In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [2]:

```
pizza_data = pd.read_csv('pizza_data.csv')
```

In [3]:

```
pizza_data.head()
```

Out[3]:

	id	dateAdded	dateUpdated	address	categories	pri
0	AVz3Y-7h3D1zeR_xDAqm	2017-06-30T05:05:40Z	2019-05-01T15:43:09Z	4203 E Kiehl Ave	Pizza,Restaurant,American restaurants,Pizza Pl...	A
1	AVweGPFF_7pvs4fzAAzQ	2016-04-02T04:02:49Z	2019-05-01T15:27:50Z	25 E Camelback Rd	Pizza,Pizza Place,Restaurants	A
2	AVwdRGa9_7pvs4fz4E3K	2016-03-03T18:39:49Z	2019-05-01T12:52:25Z	3703 Paxton Ave	Restaurant,Pizza Place,Restaurants	A
3	AVwdX4psIN2L1WUfvJB1	2016-03-29T05:08:59Z	2019-05-01T12:52:20Z	30495 John R Rd	Pizza,Carry-out food,Pizza Place,Restaurants	A
4	AVwdaeTtkufWRAb55pSH	2016-03-31T02:34:04Z	2019-05-01T12:50:45Z	3600 Eastern Ave	Pizza,American restaurants,Pizza Place,Pizza e...	A

5 rows × 24 columns

In [4]:

```
pizza_data.tail()
```

Out[4]:

	id	dateAdded	dateUpdated	address	categories	prima
9995	AVwdKXFYByjofQCxmlyZ	2016-03-21T00:12:46Z	2018-01-08T10:25:27Z	1126 E Lovejoy St	Pizza Place	Acco F
9996	AVwdKXFYByjofQCxmlyZ	2016-03-21T00:12:46Z	2018-01-08T10:25:27Z	1126 E Lovejoy St	Pizza Place	Acco F
9997	AVz6kzsQFcQ3k02bDhwQ	2017-06-30T19:56:11Z	2018-01-07T10:51:29Z	3641 E Main St	Pizza Place,Restaurants	Acco F
9998	AVz6kzsQFcQ3k02bDhwQ	2017-06-30T19:56:11Z	2018-01-07T10:51:29Z	3641 E Main St	Pizza Place,Restaurants	Acco F
9999	AVwdG9zTkufWRAb52Y2A	2015-10-23T01:57:27Z	2018-01-04T17:57:59Z	7460 W Lake Mead Blvd	Pizza Place,Restaurant,Fast Food,Pizza	Acco F

5 rows × 24 columns

In [5]:

```
pizza_data.shape
```

Out[5]:

```
(10000, 24)
```

In [6]:

```
pizza_data.columns
```

Out[6]:

```
Index(['id', 'dateAdded', 'dateUpdated', 'address', 'categories',
       'primaryCategories', 'city', 'country', 'keys', 'latitude', 'longit
ude',
       'menuPageURL', 'menus.amountMax', 'menus.amountMin', 'menus.currenc
y',
       'menus.dateSeen', 'menus.description', 'menus.name', 'name',
       'postalCode', 'priceRangeCurrency', 'priceRangeMin', 'priceRangeMa
x',
       'province'],
      dtype='object')
```

In [7]:

```
pizza_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 24 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   id                  10000 non-null  object 
 1   dateAdded           10000 non-null  object 
 2   dateUpdated         10000 non-null  object 
 3   address             10000 non-null  object 
 4   categories          10000 non-null  object 
 5   primaryCategories   10000 non-null  object 
 6   city                10000 non-null  object 
 7   country             10000 non-null  object 
 8   keys                10000 non-null  object 
 9   latitude            10000 non-null  float64
 10  longitude           10000 non-null  float64
 11  menuPageURL         1679 non-null   object 
 12  menus.amountMax     10000 non-null  float64
 13  menus.amountMin     10000 non-null  float64
 14  menus.currency      10000 non-null  object 
 15  menus.dateSeen      10000 non-null  object 
 16  menus.description   3718 non-null   object 
 17  menus.name          10000 non-null  object 
 18  name                10000 non-null  object 
 19  postalCode          9996 non-null   object 
 20  priceRangeCurrency  10000 non-null  object 
 21  priceRangeMin       10000 non-null  int64  
 22  priceRangeMax       10000 non-null  int64  
 23  province            10000 non-null  object 
dtypes: float64(4), int64(2), object(18)
memory usage: 1.8+ MB
```

In [8]:

```
pizza_data.describe()
```

Out[8]:

	latitude	longitude	menus.amountMax	menus.amountMin	priceRangeMin	priceF
count	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	1000
mean	38.371895	-90.063800	14.032670	13.416902	4.655500	2
std	4.573537	17.340379	17.607233	10.742678	9.828412	
min	21.421326	-157.802443	0.000000	0.000000	0.000000	
25%	34.422946	-104.801126	8.490000	8.000000	0.000000	2
50%	40.118880	-82.909949	13.390000	12.990000	0.000000	2
75%	40.905048	-75.194741	17.950000	16.990000	0.000000	2
max	64.850500	-71.946170	1395.000000	243.000000	50.000000	5

In [9]:

```
pizza_data.isnull().sum()
```

Out[9]:

```
id                        0
dateAdded                 0
dateUpdated               0
address                   0
categories                0
primaryCategories         0
city                      0
country                   0
keys                      0
latitude                  0
longitude                 0
menuPageURL            8321
menus.amountMax           0
menus.amountMin           0
menus.currency            0
menus.dateSeen            0
menus.description      6282
menus.name                0
name                      0
postalCode                4
priceRangeCurrency        0
priceRangeMin             0
priceRangeMax             0
province                  0
dtype: int64
```

In [14]:

```
pizza_data = pizza_data.drop(['menuPageURL','menus.description','postalCode'], axis =1)
```

In [15]:

```
pizza_data.shape
```

Out[15]:

```
(10000, 21)
```

In [16]:

```
pizza_data.nunique()
```

Out[16]:

```
id                    2285
dateAdded             2214
dateUpdated           2193
address               2278
categories             456
primaryCategories        8
city                  1028
country                  1
keys                  2285
latitude              2284
longitude             2284
menus.amountMax        773
menus.amountMin        762
menus.currency           1
menus.dateSeen        6303
menus.name            4749
name                  1827
priceRangeCurrency       1
priceRangeMin           10
priceRangeMax           14
province                44
dtype: int64
```

In [17]:

```
pizza_data['primaryCategories'].unique()
```

Out[17]:

```
array(['Accommodation & Food Services',
       'Management of Companies & Enterprises',
       'Retail,Accommodation & Food Services',
       'Accommodation & Food Services,Management of Companies & Enterprise
s',
       'Retail',
       'Wholesale Trade,Accommodation & Food Services,Manufacturing',
       'Wholesale Trade,Accommodation & Food Services',
       'Educational Services,Accommodation & Food Services'], dtype=objec
t)
```

In [18]:

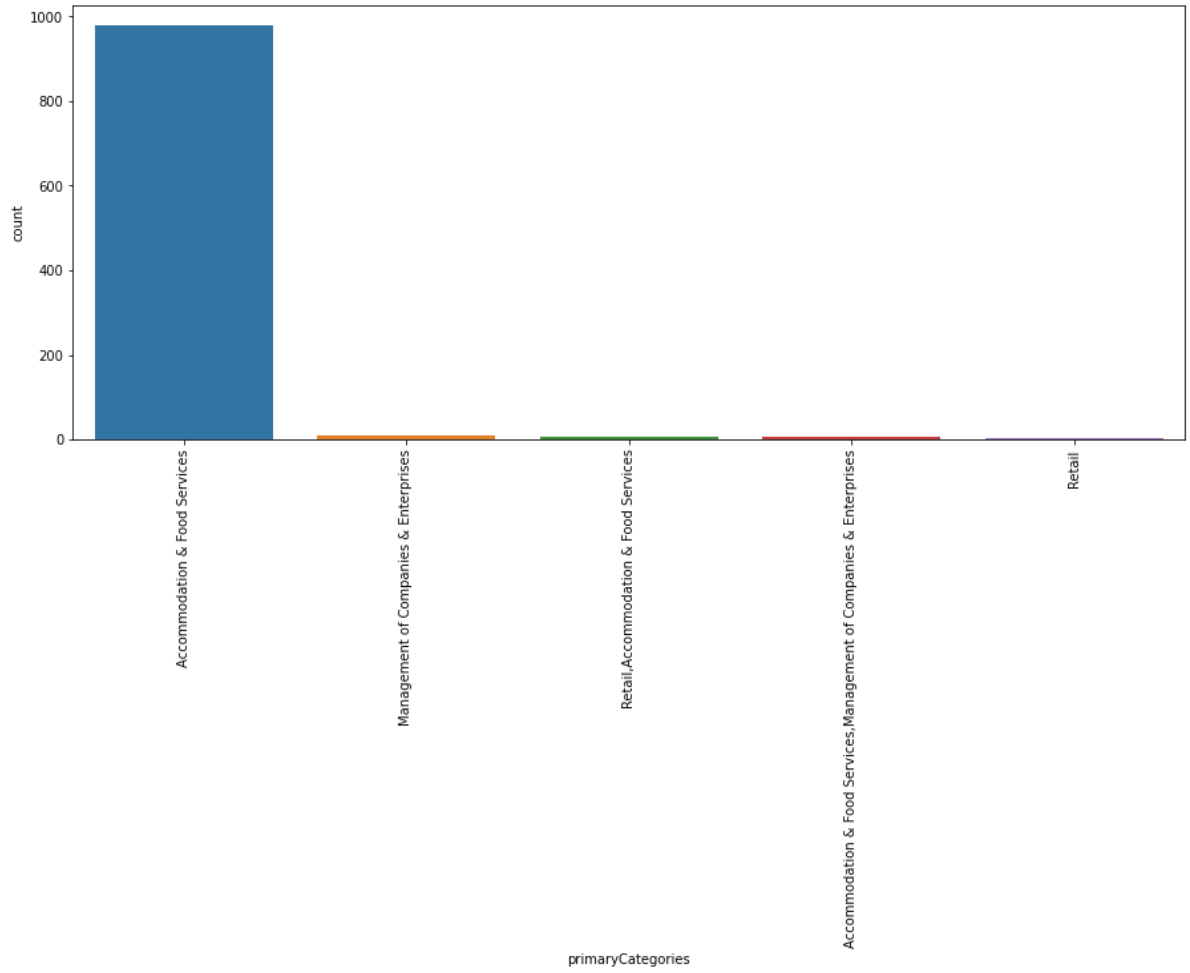
```
pizza_data['primaryCategories'].value_counts()
```

Out[18]:

```
Accommodation & Food Services                                   990
9
Management of Companies & Enterprises                             4
3
Accommodation & Food Services,Management of Companies & Enterprises      2
3
Retail,Accommodation & Food Services                              1
1
Retail
9
Wholesale Trade,Accommodation & Food Services
2
Educational Services,Accommodation & Food Services
2
Wholesale Trade,Accommodation & Food Services,Manufacturing
1
Name: primaryCategories, dtype: int64
```

In [60]:

```
plt.figure(figsize=(15,6))
sns.countplot('primaryCategories', data = pizza_data.head(1000))
plt.xticks(rotation = 90)
plt.show()
```



In [21]:

```
pizza_data['categories'].unique()
```

Out[21]:

```
array(['Pizza,Restaurant,American restaurants,Pizza Place,Restaurants',
       'Pizza,Pizza Place,Restaurants',
       'Restaurant,Pizza Place,Restaurants',
       'Pizza,Carry-out food,Pizza Place,Restaurants',
       'Pizza,American restaurants,Pizza Place,Pizza equipment and suppl
ies,Restaurants',
       'Pizza Place',
       'Restaurant,Italian Restaurant,Pizza Place,Italian Restaurant and
Pizza Place',
       'Pizza,Restaurant,Pizza Place',
       'Pizza,Restaurant,Pizza Place,Restaurants',
       'Pizza,American restaurants,Pizza Place,Restaurants',
       'Restaurant,Pizza Place Upper East Side,Pizza Place',
       'Restaurant,Pizza Place', 'Indian Restaurant,Pizza Place',
       'Pizza,Restaurant,Karaoke,Pizza Place,Restaurants',
       'Italian Restaurant,Pizza Place,Italian Restaurant and Pizza Plac
e',
```

In [22]:

```
pizza_data['categories'].value_counts()
```

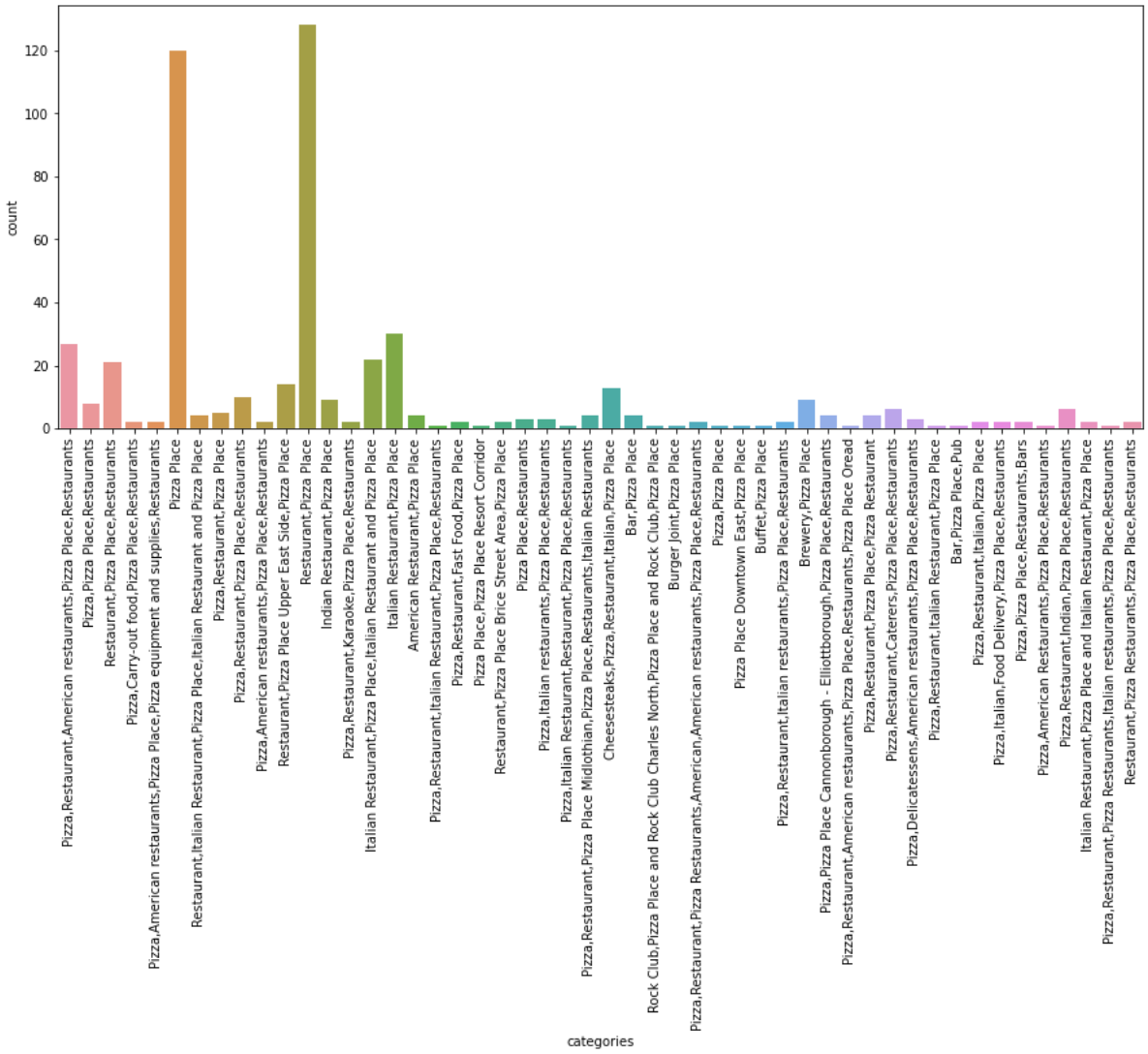
Out[22]:

```
Restaurant,Pizza Place
2690
Pizza Place
2209
Pizza,Restaurant,Pizza Place,Restaurants
370
Italian Restaurant,Pizza Place
312
Restaurant,Pizza Place,Restaurants
293

...
Bar,Pizza Place,Pizza Place and Bar
1
Pizza Place, Sandwich Place, and American Restaurant,Pizza Place,Restauran
ts                                1
Pizza,Restaurant,Pizza Restaurants,Caterers,Pizza Place,Pizza Place East F
alls,Catering,Restaurants         1
Pizza,Franchising,Pizza Place,Restaurants
1
Pizza Place,Restaurant,Fast Food,Pizza
1
Name: categories, Length: 456, dtype: int64
```

In [25]:

```
plt.figure(figsize=(15,6))
sns.countplot('categories', data = pizza_data.head(500))
plt.xticks(rotation = 90)
plt.show()
```



In [26]:

```
pizza_data['city'].unique()
```

Out[26]:

```
array(['Sherwood', 'Phoenix', 'Cincinnati', ..., 'Red Hook',
       'South Holland', 'Palmer'], dtype=object)
```

In [27]:

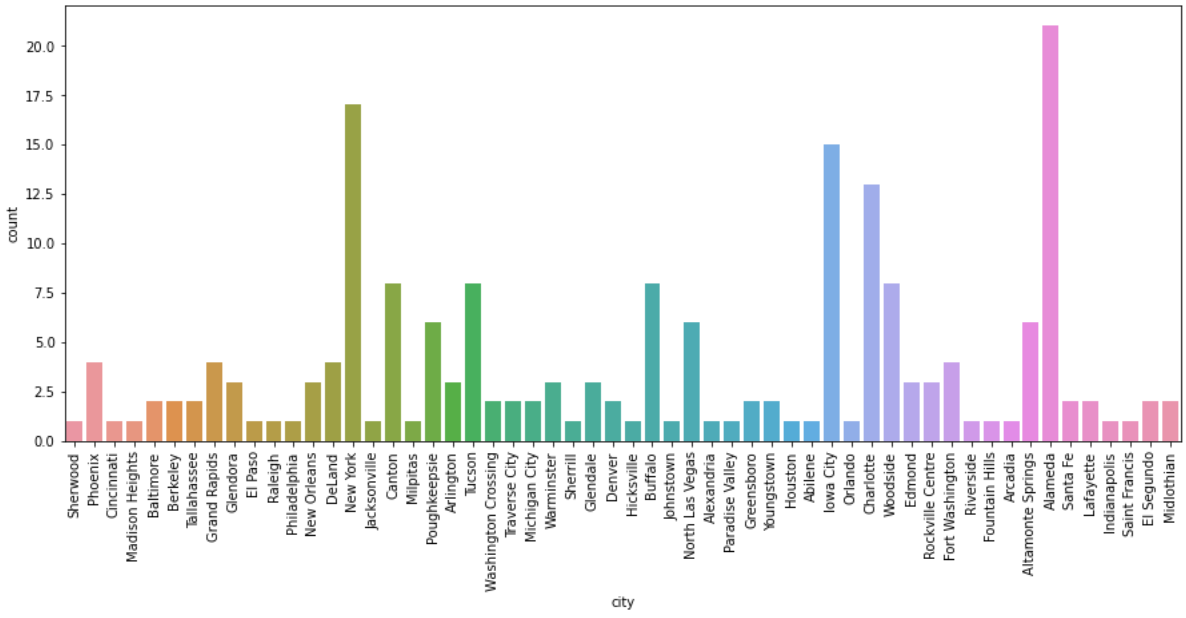
```
pizza_data['city'].value_counts()
```

Out[27]:

```
New York         655
Brooklyn         460
Los Angeles      193
Buffalo          178
Philadelphia     140
                ...
Edmonds            1
Bend               1
Fox Lake           1
Bellevue           1
Palmer             1
Name: city, Length: 1028, dtype: int64
```

In [30]:

```
plt.figure(figsize=(15,6))
sns.countplot('city', data = pizza_data.head(200))
plt.xticks(rotation = 90)
plt.show()
```



In [31]:

```
pizza_data['menus.amountMax'].unique()
```

Out[31]:

```
array([7.9800e+00, 6.0000e+00, 6.4900e+00, 5.9900e+00, 5.4900e+00,
       1.0990e+01, 4.9900e+00, 5.3900e+00, 2.5000e+00, 1.1990e+01,
       7.4300e+00, 1.8210e+01, 1.5920e+01, 2.2000e+01, 1.1500e+01,
       7.0000e+00, 5.0000e+00, 4.5000e+00, 1.2250e+01, 1.3500e+01,
       4.8900e+00, 6.9900e+00, 1.9990e+01, 1.1750e+01, 3.7500e+00,
       3.5000e+00, 1.9250e+01, 1.6950e+01, 1.2500e+01, 1.7750e+01,
       1.3750e+01, 1.8750e+01, 1.2000e+01, 1.8950e+01, 1.9750e+01,
       7.9900e+00, 9.2000e+00, 1.8990e+01, 1.9900e+00, 6.9500e+00,
       1.2950e+01, 1.0000e+00, 8.9900e+00, 8.4900e+00, 1.7000e+01,
       2.1000e+01, 1.0490e+01, 4.8500e+00, 1.2050e+01, 1.7500e+00,
       1.2990e+01, 9.2500e+00, 5.9500e+00, 1.5000e+00, 2.2490e+01,
       3.2990e+01, 1.5990e+01, 1.7990e+01, 1.6990e+01, 4.2900e+00,
       5.7500e+00, 2.0990e+01, 4.9990e+01, 9.9900e+00, 1.4990e+01,
       1.5950e+01, 3.9900e+00, 5.5000e+00, 4.2500e+00, 1.3000e+01,
       1.5000e+01, 1.0000e+01, 3.0000e+00, 1.4000e+01, 1.4950e+01,
       1.3950e+01, 2.1990e+01, 1.5750e+01, 1.6000e+01, 1.6250e+01,
       1.8250e+01, 2.4000e+01, 4.5900e+00, 2.1950e+01, 9.5000e+00,
```

In [32]:

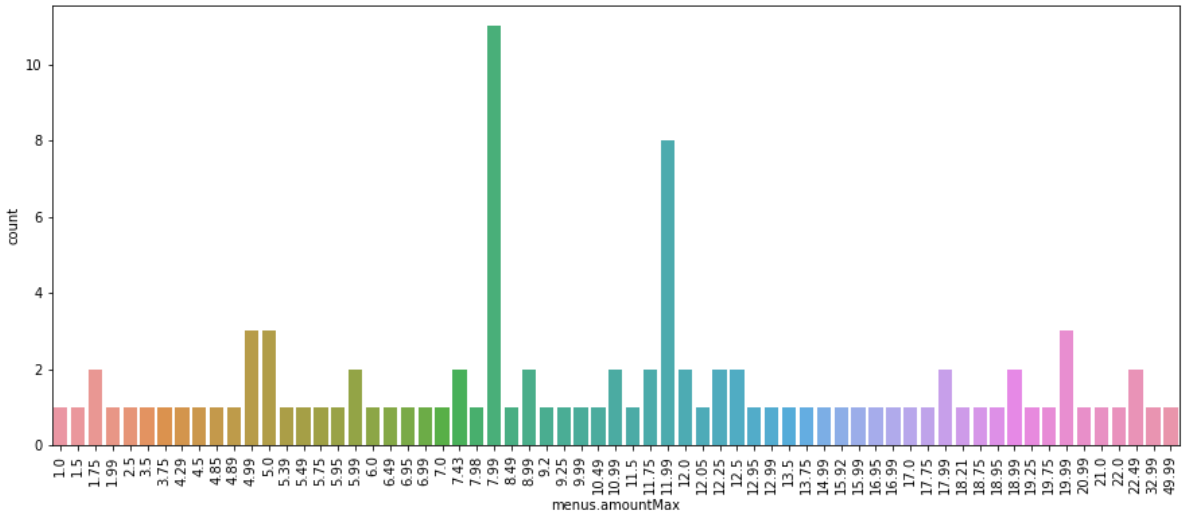
```
pizza_data['menus.amountMax'].value_counts()
```

Out[32]:

```
16.00    220
15.00    192
14.00    186
9.99     183
13.99    169
        ...
7.07       1
18.69      1
23.39      1
28.60      1
10.10      1
Name: menus.amountMax, Length: 773, dtype: int64
```

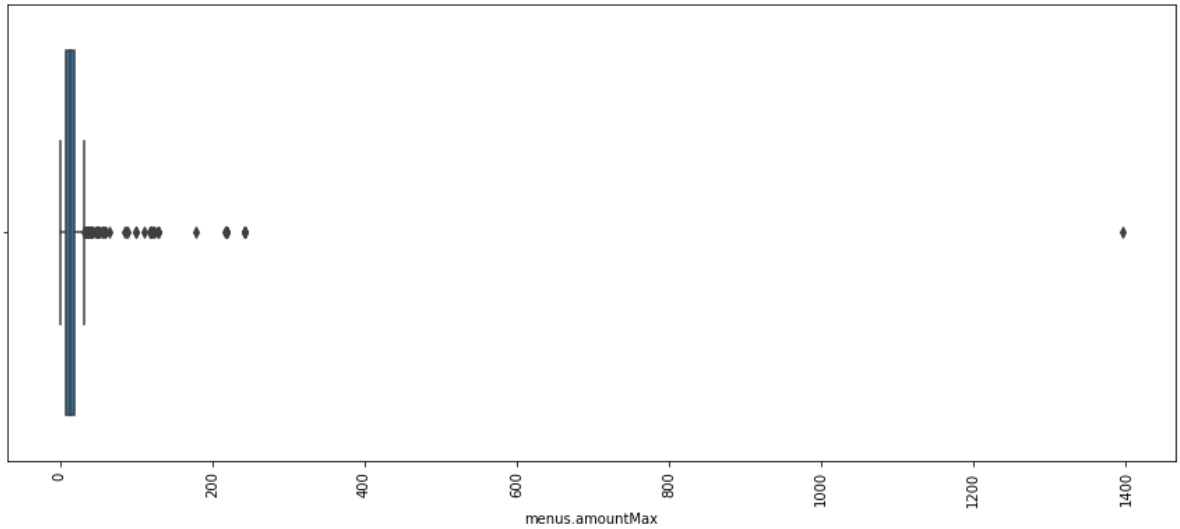
In [34]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMax', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



In [40]:

```
plt.figure(figsize=(15,6))
sns.boxplot('menus.amountMax', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [35]:

```
pizza_data['menus.amountMin'].unique()
```

Out[35]:

```
array([  7.98,   6.  ,   6.49,   5.99,   5.49,  10.99,   4.99,   5.39,
         2.5 ,  11.99,   7.43,  18.21,  15.92,  22.  ,  11.5 ,   7.  ,
         5.  ,   4.5 ,  12.25,  13.5 ,   4.89,   6.99,  19.99,  11.75,
         3.75,   3.5 ,  19.25,  16.95,  12.5 ,  17.75,  13.75,  18.75,
        12.  ,  18.95,  19.75,   7.99,   9.2 ,  18.99,   1.99,   6.95,
        12.95,   1.  ,   8.99,   8.49,  17.  ,  21.  ,  10.49,   4.85,
        12.05,   1.75,  12.99,   9.25,   5.95,   1.5 ,  22.49,  15.99,
        17.99,  16.99,   4.29,   5.75,  14.99,  49.99,   9.99,  15.95,
         3.99,   5.5 ,   4.25,  13.  ,  15.  ,  10.  ,   3.  ,  14.  ,
        14.95,  13.95,  21.99,  15.75,  20.99,  16.  ,  16.25,  18.25,
        24.  ,   4.59,  14.45,   9.5 ,   2.  ,  20.  ,   6.25,  11.95,
         4.  ,   2.15,  15.49,   8.25,  27.  ,   2.75,   9.  ,  20.49,
        16.49,  17.29,  16.79,  13.79,  12.49,  14.89,  14.29,  11.49,
        15.79,  17.49,  13.49,  12.79,  16.29,  17.89,   6.5 ,  18.  ,
        13.99,   7.5 ,   1.25,  12.85,   8.  ,   3.95,   8.69,  25.  ,
        19.  ,   8.55,   8.9 ,  19.95,  17.95,   3.39,  14.75,   8.95,
        11.25,  14.34,   6.27,   0.  ,   6.89,  14.49,   8.5 ,   7.75,
         9.75,  10.95,   7.25,  11.  ,   8.89,  29.99,  27.99,  35.99,
         2.99,  21.95,  31.95,  20.95,  13.38,   9.49,  13.55,   4.3 ,
        17.9 ,  21.25,  15.4 ,  23.25,  11.2 ,  16.45,  15.45,  17.45,
         7.3 ,  18.55,   3.25,   2.25,   1.7 ,  10.25,  10.5 ,  16.5 ,
         3.49,   7.19,   3.69,   7.49,   7.09,   7.95,   8.39,  13.05,
        18.15,   6.2 ,   6.4 ,  16.75,   4.05,   3.79,   3.29,  30.  ,
        24.95,   8.45,   3.59,   7.65,  11.85,   9.43,   8.29,   4.39,
         2.59,   3.85,  23.95,  20.5 ,  23.5 ,  17.5 ,  15.5 ,  22.95,
        19.5 ,   3.55,  22.5 ,  18.5 ,  15.25,   8.75,   9.95,  17.25,
        12.7 ,  11.05,   2.95,   5.25,  14.5 ,   4.49,   0.25,   6.79,
         1.85,  12.65,  34.99,  11.45,  17.35,  22.89,   2.79,  10.75,
         4.95,  21.5 ,  10.85,   2.4 ,  18.65,   2.49,  14.26,  21.51,
        28.99,  29.98,  30.45,  19.9 ,  20.65,  21.3 ,  13.4 ,  11.9 ,
         8.35,  29.2 ,   8.65,  28.5 ,  13.69,   6.75,  23.  ,   5.19,
         6.29,   1.98,  12.4 ,  11.15,   2.76,   3.9 ,   4.13,   2.3 ,
         3.44,   7.69,   5.2 ,   2.85,  25.95,  28.  ,  25.41,   5.08,
        22.99,  24.99,  14.9 ,   4.75,  11.6 ,  13.45,  15.3 ,  12.1 ,
         5.59,  12.59,   9.39,   2.29,  13.25,  25.99,  15.94,   6.3 ,
         3.8 ,  14.4 ,   8.15,  15.15,  10.15,   9.1 ,  10.4 ,  35.  ,
        10.8 ,  38.99,   8.79,  12.19,  18.54,   6.19,  19.89,   9.69,
        15.54,  20.25,  18.98,  55.  ,  65.  ,   7.15,  23.99,   7.45,
        24.5 ,  11.4 ,   5.29,   5.9 ,   5.1 ,   5.15,   6.45,  24.4 ,
         5.8 ,  16.15,  26.95,   1.95,  21.79,  20.29,  11.29,  10.29,
        16.59,  18.79,   6.76,  14.25,   3.19,  14.69,  19.49,   1.55,
        13.65,  32.  ,  26.  ,  17.65,  43.98,  19.98,   7.59,  85.  ,
         6.1 ,   4.6 ,  25.55,   9.45,   8.59,  12.89,   3.89, 179.  ,
         1.69,   0.5 ,  12.15,  21.35,  26.25,  12.75,  20.75,  12.55,
        12.3 ,   8.7 ,  20.9 ,  17.4 ,   2.35,  27.5 ,  12.35,  11.26,
         4.69,   5.89,   4.55,   4.45,  23.49,  33.99,  14.35,  16.67,
        27.55,  11.7 ,  32.99,  30.99,  39.99,   7.89,   1.65,   7.39,
        27.9 ,  35.34,   4.65,  16.19,   6.69,  26.5 ,   4.93,   4.44,
        10.79,  16.89,   5.05,   6.74,  17.71,  22.02,   4.91,   2.94,
         2.9 ,   4.1 ,   3.1 ,  34.  ,  39.  ,  29.  ,  23.55,  15.55,
        15.35,  10.6 ,   0.99,   1.9 ,   9.8 ,   5.7 ,  13.6 ,  11.35,
        11.65,  11.55,   4.79,   9.55,   5.97,   4.73,  14.43,  27.75,
         2.65,  32.95,  28.95,   6.35,   2.69,  15.98,   3.6 ,   9.35,
       118.99,  16.8 ,  22.4 ,  13.8 ,  15.8 ,  17.8 ,   6.6 ,   1.6 ,
```

```
 10.35,  14.85,  14.59,  18.49,   9.79,  41.5 ,   8.05,  26.4 ,
 12.69,  14.79,  15.59,   5.79, 218.99,  15.39,   5.69, 128.  ,
243.  ,  18.29,  15.29,  13.89,  17.09,  16.69,   0.65,   8.8 ,
 27.95,  33.  ,  12.67,  14.58,  25.49,   9.19,   9.98,  26.1 ,
 12.45,  48.5 ,  13.1 ,  22.55,  24.2 ,  23.04,  28.6 ,   1.49,
  2.7 ,  18.85,  16.7 ,  14.65,  18.69,   7.07,   0.85,   4.52,
 12.2 ,  18.7 ,  11.96,   5.47,  19.77,   5.45,   9.4 ,  21.75,
 18.3 ,  17.3 ,  29.55,  37.99,   4.19,   0.59,   6.55,  12.29,
 33.95,  32.45,  22.45,  29.95,  16.2 ,  27.45,   0.6 ,  26.49,
  1.46,   0.39,  13.26,  23.75,  28.75,   9.6 ,   4.35,  26.89,
 23.89,  19.79,  14.19,   0.75,  30.98,  21.87,  41.  ,  40.  ,
 27.34,  28.25,  25.13,  35.54,  24.25,  36.89,  36.98,   5.96,
 57.79,  32.89,  11.89,  50.99,  35.59,  31.98,  11.39,  56.09,
 25.5 ,   9.74,   9.29,  22.69,  27.79,  26.99,   8.24,  26.59,
 26.34,   3.05,  11.79,  11.69,  14.39,  15.89,  13.29,  17.39,
 17.79,  22.15,   8.3 ,  22.65,  11.3 , 100.  ,  59.99,   2.1 ,
  6.7 ,   9.56,  10.55,  39.95,  22.75,  34.95,  18.9 ,   2.45,
 17.59,   4.2 ,   3.45,  10.45,  21.45,  26.98,  18.44,   9.15,
 16.56,  21.98,  14.7 ,  19.4 ,  31.99,  42.99,  13.44,  15.07,
  6.05,   1.67,   1.97,  14.27, 110.95,  10.98,  21.49,  11.98,
 20.98,   8.98,  10.58,  12.09,  38.  ,   5.09,   6.39,   2.6 ,
  1.8 ,  16.9 ,  24.75,  32.97,   7.78,   7.2 ,  12.8 ,   3.4 ,
  9.89,  14.8 ,  19.8 ,  22.85,   8.4 ,  28.1 ,  26.55,  24.45,
 20.7 ,  17.17,  27.38,  24.7 ,  23.4 ,  25.2 ,  18.35,  24.35,
 21.2 ,  24.59,  24.6 ,  20.1 ,  25.1 ,  15.31,   4.24,  12.53,
 16.64,   5.12,   2.36,  15.05,  10.3 ,  16.05,  18.05,  13.3 ,
 16.1 ,  19.45,  20.4 ,  25.25,   1.79,  18.6 ,  26.75,   9.85,
 13.84,   4.09,  10.05,  59.95,   5.92,   7.92,  31.  ,   7.6 ,
  8.6 ,   5.68,  15.19,   9.24,  16.26,  13.19,  10.59, 116.99,
216.99,  22.2 ,   1.94,   0.9 ,  13.9 ,  16.6 ,  15.2 ,  15.85,
  3.22,  14.15,   2.19,   3.65,   5.85,  21.15,  21.1 ,  13.34,
 16.27,   3.77, 123.  ,  87.  ,   0.7 ,   7.8 ,   2.04,  12.98,
 17.1 ,  13.2 ,   4.9 ,   9.05,  15.46,   4.15,  13.15,   9.3 ,
 23.45,  25.45,  20.45,  13.7 ,  15.09,  14.3 ,   1.1 ,   6.65,
 17.7 ,  14.64,  16.4 ,  12.6 ,  10.65,   7.35,  17.6 ,  19.85,
 18.1 ,  20.05,   7.85,  16.09,   8.33,   2.78,   7.41,   5.65,
  4.74,  25.75,  22.25,  19.54,  18.47,   5.46,  16.35,  10.39,
  5.54,   8.16,  11.62,   1.2 ,   1.59,  10.07,   9.53,  22.59,
  9.59,  18.19,  12.39,   8.85,  47.95,  38.95,  22.6 ,  34.85,
 28.4 ,  21.85,  42.6 ,  14.6 ,   0.69,   8.19,   0.35,   4.8 ,
  3.09,   4.84,  19.15,  20.89,   5.11,  12.41,   3.87,   4.4 ,
  7.1 ,  10.1 ])
```

In [36]:

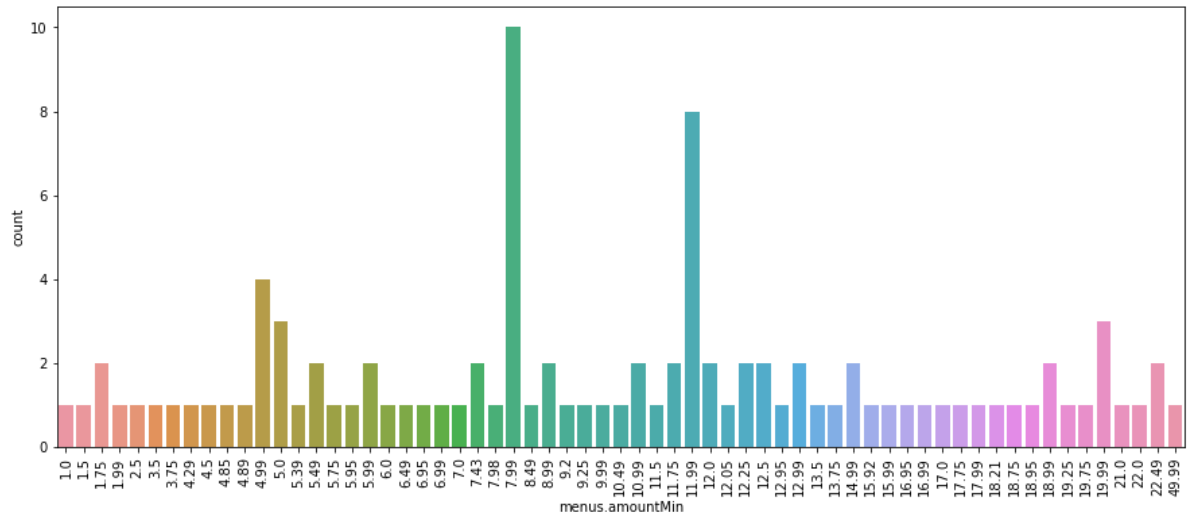
```
pizza_data['menus.amountMin'].value_counts()
```

Out[36]:

```
16.00    219
15.00    197
13.99    191
9.99     189
14.00    188
        ...
32.45      1
29.55      1
9.40       1
19.77      1
10.10      1
Name: menus.amountMin, Length: 762, dtype: int64
```

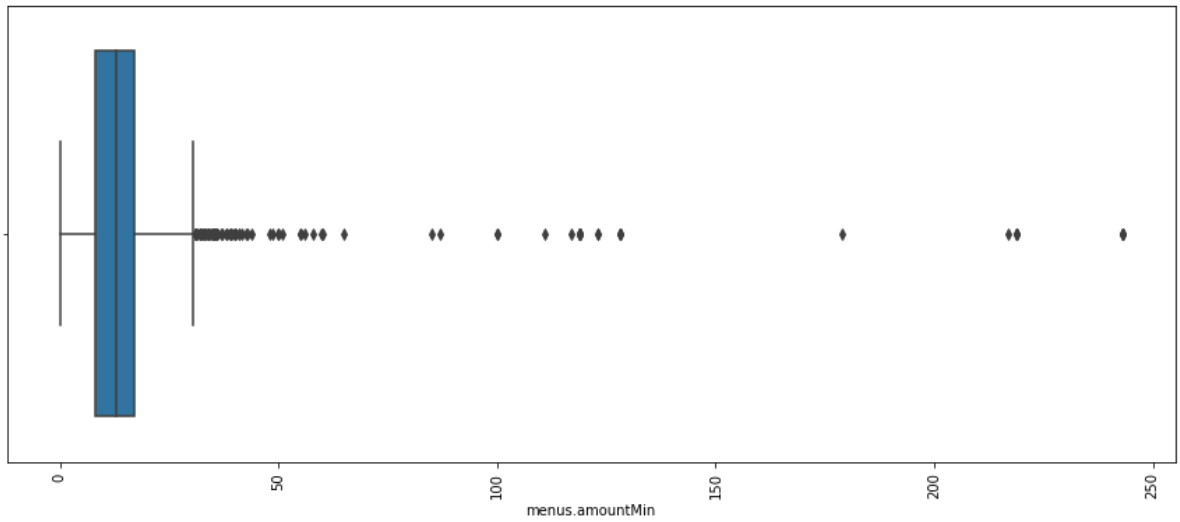
In [37]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMin', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



In [41]:

```
plt.figure(figsize=(15,6))
sns.boxplot('menus.amountMin', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [42]:

```
pizza_data['priceRangeMin'].unique()
```

Out[42]:

```
array([ 0, 25, 40,  1, 50,  3,  2, 10,  5, 20], dtype=int64)
```

In [43]:

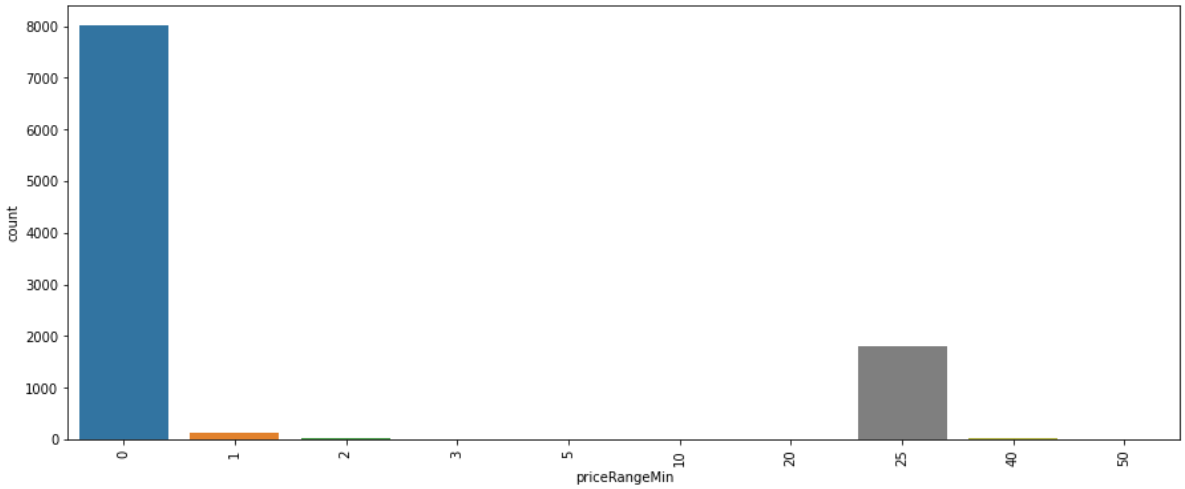
```
pizza_data['priceRangeMin'].value_counts()
```

Out[43]:

```
0     8007
25    1801
1      133
40      21
2       16
50       9
5        6
3        5
10       1
20       1
Name: priceRangeMin, dtype: int64
```

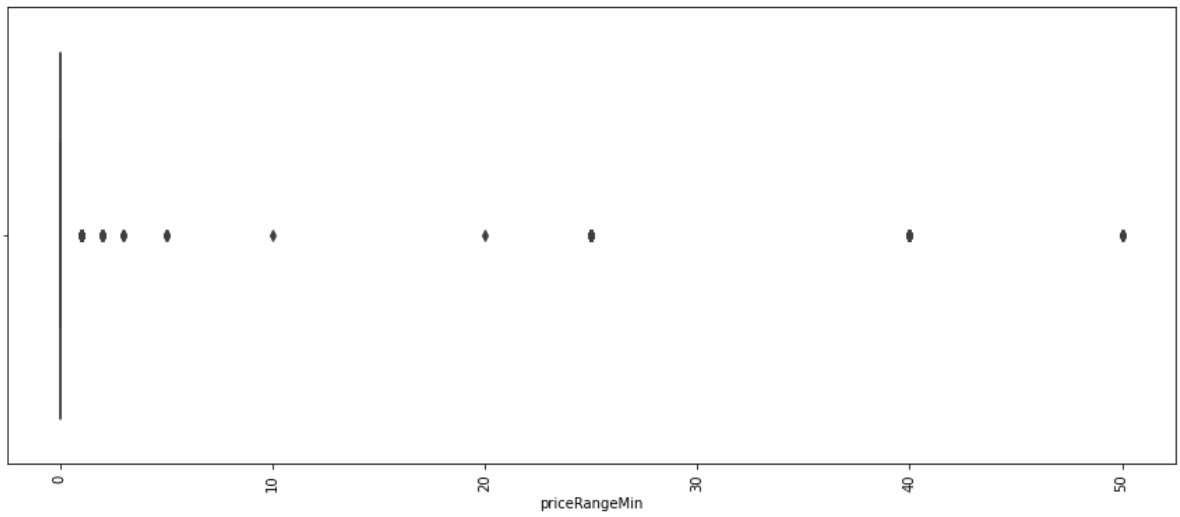
In [45]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMin', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [46]:

```
plt.figure(figsize=(15,6))
sns.boxplot('priceRangeMin', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [47]:

```
pizza_data['priceRangeMax'].unique()
```

Out[47]:

```
array([25, 40, 30,  7, 55, 26, 24, 33, 20, 35, 18, 19, 22, 10],
      dtype=int64)
```

In [48]:

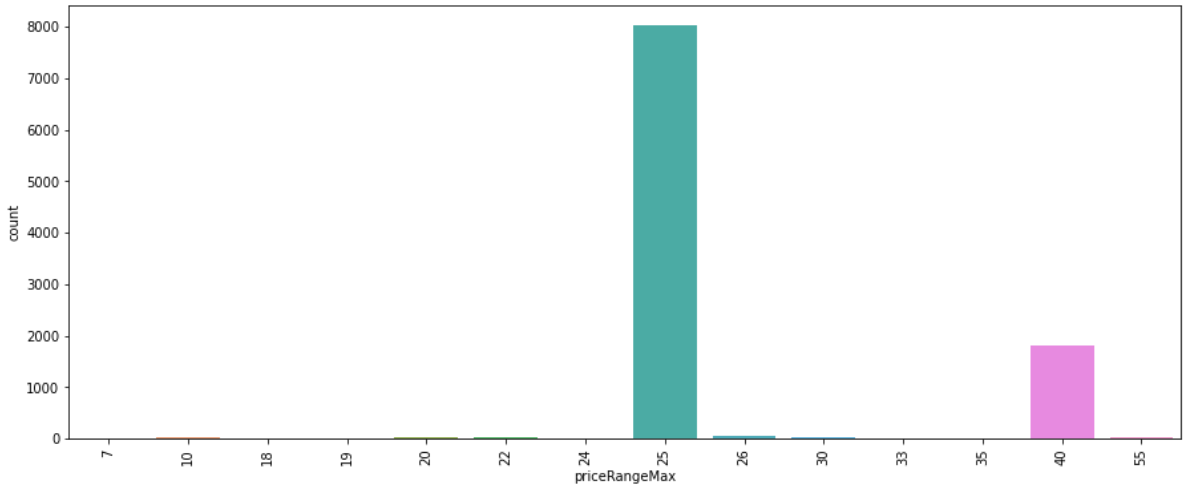
```
pizza_data['priceRangeMax'].value_counts()
```

Out[48]:

```
25    8016
40    1801
26      58
55      30
22      28
20      18
30      16
10      16
33       8
18       4
7        2
24       1
35       1
19       1
Name: priceRangeMax, dtype: int64
```

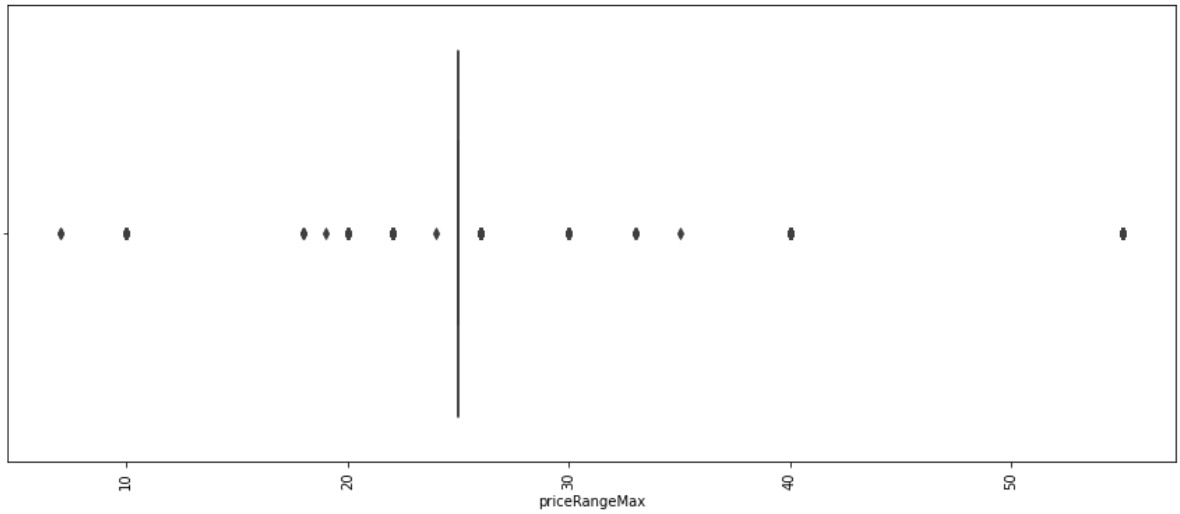
In [49]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMax', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [50]:

```
plt.figure(figsize=(15,6))
sns.boxplot('priceRangeMax', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



In [52]:

```
pizza_data['province'].unique()
```

Out[52]:

```
array(['AR', 'AZ', 'OH', 'MI', 'MD', 'CA', 'FL', 'TX', 'NC', 'PA', 'LA',
       'NY', 'IN', 'CO', 'NV', 'IA', 'OK', 'NM', 'WI', 'VA', 'OR', 'MN',
       'TN', 'WA', 'SC', 'MS', 'MO', 'GA', 'IL', 'DE', 'KS', 'UT', 'KY',
       'NE', 'SD', 'WV', 'AL', 'AK', 'ID', 'CT', 'HI', 'NJ', 'ND', 'MT'],
      dtype=object)
```

In [53]:

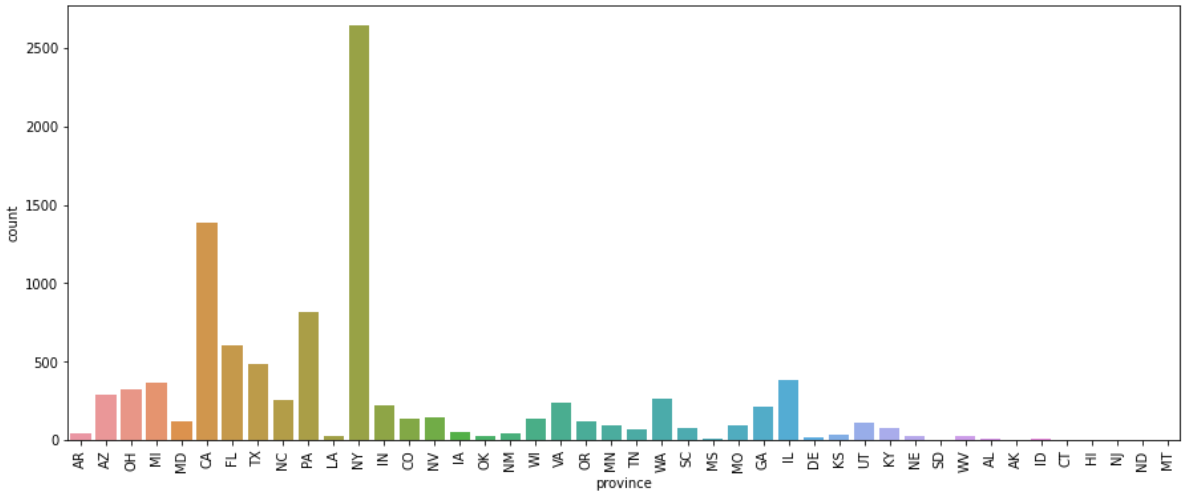
```
pizza_data['province'].value_counts()
```

Out[53]:

```
NY    2640
CA    1386
PA     814
FL     601
TX     485
IL     381
MI     366
OH     324
AZ     288
WA     262
NC     256
VA     237
IN     219
GA     214
NV     149
WI     140
CO     138
OR     122
MD     118
UT     107
MN      96
MO      95
SC      81
KY      77
TN      66
IA      53
NM      46
AR      40
KS      38
OK      27
WV      27
LA      25
NE      25
DE      19
MS       8
ID       6
AL       5
NJ       4
SD       3
AK       3
ND       3
CT       2
HI       2
MT       2
Name: province, dtype: int64
```

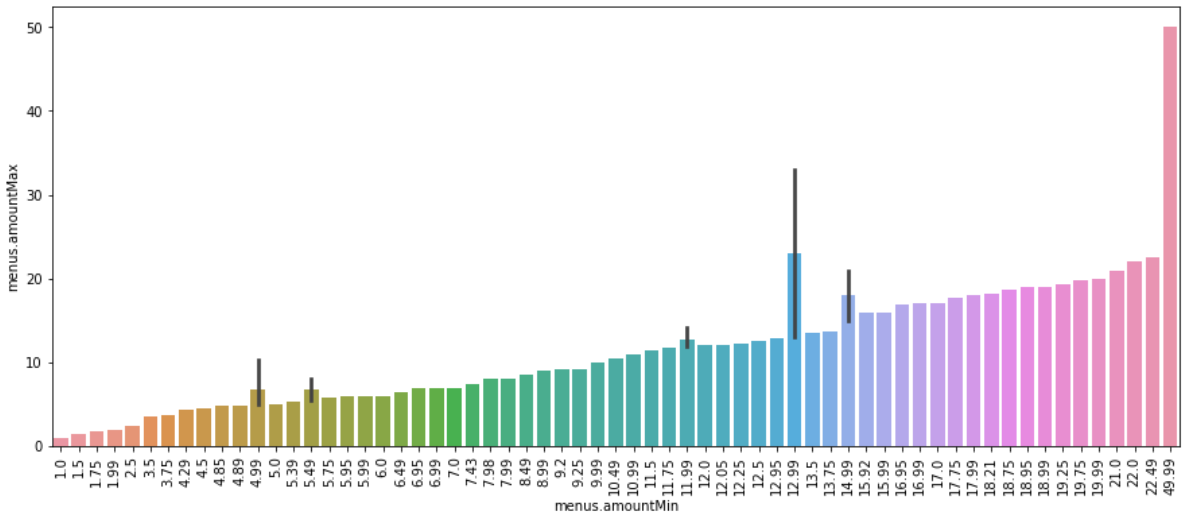
In [55]:

```
plt.figure(figsize=(15,6))
sns.countplot('province', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



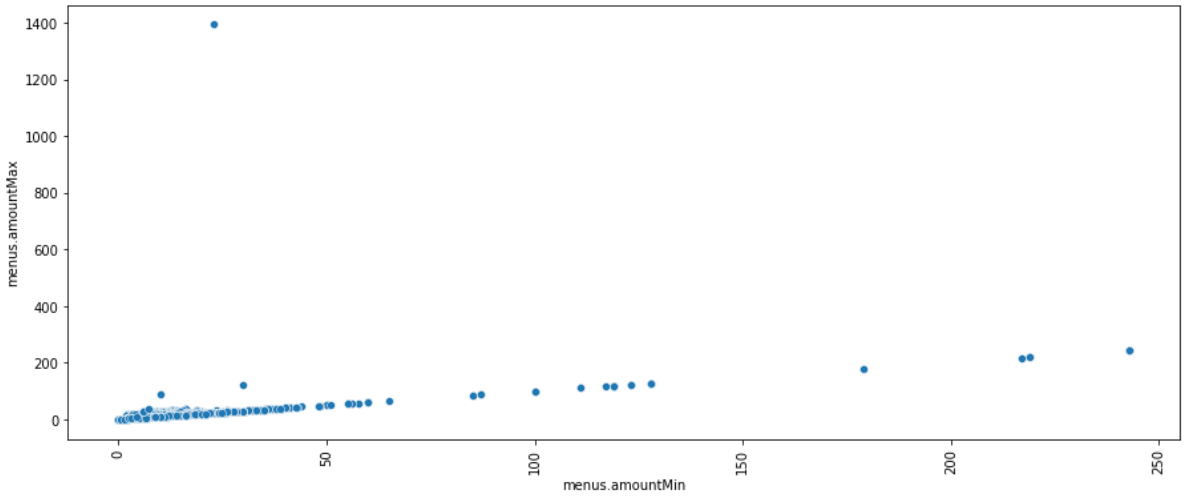
In [70]:

```
plt.figure(figsize=(15,6))
sns.barplot(y = 'menus.amountMax', x = 'menus.amountMin',
                data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



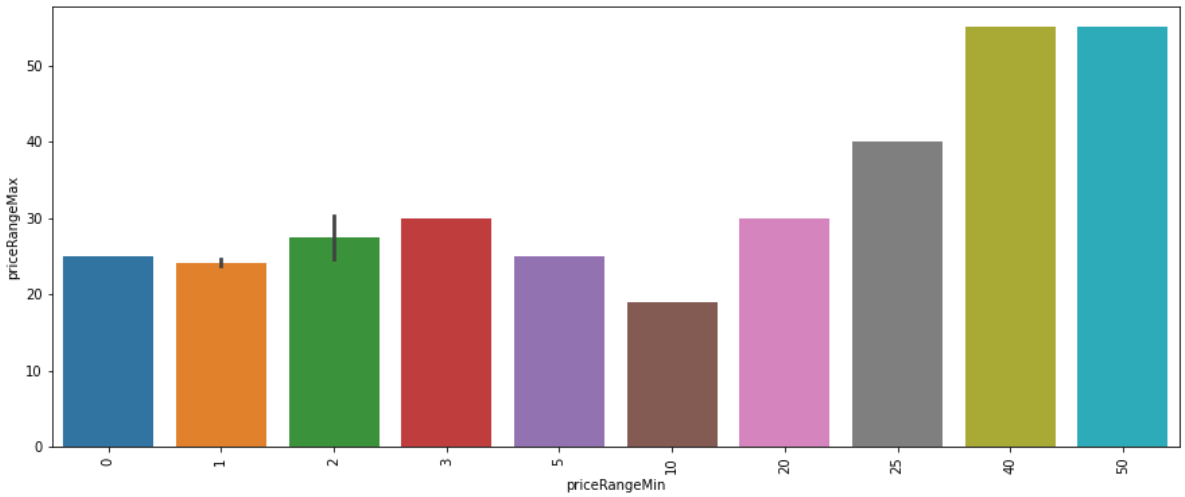
In [66]:

```
plt.figure(figsize=(15,6))
sns.scatterplot(y = 'menus.amountMax', x = 'menus.amountMin',
                data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



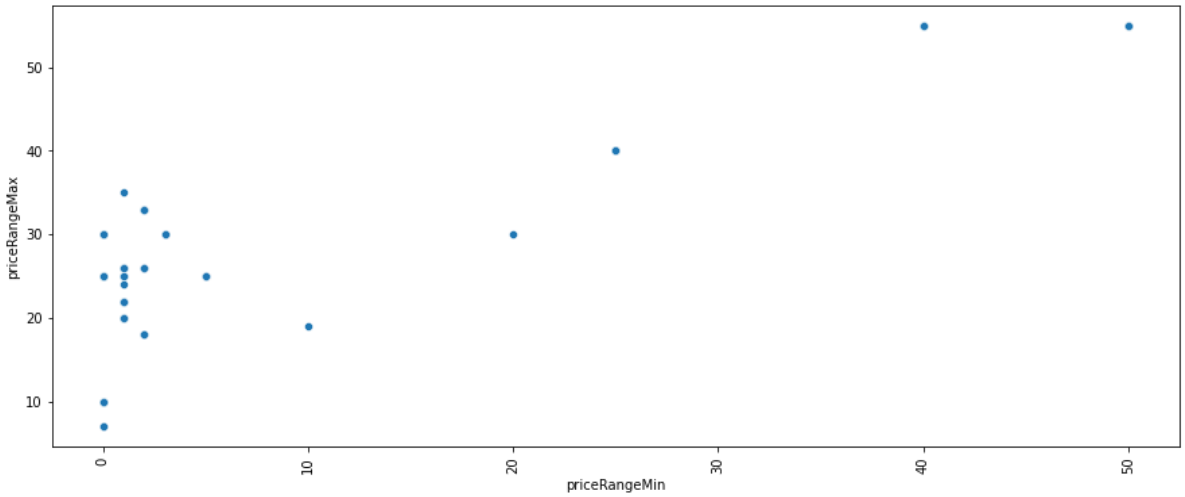
In [72]:

```
plt.figure(figsize=(15,6))
sns.barplot(y = 'priceRangeMax', x = 'priceRangeMin',
                data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



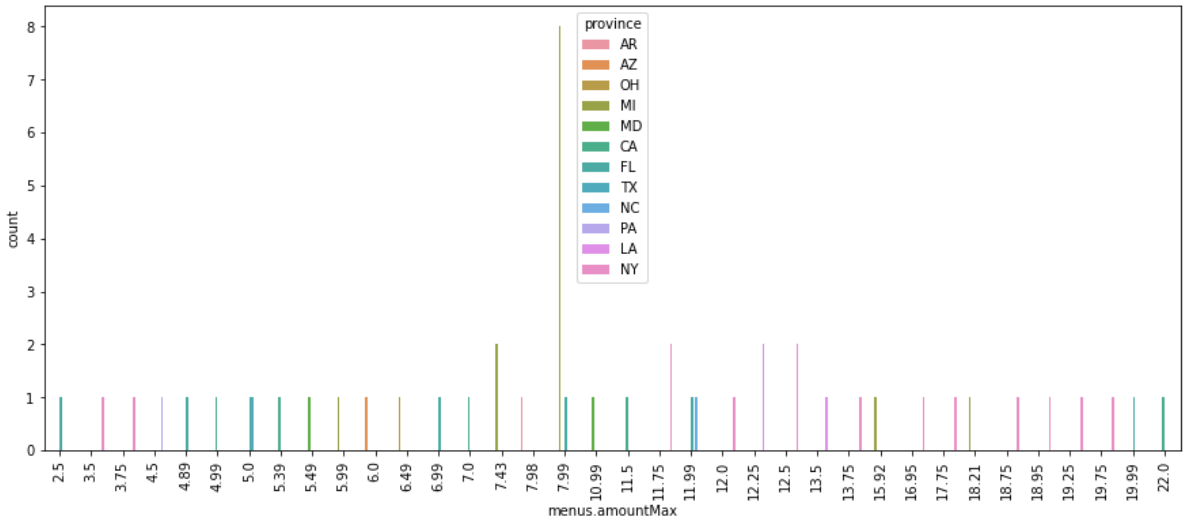
In [73]:

```
plt.figure(figsize=(15,6))
sns.scatterplot(y = 'priceRangeMax', x = 'priceRangeMin',
                data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



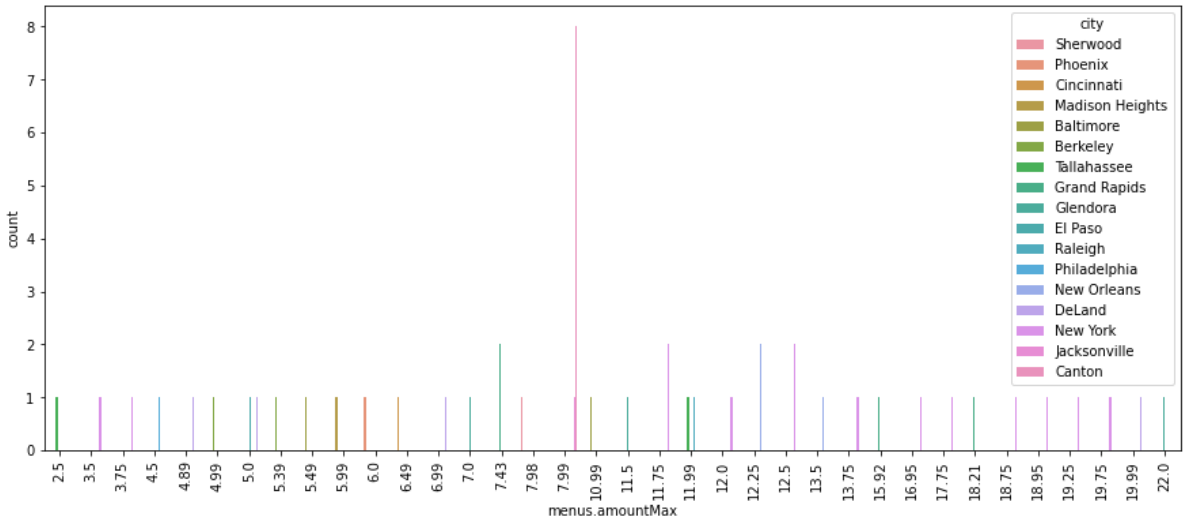
In [80]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMax', hue = 'province', data = pizza_data.head(50))
plt.xticks(rotation = 90)
plt.show()
```



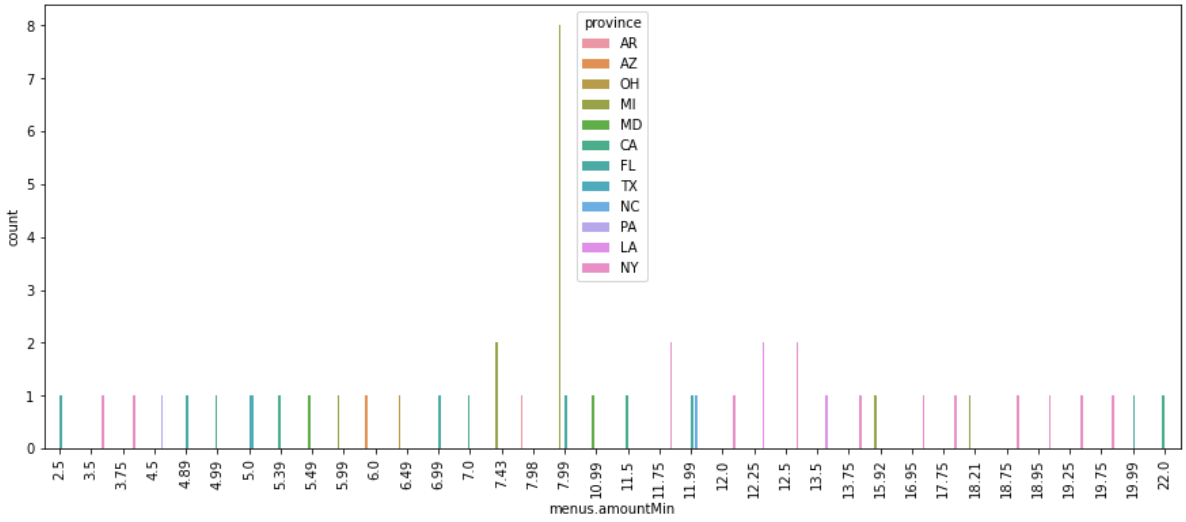
In [77]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMax', hue = 'city', data = pizza_data.head(50))
plt.xticks(rotation = 90)
plt.show()
```



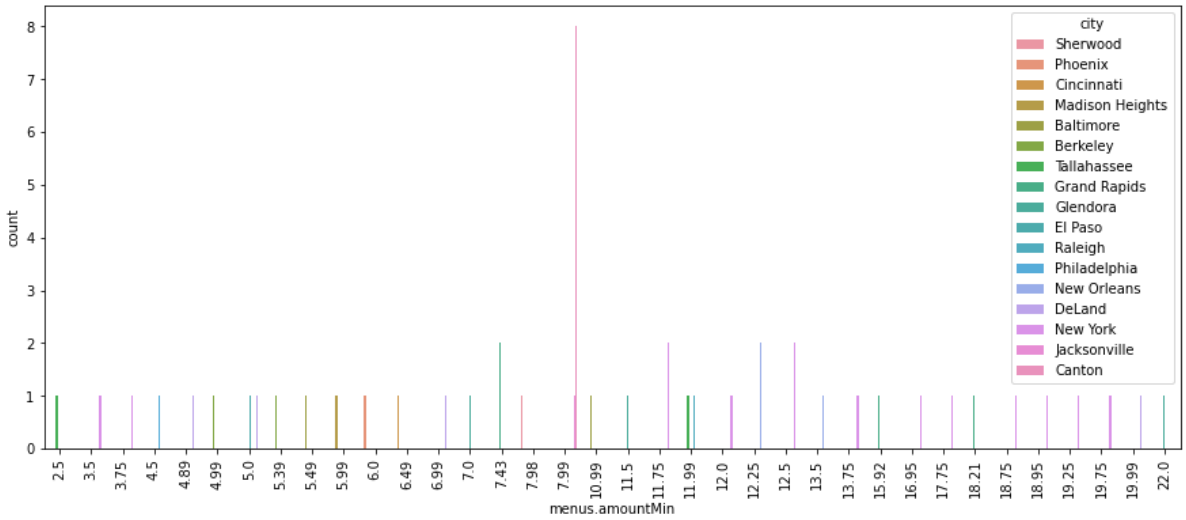
In [78]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMin', hue = 'province', data = pizza_data.head(50))
plt.xticks(rotation = 90)
plt.show()
```



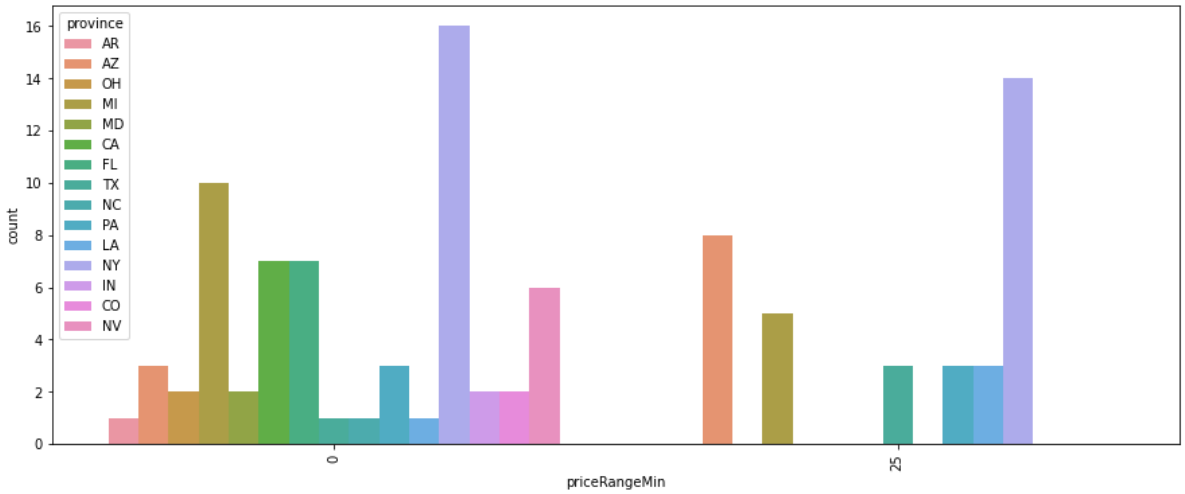
In [81]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMin', hue = 'city', data = pizza_data.head(50))
plt.xticks(rotation = 90)
plt.show()
```



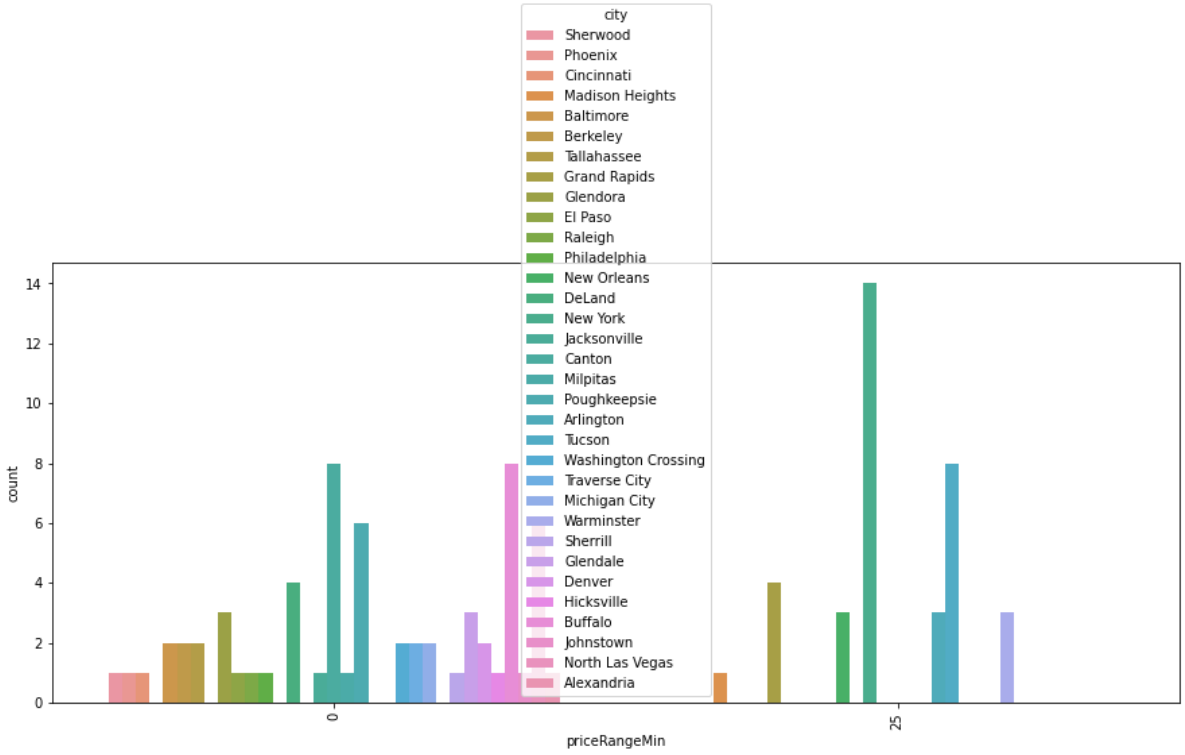
In [83]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMin', hue = 'province', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



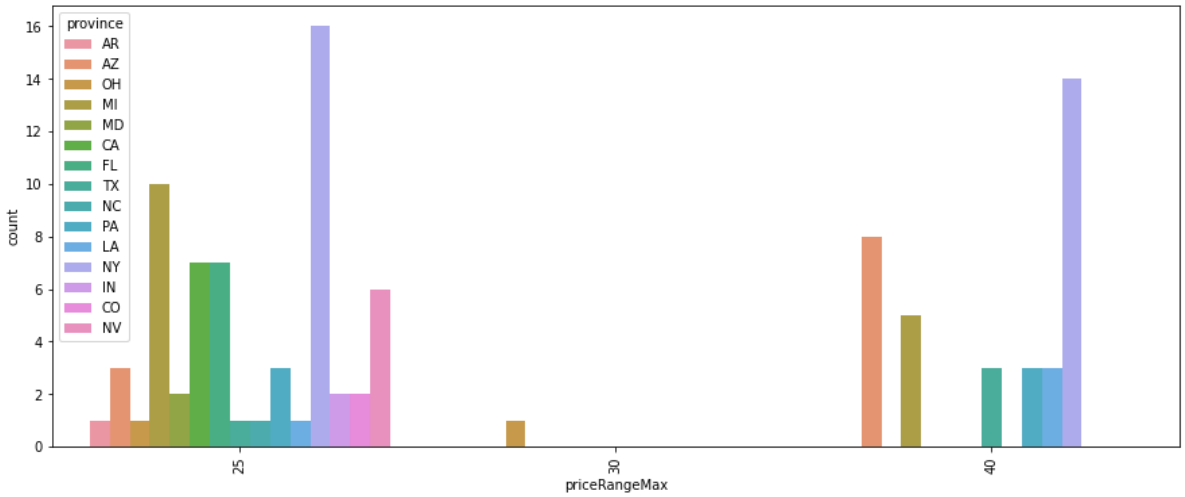
In [84]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMin', hue = 'city', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



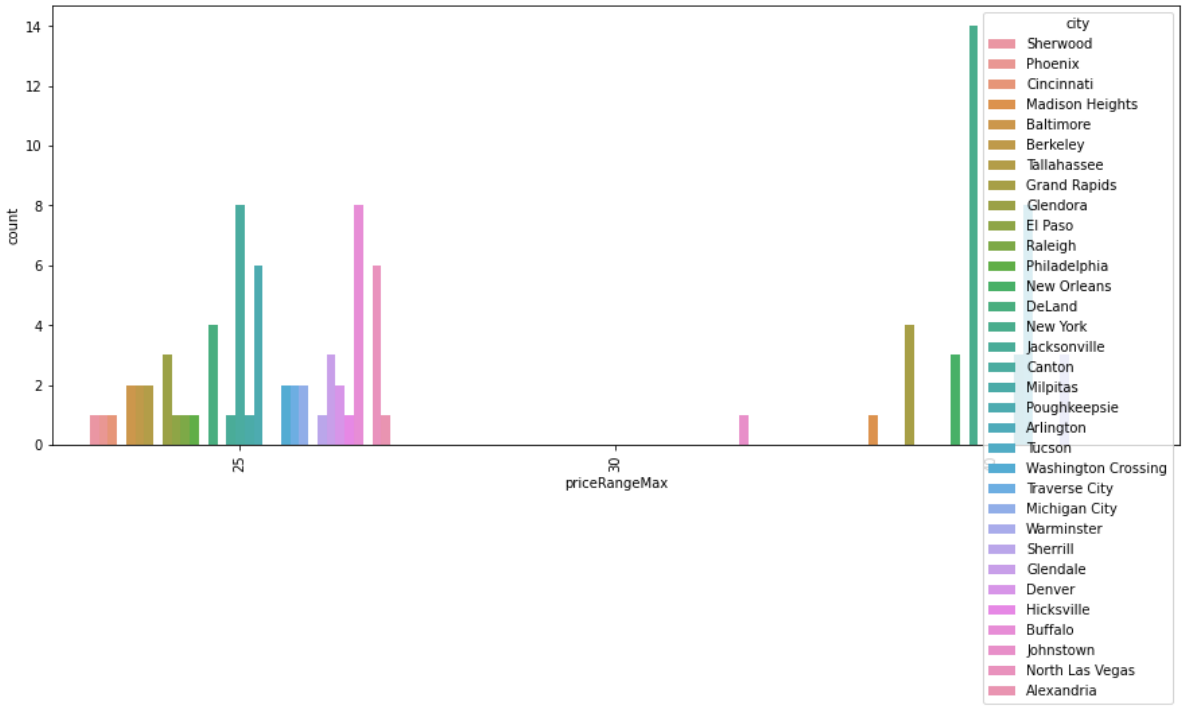
In [85]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMax', hue = 'province', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



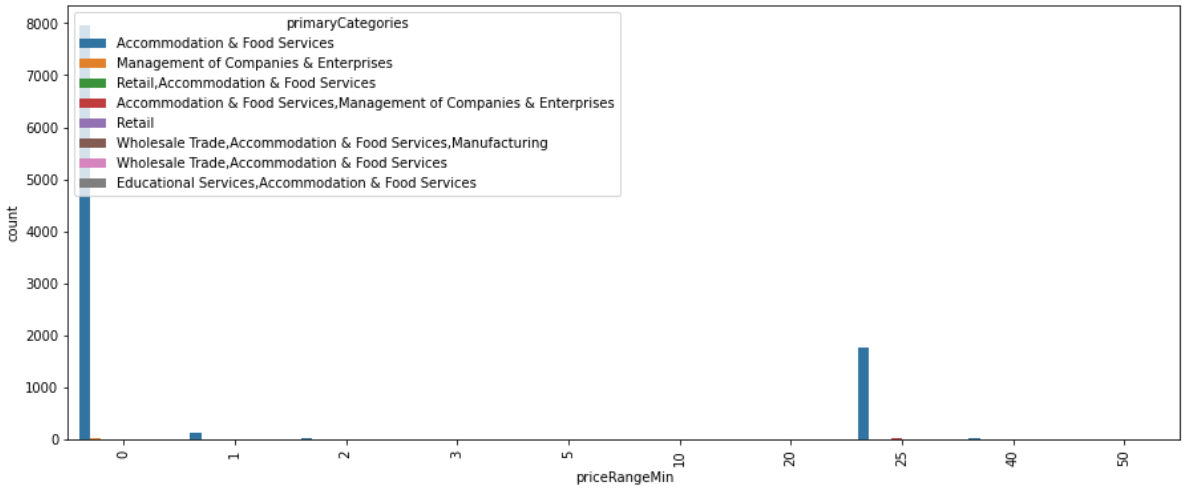
In [86]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMax', hue = 'city', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



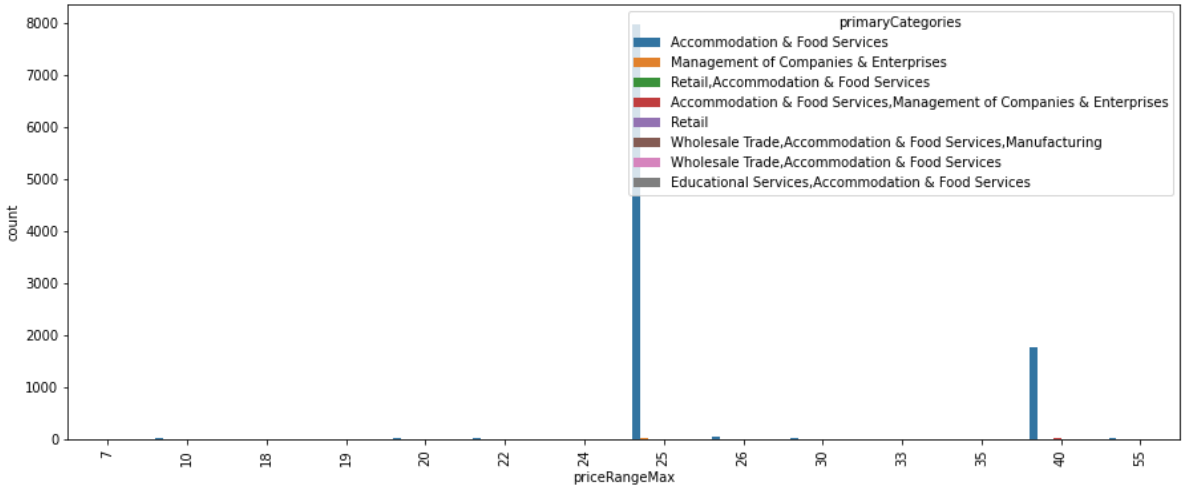
In [88]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMin', hue = 'primaryCategories', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



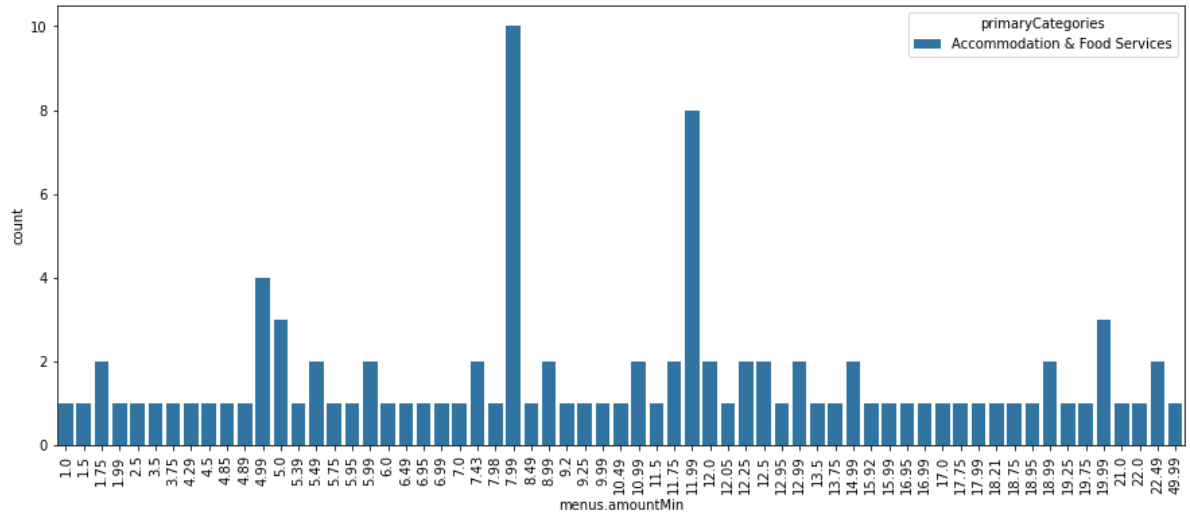
In [89]:

```
plt.figure(figsize=(15,6))
sns.countplot('priceRangeMax', hue = 'primaryCategories', data = pizza_data)
plt.xticks(rotation = 90)
plt.show()
```



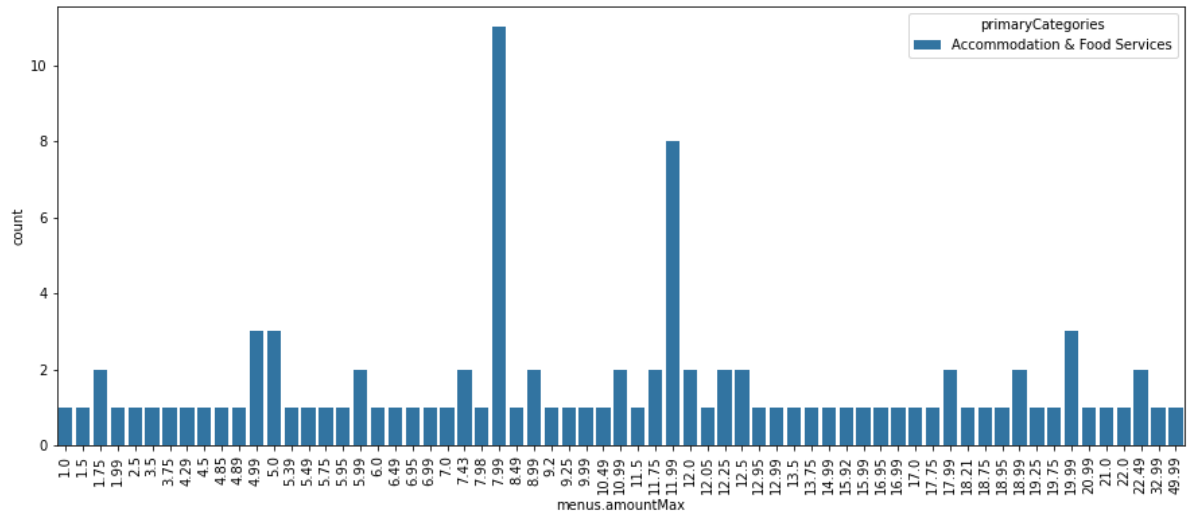
In [91]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMin', hue = 'primaryCategories', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



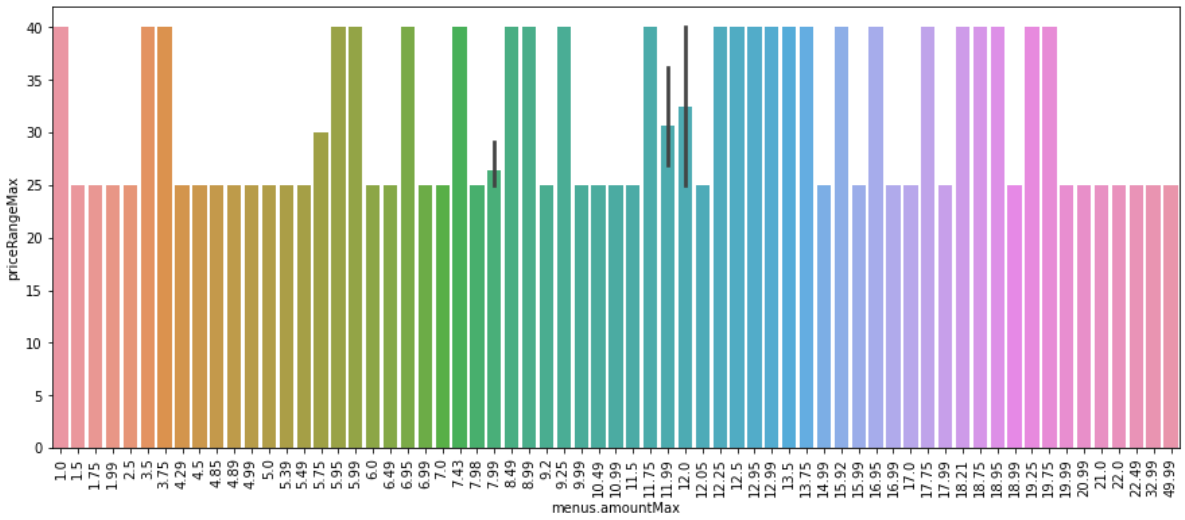
In [92]:

```
plt.figure(figsize=(15,6))
sns.countplot('menus.amountMax', hue = 'primaryCategories', data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



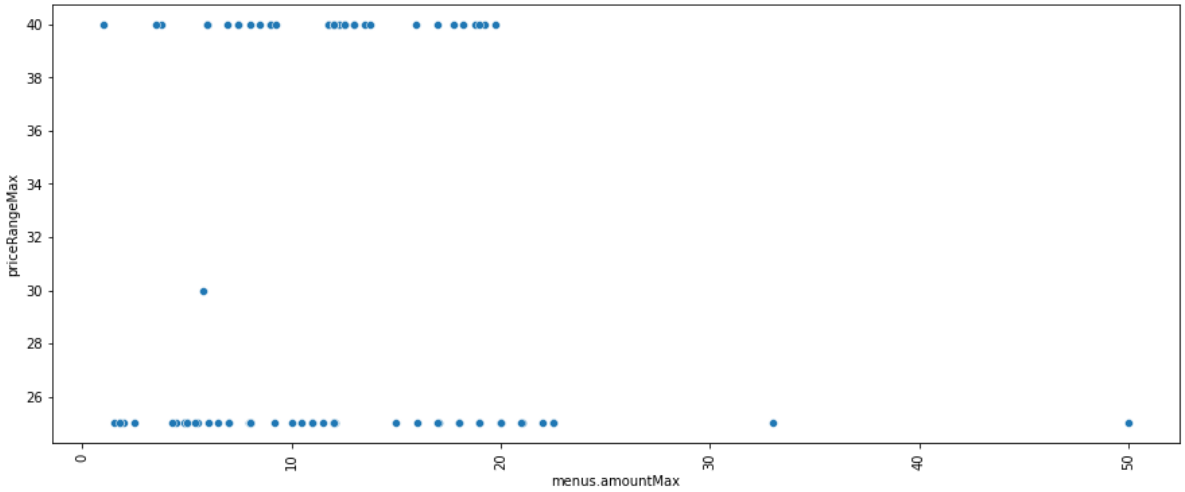
In [93]:

```
plt.figure(figsize=(15,6))
sns.barplot(y = 'priceRangeMax', x = 'menus.amountMax',
            data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



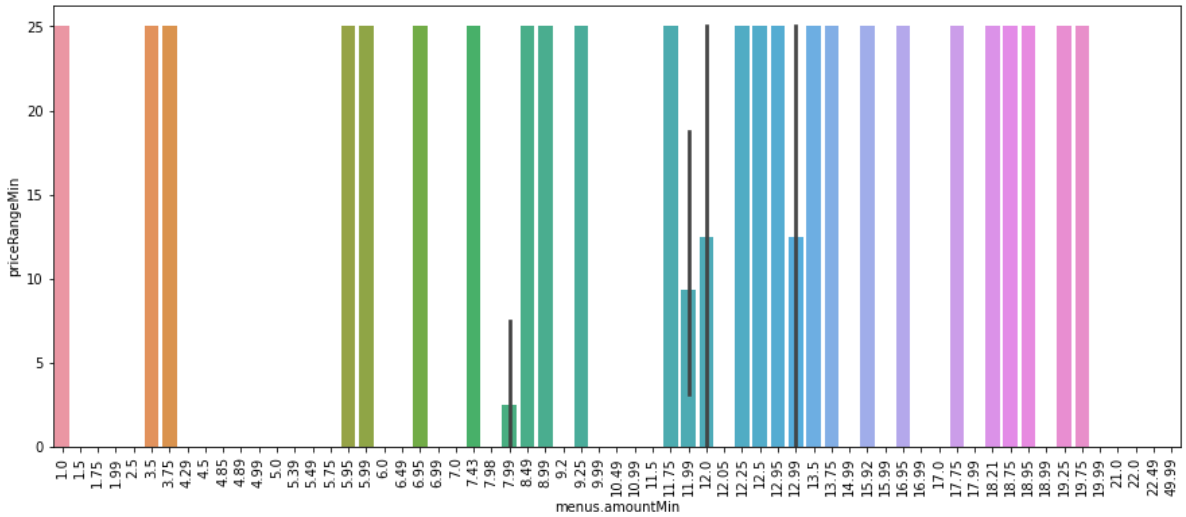
In [94]:

```
plt.figure(figsize=(15,6))
sns.scatterplot(y = 'priceRangeMax', x = 'menus.amountMax',
            data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



In [95]:

```
plt.figure(figsize=(15,6))
sns.barplot(y = 'priceRangeMin', x = 'menus.amountMin',
            data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```



In [96]:

```
plt.figure(figsize=(15,6))
sns.scatterplot(y = 'priceRangeMin', x = 'menus.amountMin',
            data = pizza_data.head(100))
plt.xticks(rotation = 90)
plt.show()
```