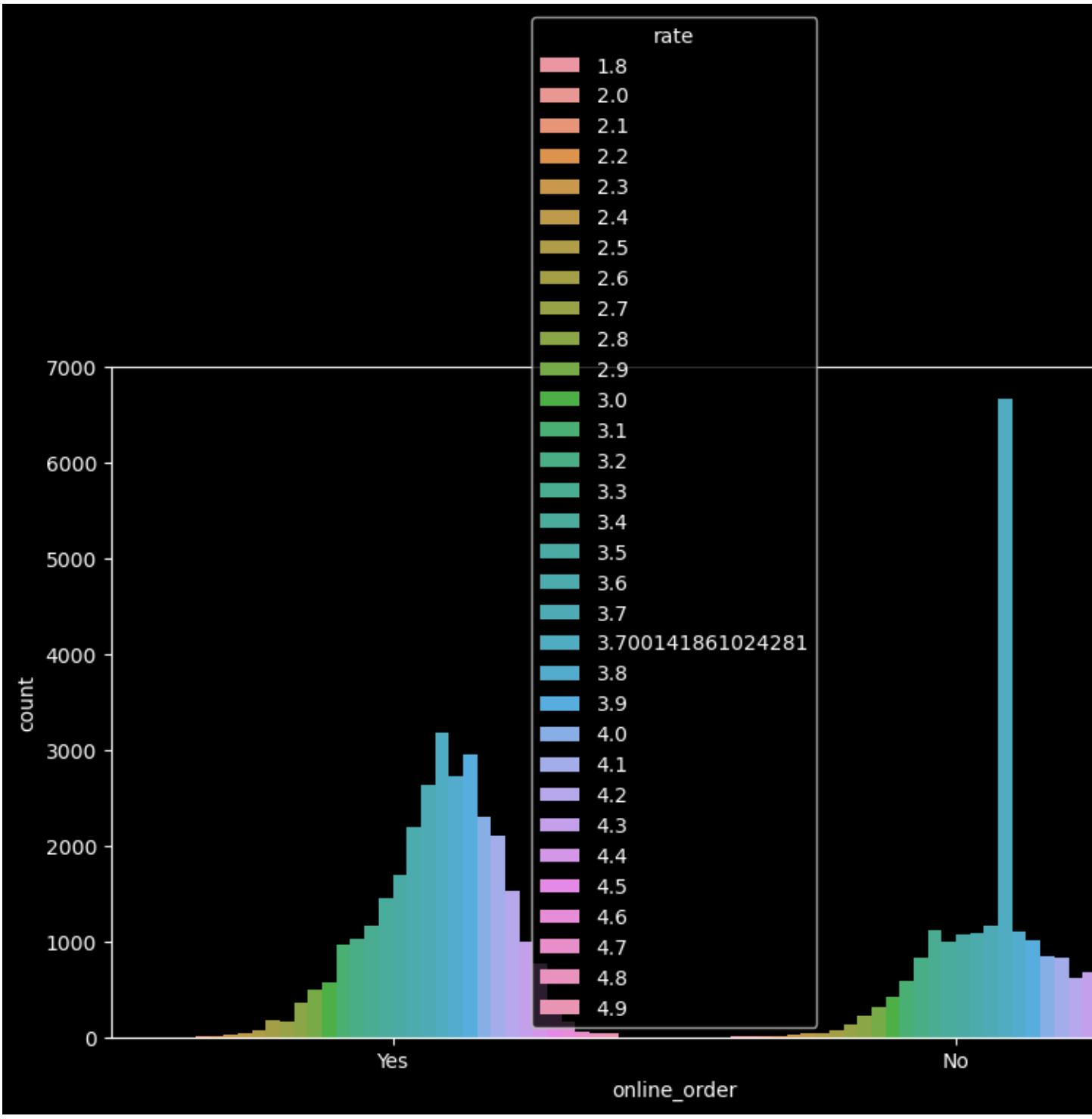



In [27]:

```
plt.figure(figsize=(10,6))
sns.countplot(x='online_order', hue='rate', data=df)
```

Out[27]:

```
<AxesSubplot:xlabel='online_order', ylabel='count'>
```

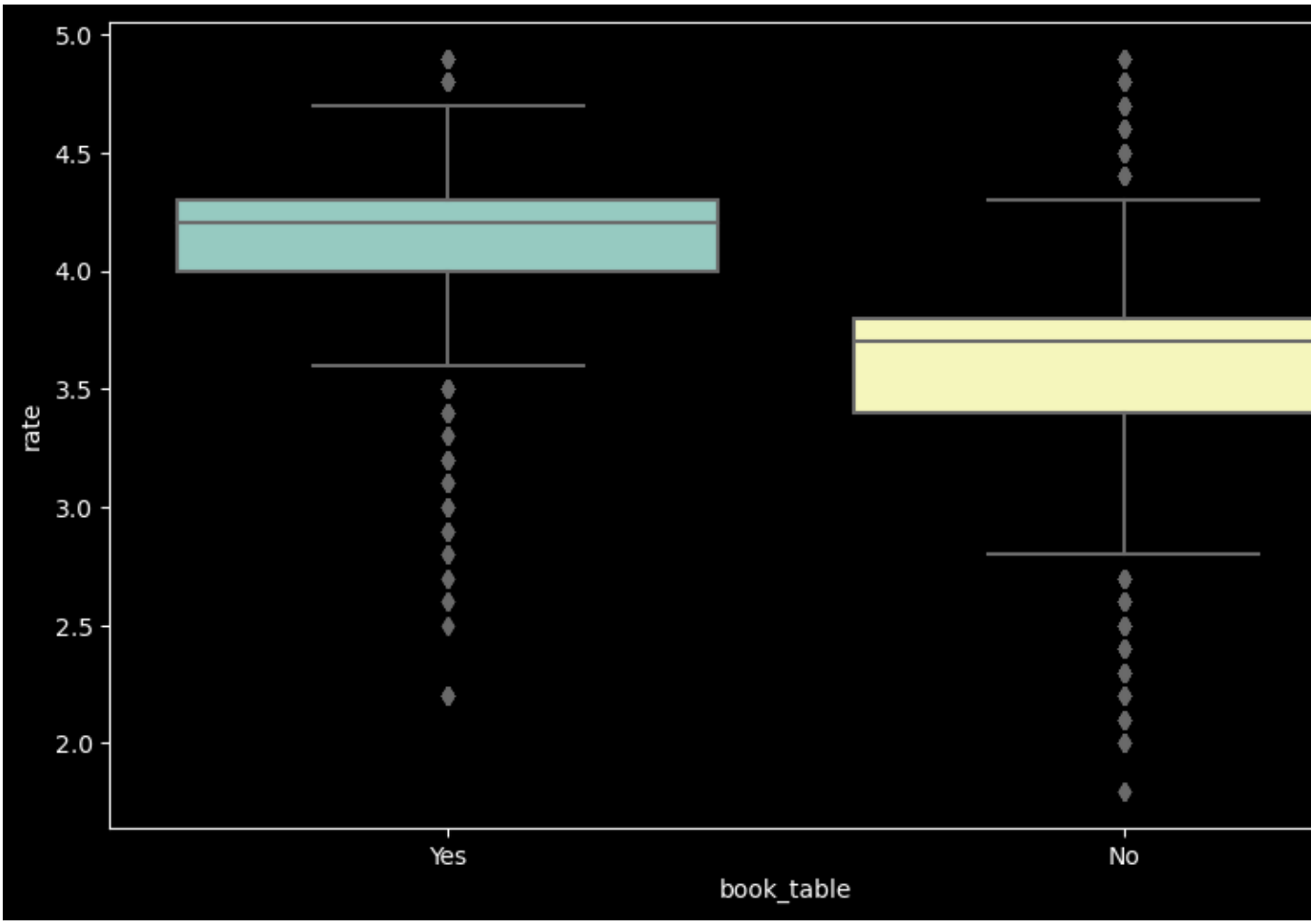



In [29]:

```
plt.figure(figsize = (10,6))
sns.boxplot(x='book_table',y = 'rate', data=df)
```

Out[29]:

```
<AxesSubplot:xlabel='book_table', ylabel='rate'>
```




In [30]:

```
df1 = df.groupby(['location', 'online_order'])['name'].count()
df1.to_csv('location_online.csv')
df1 = pd.read_csv('location_online.csv')
df1 = pd.pivot_table(df1, values=None,index = ['location'], columns =
['online_order'], fill_value = 0, aggfunc = np.sum)
df1
```

Out[30]:

| | name | |
|---|---|---|
| **online_order** | **No** | **Yes** |
| **location** | | |
| **BTM** | 1763 | 3293 |

| | name | |
|---|---|---|
| online_order | No | Yes |
| location | | |
| Banashankari | 397 | 505 |
| Banaswadi | 302 | 338 |
| Bannerghatta Road | 685 | 924 |
| Basavanagudi | 243 | 441 |
| Bellandur | 517 | 751 |
| Brigade Road | 552 | 658 |
| Brookefield | 239 | 417 |
| Church Street | 226 | 340 |
| Commercial Street | 228 | 142 |
| Cunningham Road | 168 | 322 |
| Domlur | 247 | 235 |
| Ejipura | 214 | 219 |
| Electronic City | 676 | 570 |
| Frazer Town | 287 | 427 |
| HSR | 584 | 1910 |

| name | | |
|---|---|---|
| online_order | No | Yes |
| location | | |
| Indiranagar | 697 | 1329 |
| JP Nagar | 911 | 1307 |
| Jayanagar | 552 | 1364 |
| Kalyan Nagar | 350 | 491 |
| Kammanahalli | 264 | 375 |
| Koramangala 1st Block | 384 | 852 |
| Koramangala 4th Block | 459 | 558 |
| Koramangala 5th Block | 866 | 1613 |
| Koramangala 6th Block | 445 | 682 |
| Koramangala 7th Block | 389 | 785 |
| Lavelle Road | 315 | 203 |
| MG Road | 520 | 373 |
| Malleshwaram | 309 | 412 |
| Marathahalli | 701 | 1104 |
| New BEL Road | 255 | 389 |

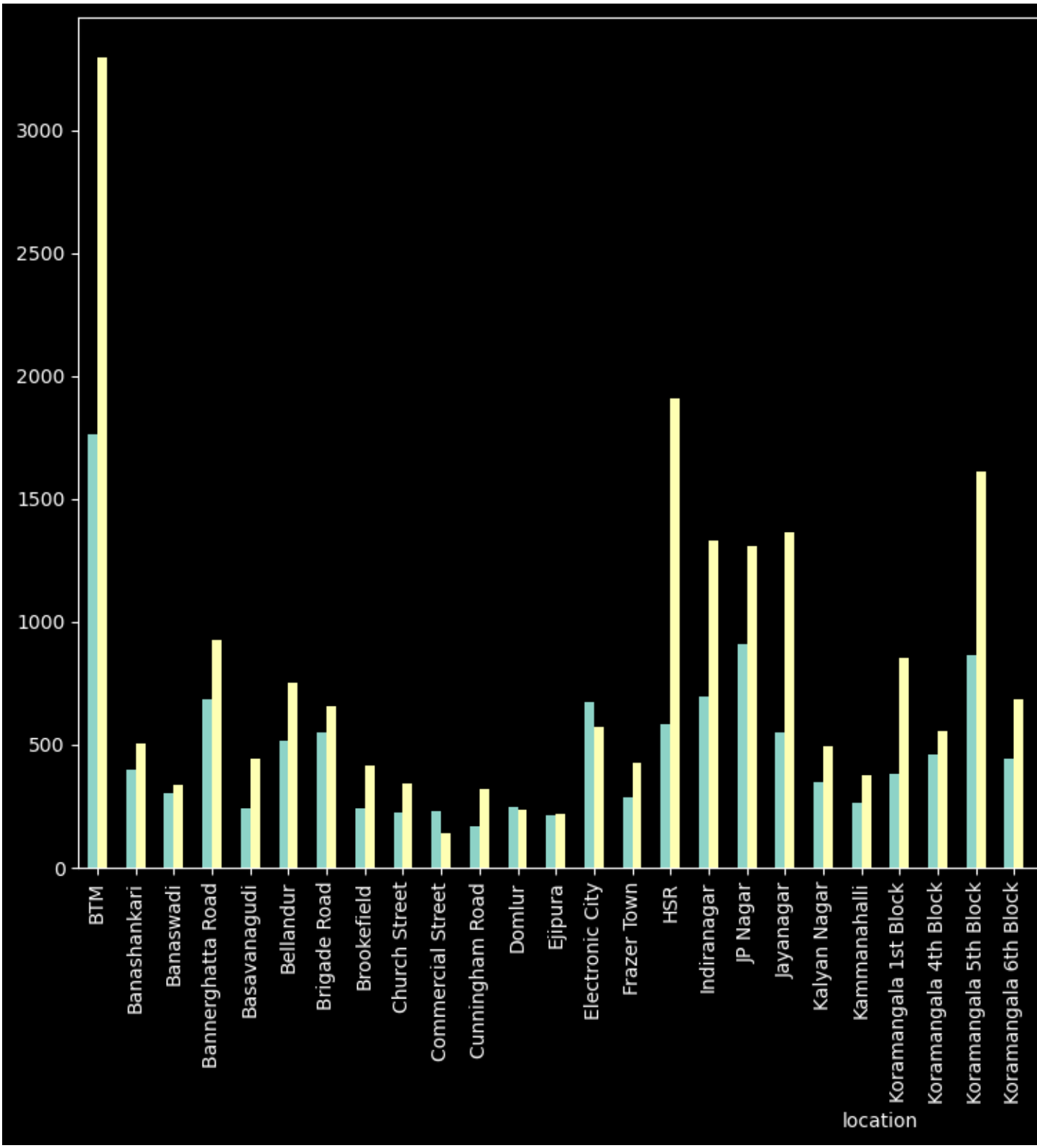
| | name | |
|---|---|---|
| online_order | No | Yes |
| location | | |
| **Old Airport Road** | 221 | 216 |
| **Rajajinagar** | 286 | 305 |
| **Residency Road** | 424 | 247 |
| **Richmond Road** | 557 | 246 |
| **Sarjapur Road** | 323 | 724 |
| **Shanti Nagar** | 289 | 219 |
| **Shivajinagar** | 354 | 144 |
| **St. Marks Road** | 176 | 167 |
| **Ulsoor** | 389 | 622 |
| **Whitefield** | 986 | 1119 |
| **others** | 2064 | 2890 |

In [31]:

```
df1.plot(kind = 'bar',figsize = (15,8))
```

Out[31]:

```
<AxesSubplot:xlabel='location'>
```



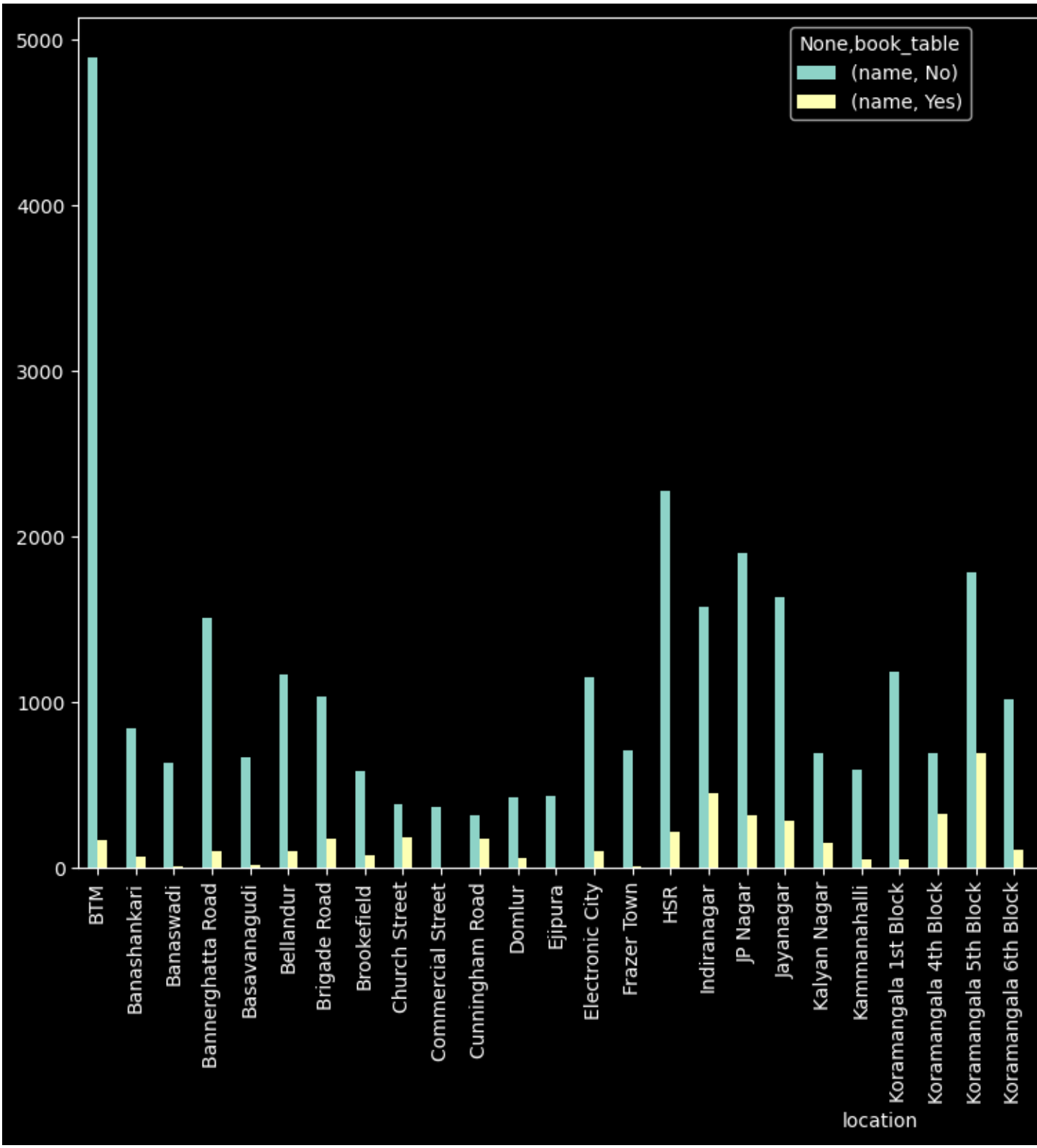

In [32]:

```
df2 = df.groupby(['location','book_table'])['name'].count()
df2.to_csv('location_booktable.csv')
df2 = pd.read_csv('location_booktable.csv')
```

```
df2 = pd.pivot_table(df2, values = None,index = ['location'], columns =
['book_table'], fill_value= 0, aggfunc = np.sum)
df2
df2.plot(kind = 'bar', figsize = (15,8))
```

Out[32]:

```
<AxesSubplot:xlabel='location'>
```

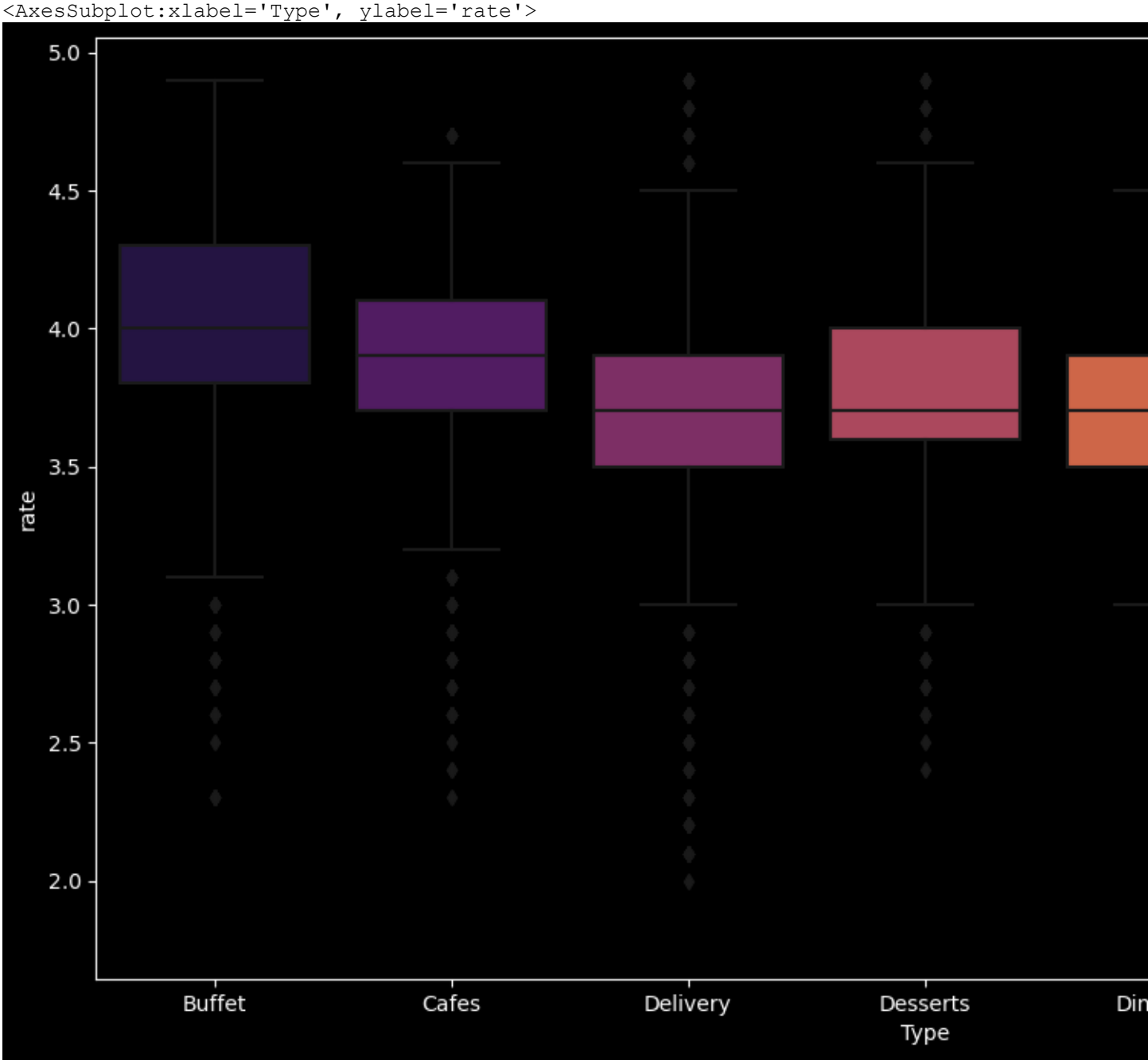



In [33]:

```
plt.figure(figsize = (14,8))
sns.boxplot(x = 'Type', y = 'rate', data = df, palette = 'inferno')
```

Out[33]:

```
<AxesSubplot:xlabel='Type', ylabel='rate'>
```

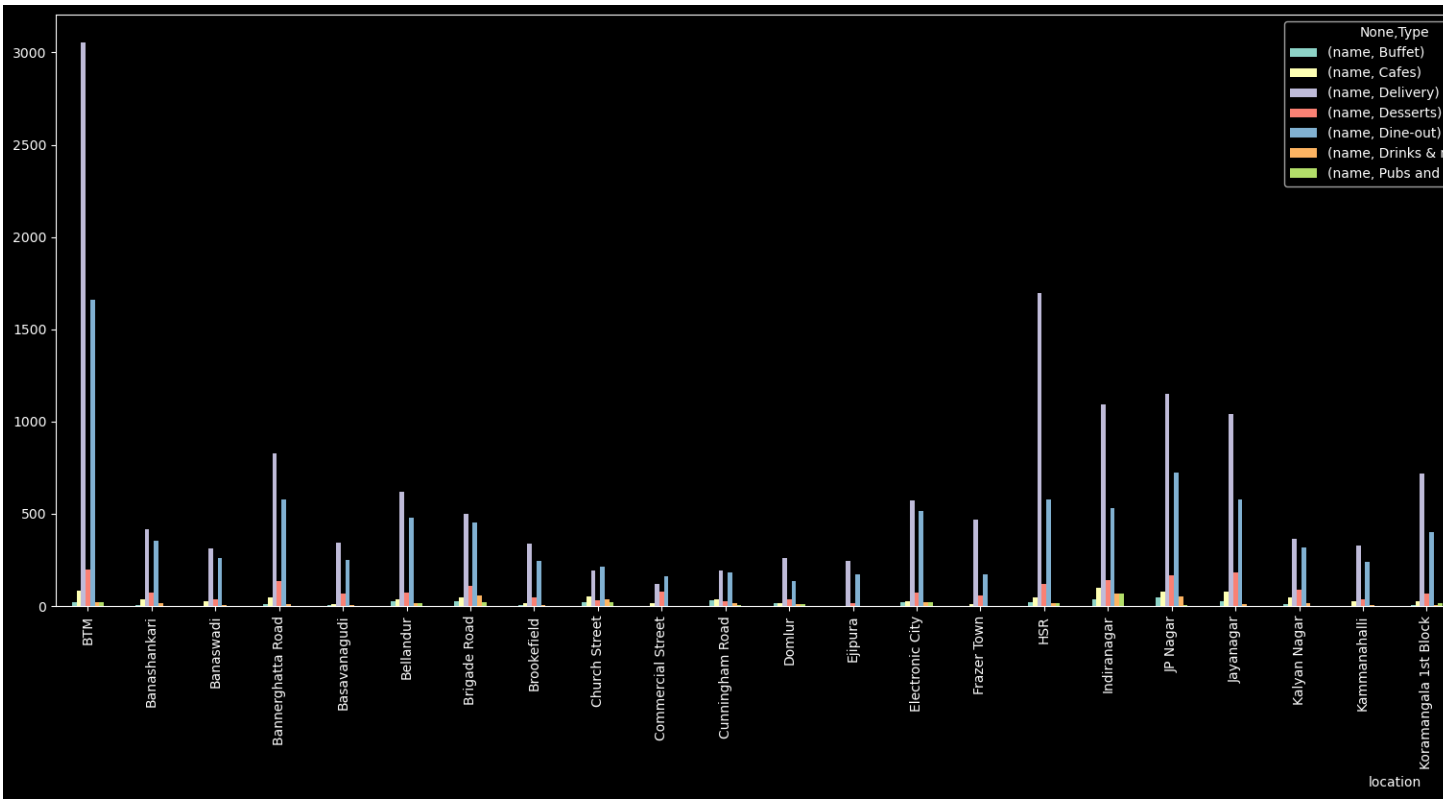



In [34]:

```
df3 = df.groupby(['location','Type'])['name'].count()
df3.to_csv('location_Type.csv')
df3 = pd.read_csv('location_Type.csv')
df3 = pd.pivot_table(df3, values = None,index = ['location'], columns =
['Type'], fill_value= 0, aggfunc = np.sum)
df3
df3.plot(kind = 'bar', figsize = (36,8))
```

Out[34]:

```
<AxesSubplot:xlabel='location'>
```




In [35]:

```
df4 = df[['location','votes']]
df4.drop_duplicates()
df5 = df4.groupby(['location'])['votes'].sum()
df5 = df5.to_frame()
df5 = df5.sort_values('votes',ascending = False)
df5.head()
```

Out[35]:

| location | votes |
|---|---|
| **Koramangala 5th Block** | 2214083 |
| **Indiranagar** | 1165909 |
| **Koramangala 4th Block** | 685156 |
| **Church Street** | 590306 |
| **JP Nagar** | 586522 |

In [36]:

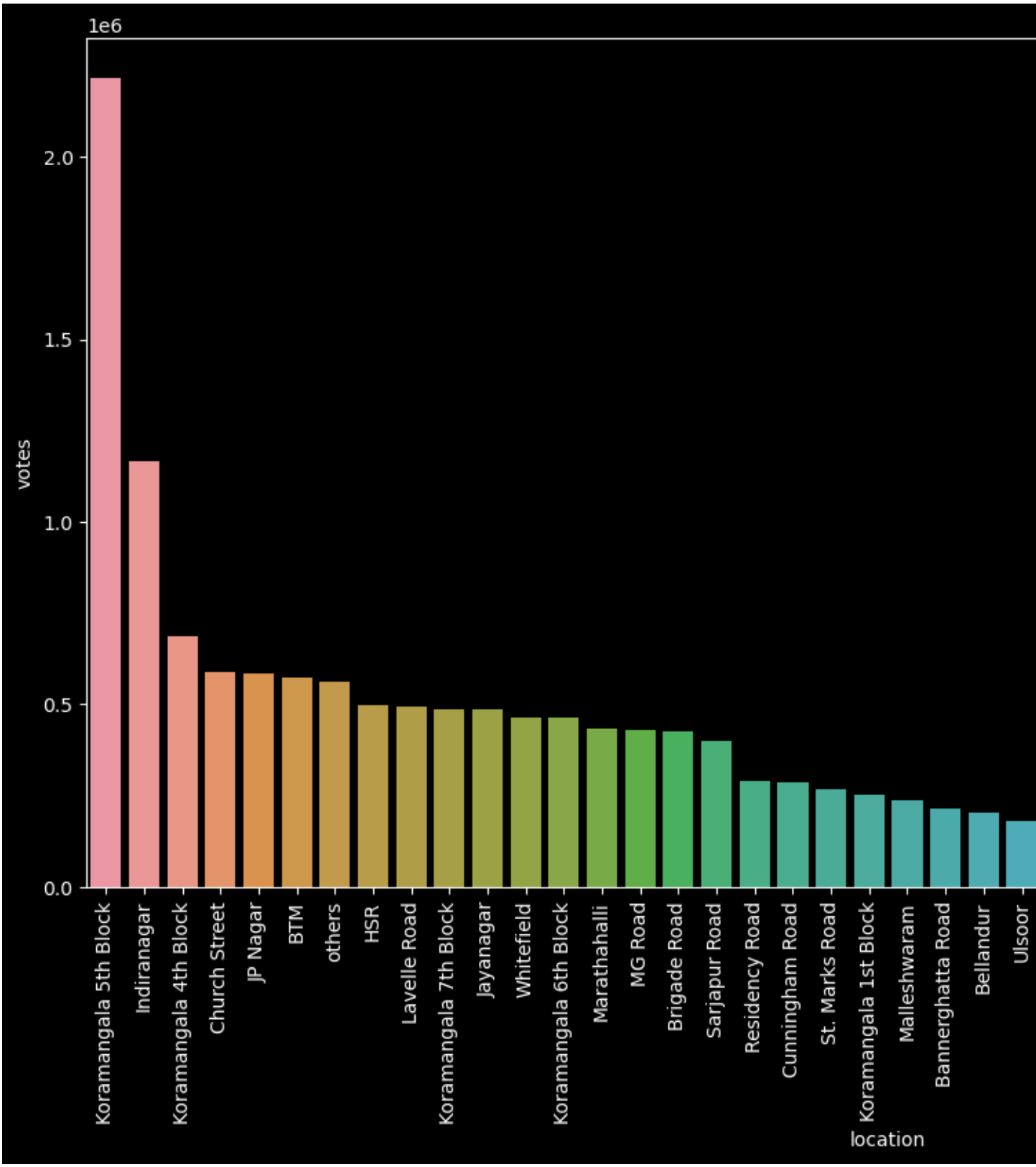
```
plt.figure(figsize = (15,8))
sns.barplot(df5.index,df5['votes'])
plt.xticks(rotation = 90)
```

```
C:\Users\HOSHANGI\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futu
reWarning: Pass the following variables as keyword args: x, y. From version 0
.12, the only valid positional argument will be `data`, and passing other arg
uments without an explicit keyword will result in an error or misinterpretati
on.
  warnings.warn(
```

Out[36]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41]),
 [Text(0, 0, 'Koramangala 5th Block'),
  Text(1, 0, 'Indiranagar'),
  Text(2, 0, 'Koramangala 4th Block'),
  Text(3, 0, 'Church Street'),
  Text(4, 0, 'JP Nagar'),
  Text(5, 0, 'BTM'),
  Text(6, 0, 'others'),
  Text(7, 0, 'HSR'),
  Text(8, 0, 'Lavelle Road'),
  Text(9, 0, 'Koramangala 7th Block'),
  Text(10, 0, 'Jayanagar'),
  Text(11, 0, 'Whitefield'),
  Text(12, 0, 'Koramangala 6th Block'),
  Text(13, 0, 'Marathahalli'),
  Text(14, 0, 'MG Road'),
  Text(15, 0, 'Brigade Road'),
  Text(16, 0, 'Sarjapur Road'),
  Text(17, 0, 'Residency Road'),
  Text(18, 0, 'Cunningham Road'),
  Text(19, 0, 'St. Marks Road'),
  Text(20, 0, 'Koramangala 1st Block'),
  Text(21, 0, 'Malleshwaram'),
  Text(22, 0, 'Bannerghatta Road'),
  Text(23, 0, 'Bellandur'),
  Text(24, 0, 'Ulsoor'),
  Text(25, 0, 'New BEL Road'),
  Text(26, 0, 'Kalyan Nagar'),
  Text(27, 0, 'Banashankari'),
  Text(28, 0, 'Old Airport Road'),
  Text(29, 0, 'Brookefield'),
  Text(30, 0, 'Richmond Road'),
  Text(31, 0, 'Electronic City'),
  Text(32, 0, 'Kammanahalli'),
  Text(33, 0, 'Frazer Town'),
  Text(34, 0, 'Domlur'),
  Text(35, 0, 'Basavanagudi'),
  Text(36, 0, 'Rajajinagar'),
  Text(37, 0, 'Shanti Nagar'),
```

```
Text(38, 0, 'Banaswadi'),
Text(39, 0, 'Commercial Street'),
Text(40, 0, 'Ejipura'),
Text(41, 0, 'Shivajinagar')])
```


1e6
2.0
1.5
1.0
0.5
0.0
votes
Koramangala 5th Block
Indiranagar
Koramangala 4th Block
Church Street
JP Nagar
BTM
others
HSR
Lavelle Road
Koramangala 7th Block
Jayanagar
Whitefield
Koramangala 6th Block
Marathahalli
MG Road
Brigade Road
Sarjapur Road
Residency Road
Cunningham Road
St. Marks Road
Koramangala 1st Block
Malleshwaram
Bannerghatta Road
Bellandur
Ulsoor
location

In [37]:

```
df.head()
```

Out[37]:

| | name | online_order | book_table | rate | votes | location | rest_type | cuisines | Cost2plate | Type | listed_in(city) | rset_type |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Jalsa | Yes | Yes | 4.1 | 775 | Banashankari | Casual Dining | North Indian, Mughlai, Chinese | 800.0 | Buffet | Banashankari | Casual Dining |
| **1** | Spice Elephant | Yes | No | 4.1 | 787 | Banashankari | Casual Dining | others | 800.0 | Buffet | Banashankari | Casual Dining |
| **2** | San Churro Cafe | Yes | No | 3.8 | 918 | Banashankari | Cafe, Casual Dining | others | 800.0 | Buffet | Banashankari | others |
| **3** | Addhuri Udupi Bhojana | No | No | 3.7 | 88 | Banashankari | Quick Bites | South Indian, North Indian | 300.0 | Buffet | Banashankari | Quick Bites |
| **4** | Grand Village | No | No | 3.8 | 166 | Basavanagudi | Casual Dining | others | 600.0 | Buffet | Banashankari | Casual Dining |

In [38]:

```
df6 = df[['cuisines','votes']]
df6.drop_duplicates()
df7 = df6.groupby(['cuisines'])['votes'].sum()
df7 = df7.to_frame()
df7 = df7.sort_values('votes',ascending = False)
df7.head()
```

Out[38]:

| cuisines | votes |
|---|---|
| **others** | 11542182 |

| cuisines | votes |
|---|---|
| **North Indian** | 516310 |
| **North Indian, Chinese** | 258225 |
| **South Indian** | 161975 |
| **North Indian, Mughlai** | 103706 |

In [39]:

```
df7 =df7.iloc[1:, :]
df7.head()
```

Out[39]:

| cuisines | votes |
|---|---|
| **North Indian** | 516310 |
| **North Indian, Chinese** | 258225 |
| **South Indian** | 161975 |
| **North Indian, Mughlai** | 103706 |
| **Chinese** | 101728 |

In [44]:

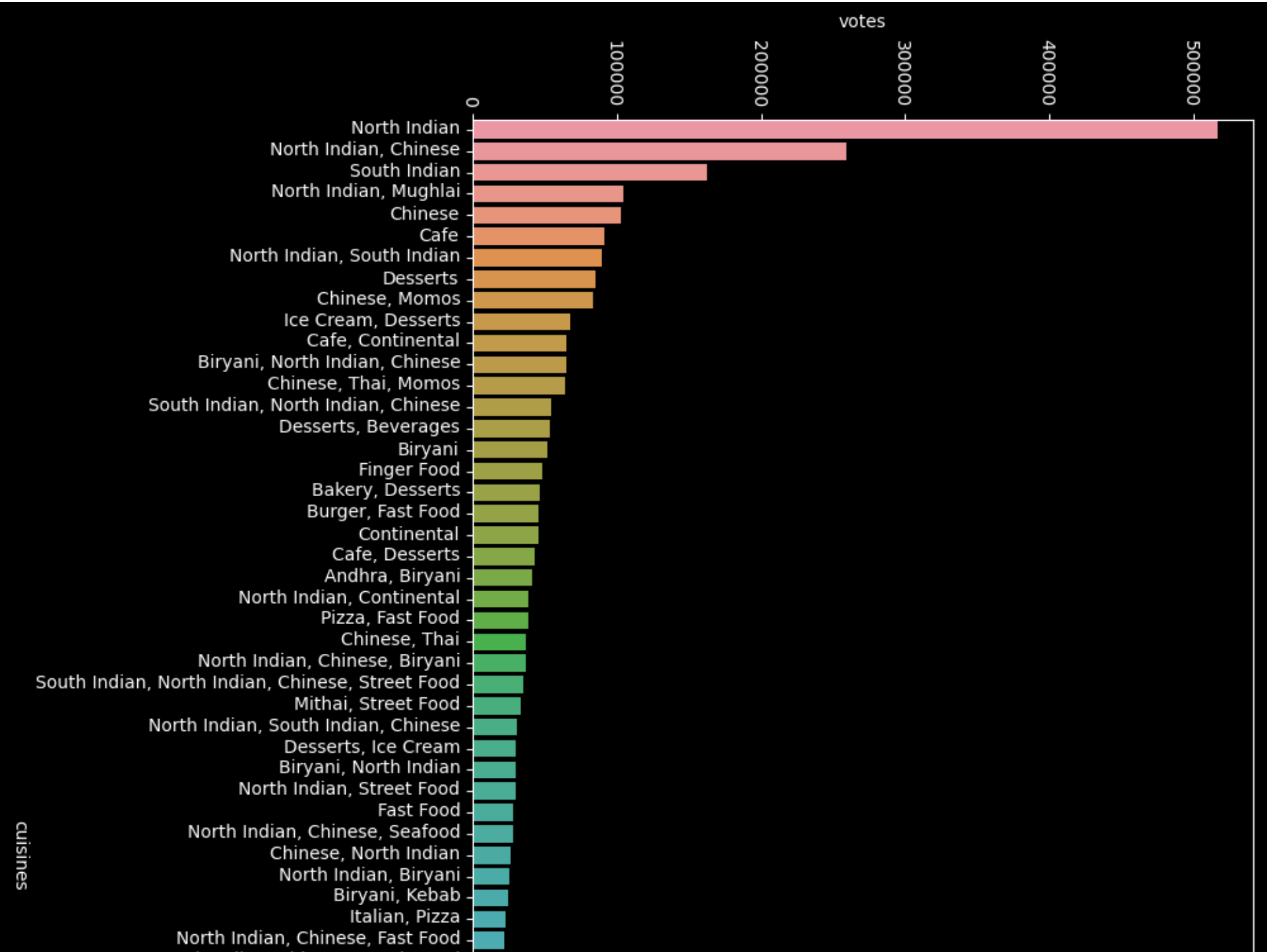
```
plt.figure(figsize = (15,8))
sns.barplot(df7.index, df7['votes'])
plt.xticks(rotation = 90)
```

```
C:\Users\HOSHANGI\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futu
reWarning: Pass the following variables as keyword args: x, y. From version 0
.12, the only valid positional argument will be `data`, and passing other arg
uments without an explicit keyword will result in an error or misinterpretati
on.
  warnings.warn(
```

Out[44]:

```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67,
        68]),
 [Text(0, 0, 'North Indian'),
  Text(1, 0, 'North Indian, Chinese'),
  Text(2, 0, 'South Indian'),
  Text(3, 0, 'North Indian, Mughlai'),
  Text(4, 0, 'Chinese'),
  Text(5, 0, 'Cafe'),
  Text(6, 0, 'North Indian, South Indian'),
  Text(7, 0, 'Desserts'),
  Text(8, 0, 'Chinese, Momos'),
  Text(9, 0, 'Ice Cream, Desserts'),
  Text(10, 0, 'Cafe, Continental'),
  Text(11, 0, 'Biryani, North Indian, Chinese'),
  Text(12, 0, 'Chinese, Thai, Momos'),
  Text(13, 0, 'South Indian, North Indian, Chinese'),
  Text(14, 0, 'Desserts, Beverages'),
  Text(15, 0, 'Biryani'),
  Text(16, 0, 'Finger Food'),
  Text(17, 0, 'Bakery, Desserts'),
  Text(18, 0, 'Burger, Fast Food'),
  Text(19, 0, 'Continental'),
  Text(20, 0, 'Cafe, Desserts'),
  Text(21, 0, 'Andhra, Biryani'),
  Text(22, 0, 'North Indian, Continental'),
  Text(23, 0, 'Pizza, Fast Food'),
  Text(24, 0, 'Chinese, Thai'),
  Text(25, 0, 'North Indian, Chinese, Biryani'),
  Text(26, 0, 'South Indian, North Indian, Chinese, Street Food'),
  Text(27, 0, 'Mithai, Street Food'),
  Text(28, 0, 'North Indian, South Indian, Chinese'),
  Text(29, 0, 'Desserts, Ice Cream'),
  Text(30, 0, 'Biryani, North Indian'),
  Text(31, 0, 'North Indian, Street Food'),
  Text(32, 0, 'Fast Food'),
  Text(33, 0, 'North Indian, Chinese, Seafood'),
  Text(34, 0, 'Chinese, North Indian'),
  Text(35, 0, 'North Indian, Biryani'),
  Text(36, 0, 'Biryani, Kebab'),
  Text(37, 0, 'Italian, Pizza'),
  Text(38, 0, 'North Indian, Chinese, Fast Food'),
  Text(39, 0, 'North Indian, Chinese, Continental'),
  Text(40, 0, 'Desserts, Bakery'),
  Text(41, 0, 'Arabian'),
  Text(42, 0, 'North Indian, Chinese, South Indian'),
  Text(43, 0, 'Pizza'),
  Text(44, 0, 'South Indian, North Indian'),
  Text(45, 0, 'North Indian, Mughlai, Chinese'),
  Text(46, 0, 'Beverages, Fast Food'),
  Text(47, 0, 'South Indian, Biryani'),
```

```
Text(48, 0, 'Fast Food, Rolls'),
Text(49, 0, 'Bakery'),
Text(50, 0, 'Andhra'),
Text(51, 0, 'Street Food'),
Text(52, 0, 'Kerala'),
Text(53, 0, 'Cafe, Bakery'),
Text(54, 0, 'Biryani, Fast Food'),
Text(55, 0, 'South Indian, Fast Food'),
Text(56, 0, 'North Indian, Fast Food'),
Text(57, 0, 'Bakery, Fast Food'),
Text(58, 0, 'Fast Food, Burger'),
Text(59, 0, 'Cafe, Fast Food'),
Text(60, 0, 'South Indian, Chinese'),
Text(61, 0, 'Beverages, Desserts'),
Text(62, 0, 'Ice Cream'),
Text(63, 0, 'Beverages'),
Text(64, 0, 'Biryani, South Indian'),
Text(65, 0, 'Kerala, South Indian'),
Text(66, 0, 'Fast Food, Beverages'),
Text(67, 0, 'South Indian, Chinese, North Indian'),
Text(68, 0, 'Mithai')])
```


votes
0
100000
200000
300000
400000
500000
North Indian
North Indian, Chinese
South Indian
North Indian, Mughlai
Chinese
Cafe
North Indian, South Indian
Desserts
Chinese, Momos
Ice Cream, Desserts
Cafe, Continental
Biryani, North Indian, Chinese
Chinese, Thai, Momos
South Indian, North Indian, Chinese
Desserts, Beverages
Biryani
Finger Food
Bakery, Desserts
Burger, Fast Food
Continental
Cafe, Desserts
Andhra, Biryani
North Indian, Continental
Pizza, Fast Food
Chinese, Thai
North Indian, Chinese, Biryani
South Indian, North Indian, Chinese, Street Food
Mithai, Street Food
North Indian, South Indian, Chinese
Desserts, Ice Cream
Biryani, North Indian
North Indian, Street Food
Fast Food
North Indian, Chinese, Seafood
Chinese, North Indian
North Indian, Biryani
Biryani, Kebab
Italian, Pizza
North Indian, Chinese, Fast Food
cuisines

In [ ]: