1 Project 1-Airbnb New York

Reading in the AN_NYC_2019.csv data.
The first few lines of the dataset will be shown.

In [1]:

```
from datetime import datetime, timedelta, date
import pandas as pd
%matplotlib inline
from sklearn.metrics import classification_report, confus
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.cluster import KMeans
import sklearn
from sklearn.model_selection import KFold, cross_val_scor
```

In [2]:

```
df = pd.read_csv("C:/Users/lenovo/Desktop/AB_NYC_2019.csv
df.head()
```

	id	name	host_id	host_name	neighbourhood_group	nei
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Ken
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Mid
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harl
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clin
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East

In [3]:

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

```
id                                  0
name                               16
host_id                             0
host_name                          21
neighbourhood_group                 0
neighbourhood                       0
latitude                            0
longitude                           0
room_type                           0
price                               0
minimum_nights                      0
number_of_reviews                   0
last_review                     10052
reviews_per_month               10052
calculated_host_listings_count      0
availability_365                    0
dtype: int64
```

In [4]:

```
df.describe()
```

	id	host_id	latitude	longitude	price
count	4.889500e+04	4.889500e+04	48895.000000	48895.000000	48895.00000
mean	1.901714e+07	6.762001e+07	40.728949	-73.952170	152.720687
std	1.098311e+07	7.861097e+07	0.054530	0.046157	240.154170
min	2.539000e+03	2.438000e+03	40.499790	-74.244420	0.000000
25%	9.471945e+06	7.822033e+06	40.690100	-73.983070	69.000000
50%	1.967728e+07	3.079382e+07	40.723070	-73.955680	106.000000
75%	2.915218e+07	1.074344e+08	40.763115	-73.936275	175.000000
max	3.648724e+07	2.743213e+08	40.913060	-73.712990	10000.00000

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
id                                48895 non-null in
t64
name                              48879 non-null ob
ject
host_id                           48895 non-null in
t64
host_name                         48874 non-null ob
ject
neighbourhood_group               48895 non-null ob
ject
neighbourhood                     48895 non-null ob
ject
latitude                          48895 non-null fl
oat64
longitude                         48895 non-null fl
oat64
room_type                         48895 non-null ob
ject
price                             48895 non-null in
t64
minimum_nights                    48895 non-null in
t64
number_of_reviews                 48895 non-null in
t64
last_review                       38843 non-null ob
ject
reviews_per_month                 38843 non-null fl
oat64
calculated_host_listings_count    48895 non-null in
t64
availability_365                  48895 non-null in
t64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB
```

1.1 Data Processing

In [6]:

```
#Based on the dataset we check, there are some blank in t
df['reviews_per_month'].fillna(0, inplace= True)
df['name'].fillna('#', inplace=True)
df['host_name'].fillna('*', inplace=True)
```

In [7]:

```
#Drop the two columns we do not need: last-review, and ca
df.drop(['last_review'],axis=1,inplace=True)
df.drop(['calculated_host_listings_count'],axis=1,inplace
```

In [8]:

```
df.head()
```

	id	name	host_id	host_name	neighbourhood_group	nei
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Ken
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Mid
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harl
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clin
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East

In [9]:

```
#Check whether there is null values in the dataset
df.isnull().sum()
```

```
id                      0
name                    0
host_id                 0
host_name               0
neighbourhood_group     0
neighbourhood           0
latitude                0
longitude               0
room_type               0
price                   0
minimum_nights          0
number_of_reviews       0
reviews_per_month       0
availability_365        0
dtype: int64
```

In [10]:

```
df['neighbourhood_group'].value_counts()
```

```
Manhattan        21661
Brooklyn         20104
Queens            5666
Bronx             1091
Staten Island      373
Name: neighbourhood_group, dtype: int64
```

In [11]:

```
df['room_type'].value_counts()
```

```
Entire home/apt    25409
Private room       22326
Shared room         1160
Name: room_type, dtype: int64
```

In [12]:

```
df['price'].describe()
```

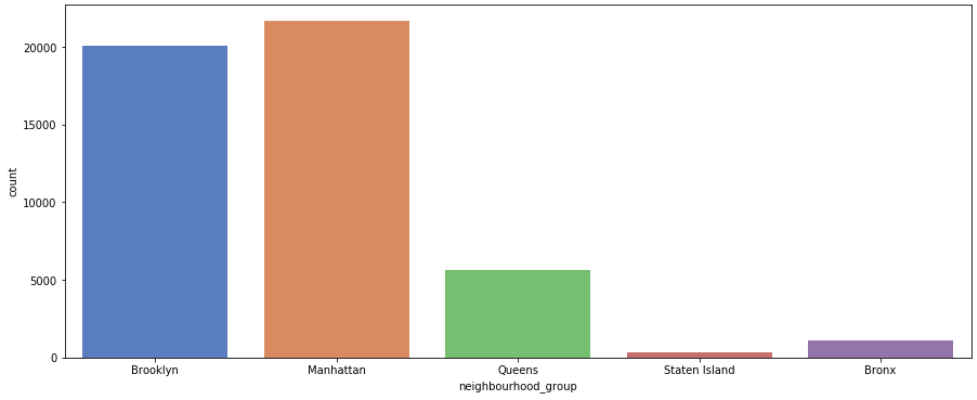
```
count    48895.000000
mean       152.720687
std        240.154170
min          0.000000
25%         69.000000
50%        106.000000
75%        175.000000
max      10000.000000
Name: price, dtype: float64
```

The following plot represents the count of Airbnb's in the different neighbourhood groups. From the plot, we can easily visualize that maximum number of houses or apartments listed on Airbnb is in

1.2 Personas

In [13]:

```
f,ax = plt.subplots(figsize=(15,6))
ax = sns.countplot(df.neighbourhood_group,palette="muted"
plt.show()
```



From the Chart, we can see that the listings of room offer most in Manhattan of NYC.

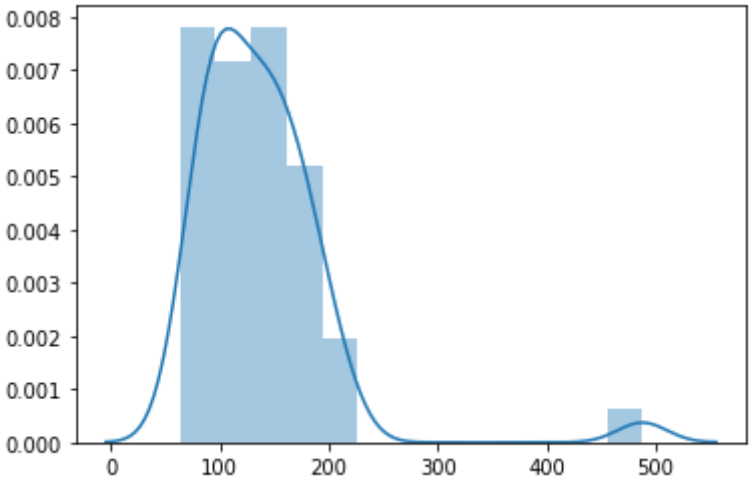
1.2.1 Price Distribution of Airbnb in Brooklyn

The price distribution of Airbnb in Brooklyn averages mostly

around 70-500 dollars per night depending upon the neighbourhood.

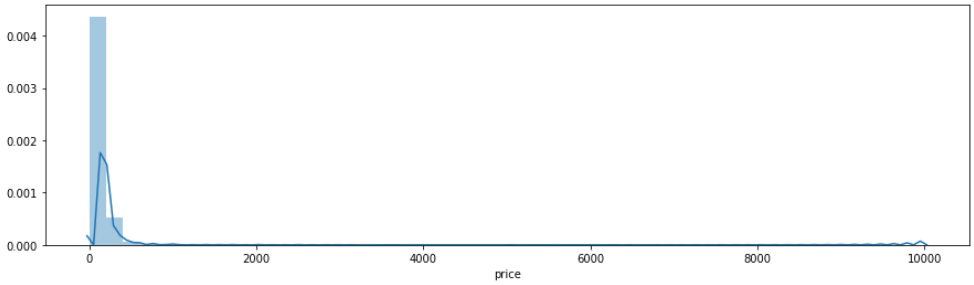
In [14]:

```
df1 = df[df.neighbourhood_group == "Brooklyn"][["neighbou
d = df1.groupby("neighbourhood").mean()
sns.distplot(d)
plt.show()
```



In [15]:

```
f,ax = plt.subplots(figsize=(15,4))
df1 = df[df.neighbourhood_group=="Brooklyn"]['price']
sns.distplot(df1)
plt.show()
```



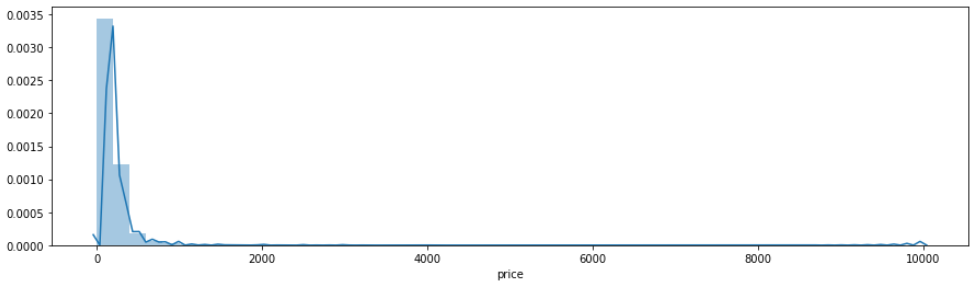
The price distribution of Airbnb in Brooklyn averages around 60-600 dollars per night.

1.2.2 Price Distribution of Airbnb in Manhattan

The price distribution of Airbnb in Manhattan averages around 80-490 dollars per night depending upon the neighbourhood.

In [16]:

```
f,ax = plt.subplots(figsize=(15,4))
df1 = df[df.neighbourhood_group=="Manhattan"]['price']
sns.distplot(df1)
plt.show()
```

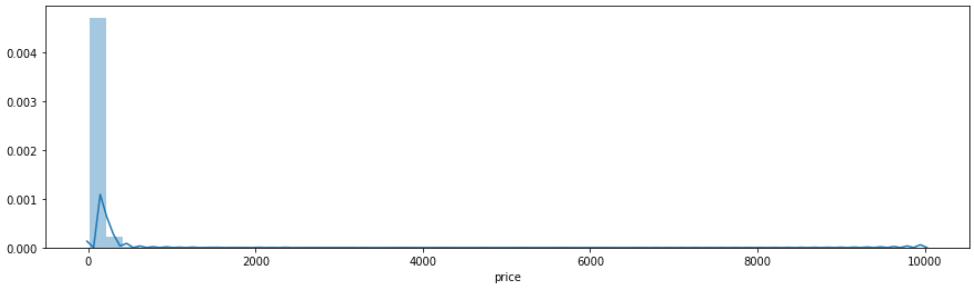


1.2.3 Price Distribution of Airbnb in Queens

The price distribution of Airbnb in Queens averages around 60-280 dollars per night depending upon the neighbourhood.

In [17]:

```
f,ax = plt.subplots(figsize=(15,4))
df1 = df[df.neighbourhood_group=="Queens"]['price']
sns.distplot(df1)
plt.show()
```

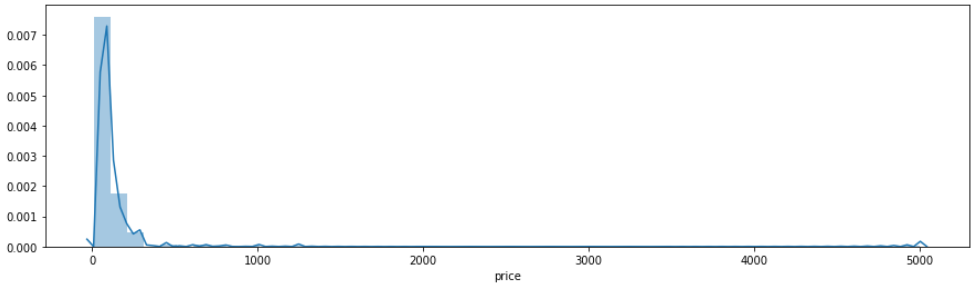


1.2.4 Price Distribution of Airbnb in Staten Island

The price distribution of Airbnb in Staten Islands averages around 50-800 dollars per night depending upon the neighbourhood.

In [18]:

```
f,ax = plt.subplots(figsize=(15,4))
df1 = df[df.neighbourhood_group=="Staten Island"]['price']
sns.distplot(df1)
plt.show()
```

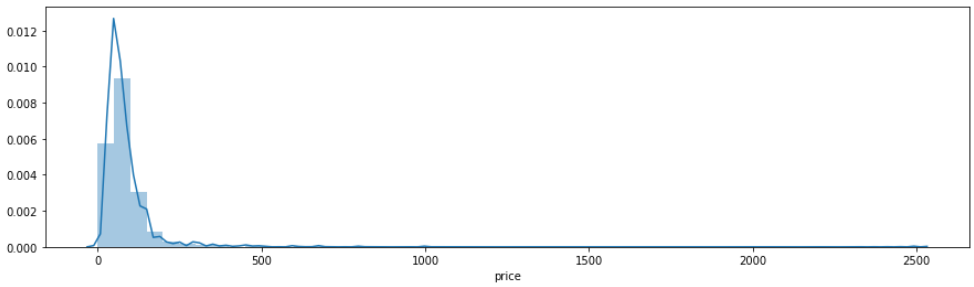


1.2.5 Price Distribution of Airbnb in Bronx

The price distribution of Airbnb in Bronx averages around 50-450 dollars per night depending upon the neighbourhood.

In [19]:

```
f,ax = plt.subplots(figsize=(15,4))
df1 = df[df.neighbourhood_group=="Bronx"]['price']
sns.distplot(df1)
plt.show()
```



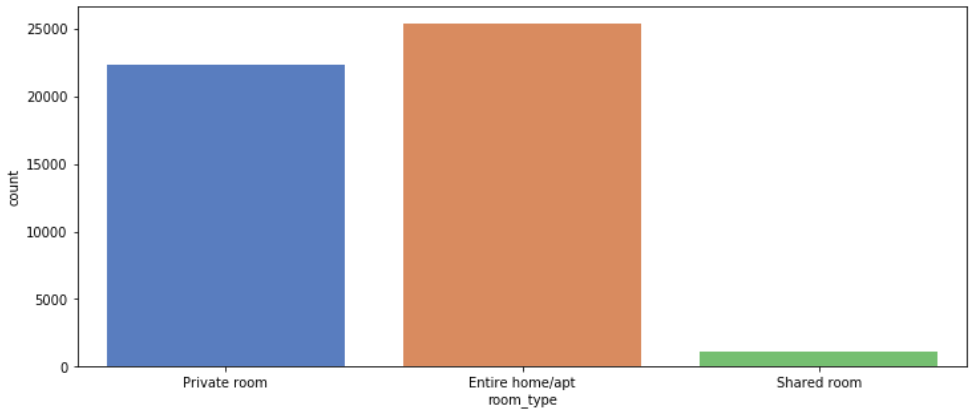
From the price distribution, we can see that Manhattan is most density of room offers and in a relative low price.

1.2.6 Room Types

The chart shows that the three type of rooms: Private room Entire home and Shared room. The most welcomed room_type listed on Airbnb are private rooms and entire home and apartments and shared rooms are listed in a small number on Airbnb.

In [20]:

```
f,ax = plt.subplots(figsize=(12,5))
ax = sns.countplot(df.room_type,palette="muted")
plt.show()
```

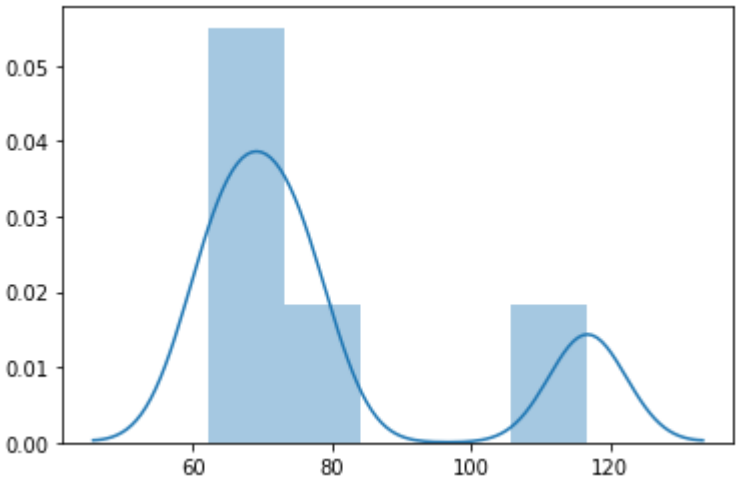


1.2.7 Price Distribution of Private rooms

Private rooms on average are from 60-120 dollars per night on an average depending on the neghbourhood group locations.

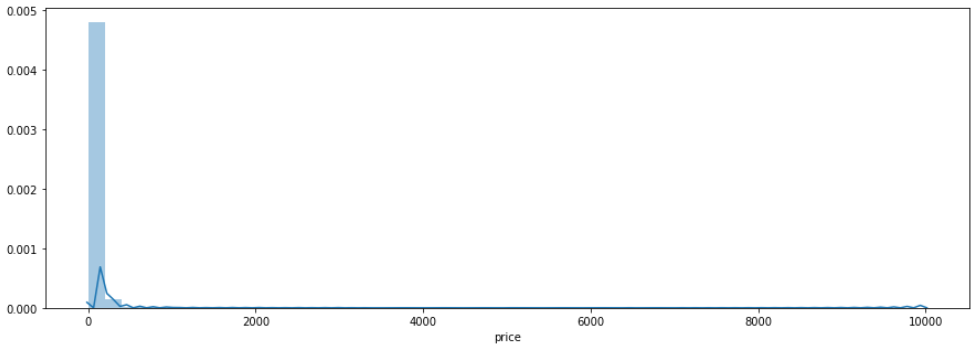
In [21]:

```
df1 = df[df.room_type == "Private room"][["neighbourhood_
d = df1.groupby("neighbourhood_group").mean()
sns.distplot(d)
plt.show()
```



In [22]:

```
df1 = df[df.room_type=='Private room']['price']
f,ax = plt.subplots(figsize=(15,5))
ax = sns.distplot(df1)
plt.show()
```

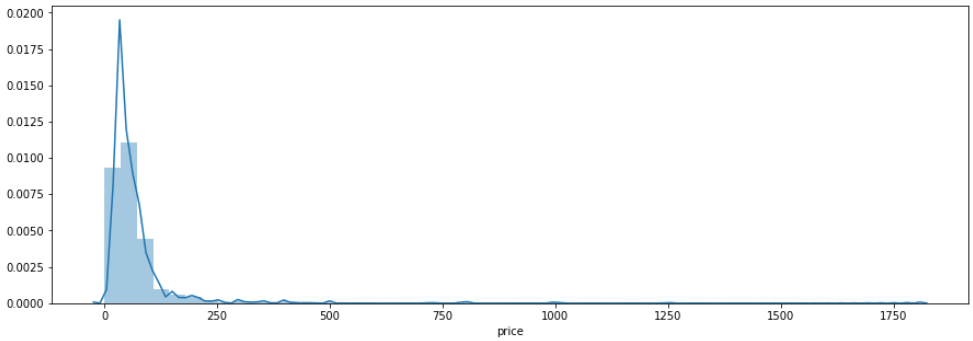


1.2.8 Price Distribution of Shared rooms

Most of the shared rooms have the price range between 50-70 dollars per night depending upon the neighbourhood groups.And the price range mostly distributed from 50 to 110.

In [23]:

```
df1 = df[df.room_type=='Shared room']['price']
f,ax = plt.subplots(figsize=(15,5))
ax = sns.distplot(df1)
plt.show()
```



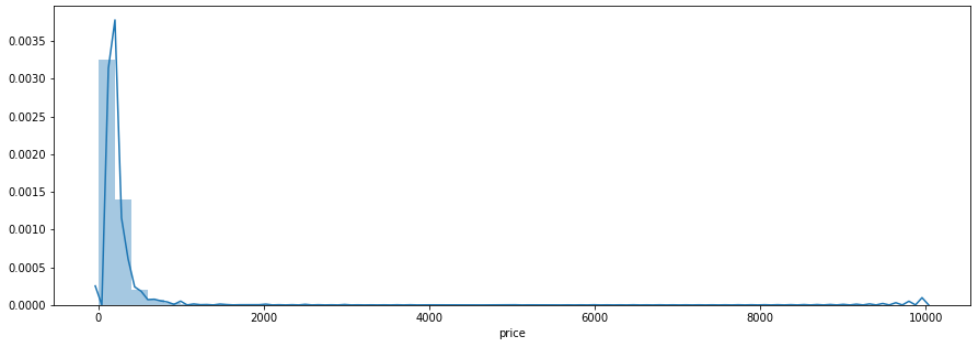
1.2.9 Price Distribution of Entire home/apt

The average price of entire home or apartment varies from 120-250 dollars per night depending upon the neighbourhood they

given house is situated.

In [24]:

```
df1 = df[df.room_type=='Entire home/apt']['price']
f,ax = plt.subplots(figsize=(15,5))
ax = sns.distplot(df1)
plt.show()
```



The room type, Entire home/apt is the relatively expensive than the other two room types.

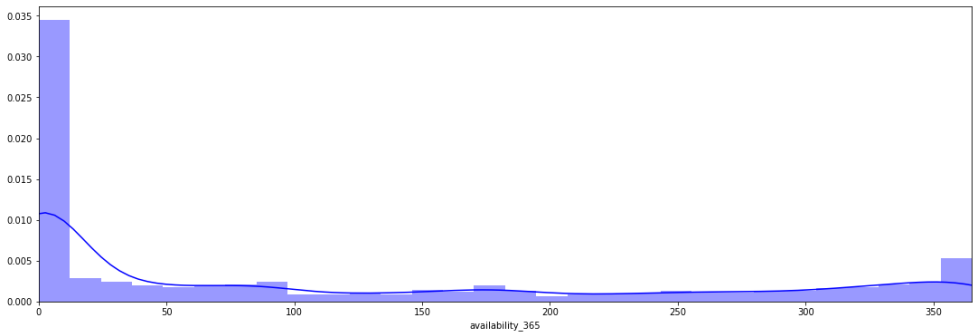
1.2.10 Representing availability of rooms

This chart shows the distribution of the availability of every housing in listing

In [25]:

```
fig, axes = plt.subplots(1,1,figsize=(18.5, 6))
sns.distplot(df['availability_365'], rug=False, kde=True,
axes.set_xlabel('availability_365')
axes.set_xlim(0, 365)
```

```
(0, 365)
```

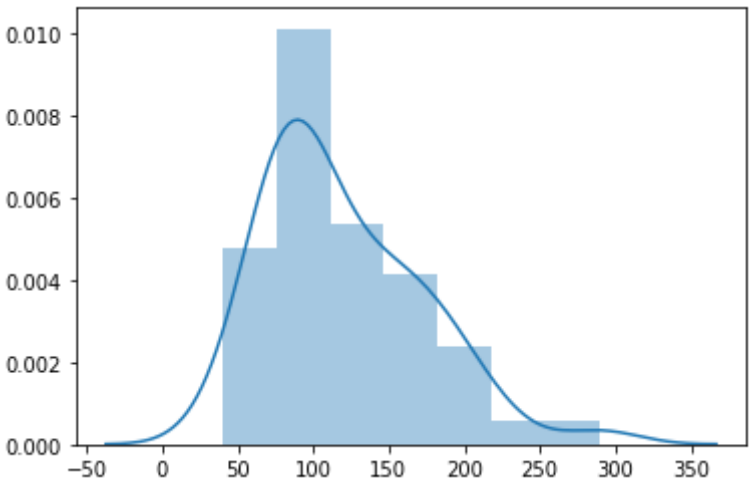


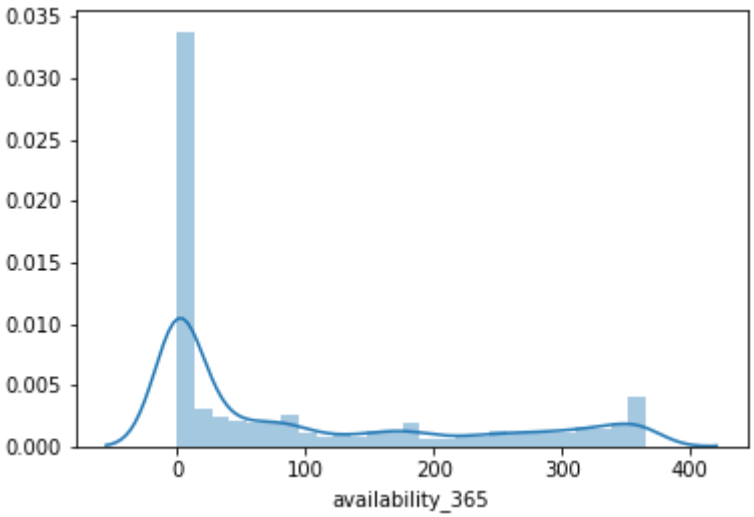
Mostly clusted within 50 days

1.2.10.1 Availability Days distribution of Airbnb in Brooklyn

In [26]:

```
df1 = df[df.neighbourhood_group == "Brooklyn"][["neighbou
d = df1.groupby("neighbourhood").mean()
sns.distplot(d)
plt.show()
f,ax = plt.subplots(figsize=(6,4))
df1 = df[df.neighbourhood_group=="Brooklyn"]['availabilit
sns.distplot(df1)
plt.show()
```

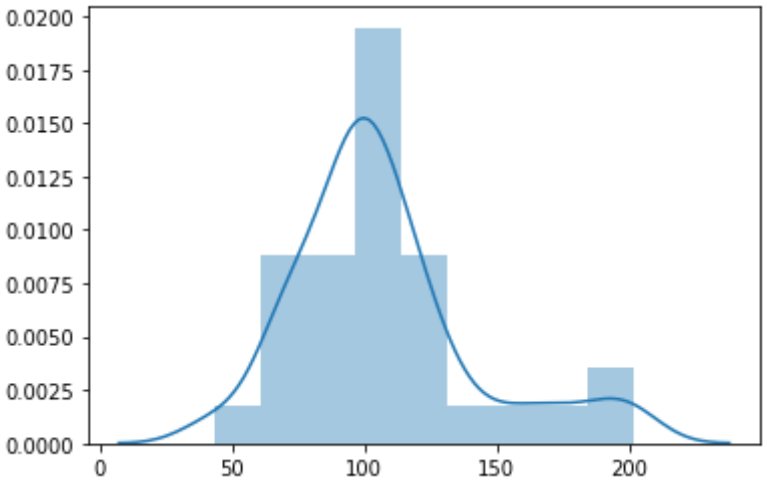


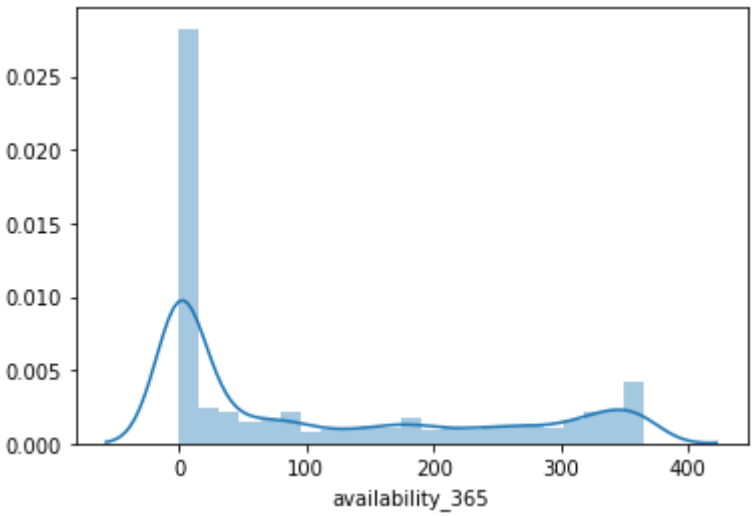


1.2.10.2 Availability Days distribution of Airbnb in Manhattan

In [27]:

```
df2 = df[df.neighbourhood_group == "Manhattan"][["neighbou
d = df2.groupby("neighbourhood").mean()
sns.distplot(d)
plt.show()
f,ax = plt.subplots(figsize=(6,4))
df2 = df[df.neighbourhood_group=="Manhattan"]['availabili
sns.distplot(df2)
plt.show()
```

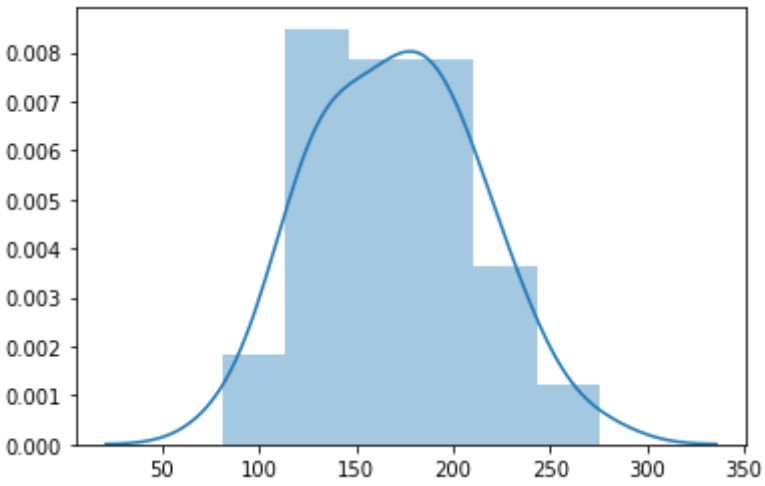


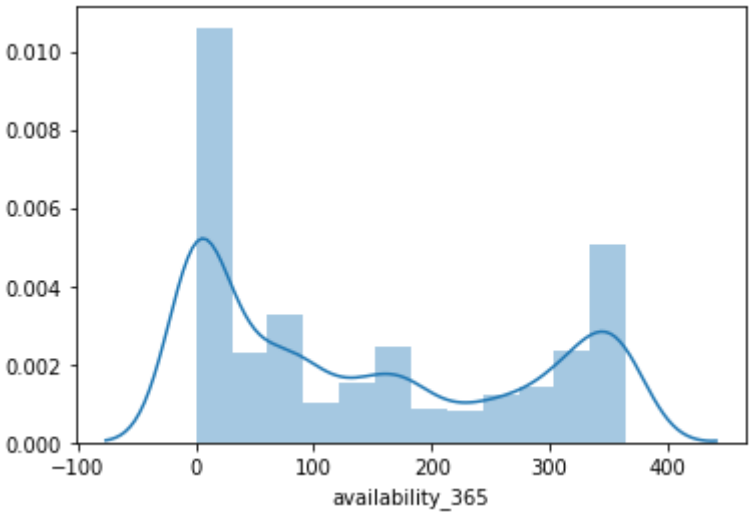


1.2.10.3 Availability Days distribution of Airbnb in Queens

In [28]:

```
df3 = df[df.neighbourhood_group == "Queens"][["neighbourh
d = df3.groupby("neighbourhood").mean()
sns.distplot(d)
plt.show()
f,ax = plt.subplots(figsize=(6,4))
df3 = df[df.neighbourhood_group=="Queens"]['availability_
sns.distplot(df3)
plt.show()
```

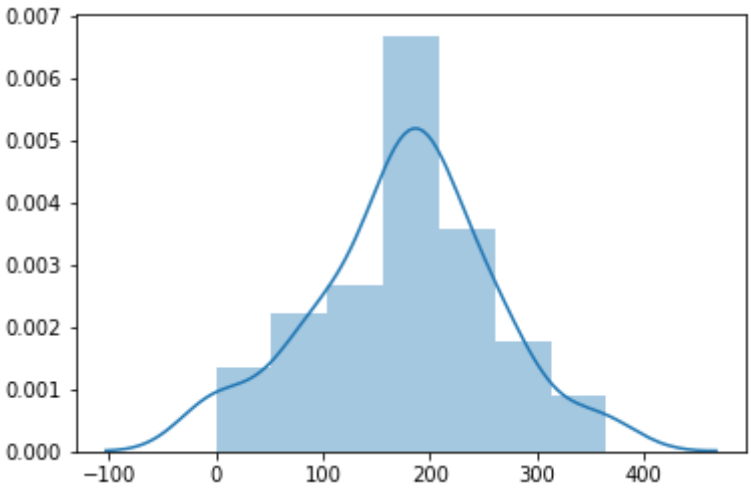


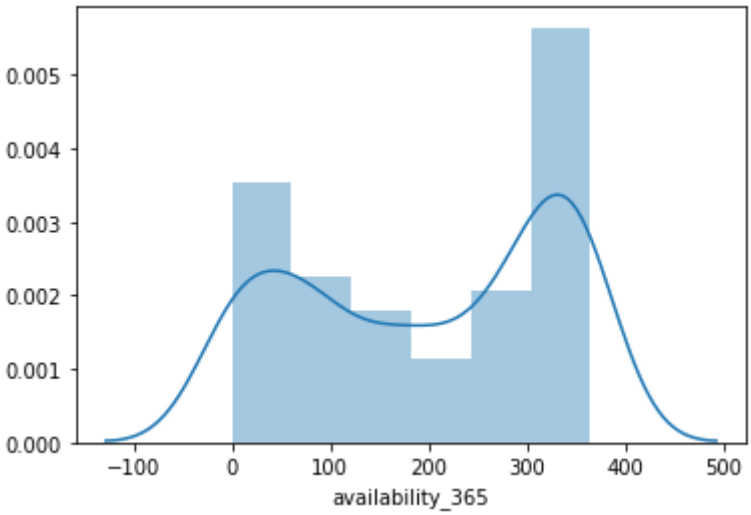


1.2.10.4 Availability Days distribution of Airbnb in Staten Island

In [29]:

```
df4 = df[df.neighbourhood_group == "Staten Island"][["neig
d = df4.groupby("neighbourhood").mean()
sns.distplot(d)
plt.show()
f,ax = plt.subplots(figsize=(6,4))
df4 = df[df.neighbourhood_group=="Staten Island"]['availal
sns.distplot(df4)
plt.show()
```





From the charts, the most least availability areas are Brooklyn and Manhanttan, and the most-available-days areas goes to Staten Island.

1.3 Insight

Based on the plots above,we found that the most revenue,which is (price*(365-availability_days)/365) goes to Manhanttan for neightborhood_group feature, and Entire room/Apt for the room_type feature. We assume the host who have Entire room/Apt are the high-value customers, and rest are low-value customers.

Therefore, we assume: null hypothesis: there is no difference between high-value and low-value hosts in avg_revenue. Alternative hypothesis: there is such difference between high-value and low-value hosts in avg_revenue.

1.4 Hypothesis Test

Before doing the hypothesis,we need to calculate the average daily revenue of each housing,this is an important feature to build the regression.On the other hand,we define the high value host and the low value host by the room type,it is an important feature to do hypothese.

In [3]:

```
df['avg_revenue']  = df['price']*(365-df['availability_36
df_hv = df.loc[df.room_type=='Entire home/apt', 'segment'
df_lv = df.loc[df.room_type!='Entire home/apt', 'segment'
df.head()
```

	id	name	host_id	host_name	neighbourhood_group	nei
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Ken
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Mid
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harl
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clin
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East

All the data are randomly assigned to make use of 90% to control,and test on remain 10% of the data;

In [4]:

```
df_test_hv = df[df.segment == 'high-value'].sample(frac=0
df_test_hv.head()
```

	id	name	host_id	host_name	neighbourhood_gro
42130	32714435	Heart of NYC-deluxe 1BR apt with gorgeous views	30283594	Kara	Manhattan
19565	15648096	Spacious 2 bedroom close to Manhattan	100971588	NaN	Bronx
28110	21935551	Super clean / centrally located extra large st...	88713943	Lee	Manhattan
23217	18795229	Ground Floor Studio	130822245	Joe Berat	Queens
34793	27580328	Large comfortable home 2 blocks from Times Square	207833780	Genisley	Manhattan

In [5]:

```
df_test_lv = df[df.segment == 'low-value'].sample(frac=0.1
df_test_lv.head()
```

	id	name	host_id	host_name	neighbourhood_gr
35727	28350560	Spacious Bdrm in N.Y.C. (30 mins to Midtown)	37252076	Iara	Queens
35563	28226667	BP-BEAUTIFUL COZY ROOM FOR 2 NEAR MANHATTAN ...	213208277	Darry	Brooklyn
15807	12791778	Newly renovated williamsburg bedroom near subway	17239096	David	Brooklyn
32804	25837179	ASTORIA in QUEENS	194130534	Miryung	Queens
37736	29902956	#Private Room & Bath 30 min to Wall st NYC	224850313	Roman	Brooklyn

In [6]:

```
df_test= pd.concat([df_test_hv,df_test_lv], axis=0)
df_test.head()
```

	id	name	host_id	host_name	neighbourhood_gro
42130	32714435	Heart of NYC-deluxe 1BR apt with gorgeous views	30283594	Kara	Manhattan
19565	15648096	Spacious 2 bedroom close to Manhattan	100971588	NaN	Bronx
28110	21935551	Super clean / centrally located extra large st...	88713943	Lee	Manhattan
23217	18795229	Ground Floor Studio	130822245	Joe Berat	Queens
34793	27580328	Large comfortable home 2 blocks from Times Square	207833780	Genisley	Manhattan

In [7]:

```
df_control = df[~df.id.isin(df_test.id)]
df_control.head()
```

	id	name	host_id	host_name	neighbourhood_group	nei
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Ken
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Mid
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harl
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clin
5	5099	Large Cozy 1 BR Apartment In Midtown East	7322	Chris	Manhattan	Mur

In [8]:

```
test_results = df_test.avg_revenue
control_results = df_control.avg_revenue
```

In [9]:

```
from scipy import stats
test_result = stats.ttest_ind(test_results, control_resul
print(test_result)
```

```
Ttest_indResult(statistic=1.2173529977342428, pvalu
e=0.2234758134587995)
```

In [11]:

```
df_test['group'] = 'test'
df_control['group'] = 'control'
df_customers = pd.concat([df_test,df_control],axis=0)
df_customers
```

```
C:\Users\lenovo\Anaconda3\lib\site-packages\ipykern
el_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice fr
om a DataFrame.
Try using .loc[row_indexer,col_indexer] = value ins
tead

See the caveats in the documentation: http://pandas.py
data.org/pandas-docs/stable/user_guide/indexing.html#returnin
g-a-view-versus-a-copy (http://pandas.pydata.org/pandas-docs/
stable/user_guide/indexing.html#returning-a-view-versus-a-cop
y)
```

	id	name	host_id	host_name	neighbourhood_
42130	32714435	Heart of NYC-deluxe 1BR apt with gorgeous views	30283594	Kara	Manhattan
19565	15648096	Spacious 2 bedroom close to Manhattan	100971588	NaN	Bronx
28110	21935551	Super clean / centrally located extra large st...	88713943	Lee	Manhattan
23217	18795229	Ground Floor Studio	130822245	Joe Berat	Queens
34793	27580328	Large comfortable home 2 blocks from Times Square	207833780	Genisley	Manhattan
...	...	...	...	...	...
48889	36484363	QUIT PRIVATE HOUSE	107716952	Michael	Queens

	id	name	host_id	host_name	neighbourhood_
48890	36484665	Charming one bedroom - newly renovated rowhouse	8232441	Sabrina	Brooklyn
48891	36485057	Affordable room in Bushwick/East Williamsburg	6570630	Marisol	Brooklyn
48892	36485431	Sunny Studio at Historical Neighborhood	23492952	Ilgar & Aysel	Manhattan
48894	36487245	Trendy duplex in the very heart of Hell's Kitchen	68119814	Christophe	Manhattan

48895 rows × 19 columns

In [70]:

```
import statsmodels.formula.api as smf
from statsmodels.stats.anova import anova_lm
model = smf.ols(formula='avg_revenue ~ segment + group ',
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:            avg_revenue   R-squared:
0.052
Model:                            OLS   Adj. R-squa
red:                  0.052
Method:                 Least Squares   F-statisti
c:                     1354.
Date:                Tue, 07 Apr 2020   Prob (F-sta
tistic):               0.00
Time:                        18:17:42   Log-Likelih
ood:            -3.1865e+05
No. Observations:               48895   AIC:
6.373e+05
Df Residuals:                   48892   BIC:
6.373e+05
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                             coef    std err
t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept               134.9557      1.056    127.7
67      0.000     132.885     137.026
segment[T.low-value]   -77.0676      1.482    -52.0
07      0.000     -79.972     -74.163
group[T.test]            5.0771      2.468      2.0
57      0.040       0.240       9.914
==============================================================================
Omnibus:                   121528.240   Durbin-Wats
on:                  1.865
Prob(Omnibus):                  0.000   Jarque-Bera
(JB):       3141028278.481
```

```
Skew:                          26.819   Prob(JB):
0.00
Kurtosis:                    1243.520   Cond. No.
3.82
==============================================================
============================

Warnings:
[1] Standard Errors assume that the covariance matr
ix of the errors is correctly specified.
```

1.4.1 Conclusion

Acording to the result of the above model, this column " P>|t|" value shows that we can conclude that null hypothesis: there is no difference between high-value and low-value hosts in avg_revenue is not true, and we can accept alternative hypothesis: there is such difference between high-value and low-value hosts in avg_revenue.

1.5 Suggestion based on our insight:

To improve users significant and experience

1. We suggusted that to improve the privacy and comfort levels for private and shared rooms to attract more customers and make them stay longer.

2. We can give any coupons for those customers who order the private rooms and shared rooms for more than 5 days or more. This can be increase the revenue of private rooms and shared rooms