



Airbnb Analysis

Load Library

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

Load Data

```
In [2]: Airbnb = pd.read_csv(r"C:\Users\dell\OneDrive\Desktop\Airbnb_Open_Data.csv")
```

```
In [3]: # read Sample 15 rows
Airbnb.sample(15)
```

Out[3]:

	id	NAME	host id	host_identity_verified	host name
38352	22183146	Smart Studio in Trendy Tribeca, Indoor pool + ...	47388084852	verified	Blueground
88707	49994212	Cheap \$45/night@15 minuets to Mahattaton	258866933	verified	Jazzy
46419	26638550	Cozy house in Woodside, comfortable are!!	22642120149	unconfirmed	Shogo
63338	35982913	Lovely Room Private BATHROOM	29811306211	verified	Kristina & Jack
84215	47513281	Luxury 2200ft ² 4Bed in East Village	58268536955	verified	Jonty
76524	43265541	Lux 2 Bedroom NYC Apt on the Hudson River!	59872864093	unconfirmed	Kara
49087	28112086	Amazing private bedroom next to Central Pk & Sub	25331393151	unconfirmed	Eva
100669	56600825	Bright and Cozy 1BR in Crown Heights	36114607787	verified	Morgan
6293	4476960	Hamilton Hts Beauty, Manhattan 1 BR	58185369523	unconfirmed	Kumar
35291	20492555	Welcome Home To Ridgewood!	50985804096	verified	Alejandro

	id	NAME	host id	host_identity_verified	host name
80292	45346608	Cosy bedroom in prime Williamsburg	1450502892	unconfirmed	Jonathan
57282	32638185	Surfers Dream - Brand New renovated home-KING...	64153404026	verified	Nathan
88603	49936773	1 bdrm in Park Slope , Bklyn.	20600295222	unconfirmed	Phyllis
66157	37539847	Manhattan Home Base	83382717875	unconfirmed	Richard
18974	11480676	Serene private room in chic Astoria apartment	56088088267	unconfirmed	Samantha

15 rows × 26 columns

Data Cleaning & Preparation

```
In [4]: # shape
Airbnb.shape
```

```
Out[4]: (102599, 26)
```

```
In [5]: Airbnb_Clean = Airbnb.copy()
```

```
In [6]: # Replacing column name
Airbnb_Clean.columns = Airbnb_Clean.columns.str.replace(" ", "_")
```

```
In [7]: # information of Airbnb
Airbnb_Clean.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 102599 entries, 0 to 102598
Data columns (total 26 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     102599 non-null  int64
1   NAME                                  102349 non-null  object
2   host_id                               102599 non-null  int64
3   host_identity_verified                102310 non-null  object
4   host_name                             102193 non-null  object
5   neighbourhood_group                   102570 non-null  object
6   neighbourhood                         102583 non-null  object
7   lat                                   102591 non-null  float64
8   long                                  102591 non-null  float64
9   country                               102067 non-null  object
10  country_code                          102468 non-null  object
11  instant_bookable                      102494 non-null  object
12  cancellation_policy                   102523 non-null  object
13  room_type                             102599 non-null  object
14  Construction_year                     102385 non-null  float64
15  price                                  102352 non-null  object
16  service_fee                           102326 non-null  object
17  minimum_nights                        102190 non-null  float64
18  number_of_reviews                     102416 non-null  float64
19  last_review                           86706 non-null  object
20  reviews_per_month                     86720 non-null  float64
21  review_rate_number                    102273 non-null  float64
22  calculated_host_listings_count        102280 non-null  float64
23  availability_365                       102151 non-null  float64
24  house_rules                           50468 non-null  object
25  license                                2 non-null      object
dtypes: float64(9), int64(2), object(15)
memory usage: 20.4+ MB

```

```

In [8]: for col in Airbnb_Clean.columns:
        print(Airbnb_Clean[col].value_counts())
        print("--"*50)

```

```

id
20305326      2
20305878      2
20306430      2
20306983      2
20307535      2
..
57356371      1
57356923      1
57357476      1
57358028      1
57367417      1
Name: count, Length: 102058, dtype: int64

```

```

-----
NAME
Home away from home          33
Hillside Hotel               30
Water View King Bed Hotel Room 30
Brooklyn Apartment          27
New york Multi-unit building 27
..
For all classical lover's 4   1
"San-Paraíso" 80s curated Retro 3 Bedroom LES Pad 1
SUITE- PRIVATE 1/2 BATH, 4 BEDS IN WILLIAMSBURG!! 1
Spacious bedroom in Washington Heights 1
Stunning Rare Garden Family Home w/Loft! #10303 1
Name: count, Length: 61281, dtype: int64

```

```

-----
host_id
78730595133    2
50358760342    2
89461531863    2
86549670696    2
15675388930    2
..
11038605149    1
29320426760    1
14430571516    1
56457739998    1
60176837202    1
Name: count, Length: 102057, dtype: int64

```

```

-----
host_identity_verified
unconfirmed    51200
verified       51110
Name: count, dtype: int64

```

```

-----
host_name
Michael      881
David        764

```

John	581
Alex	546
Sonder (NYC)	516
...	
Ujunwa	1
Richard Nelson	1
Omri-Dan	1
Yavuz	1
Shreen	1

Name: count, Length: 13190, dtype: int64

neighbourhood_group

Manhattan	43792
Brooklyn	41842
Queens	13267
Bronx	2712
Staten Island	955
brookln	1
manhatan	1

Name: count, dtype: int64

neighbourhood

Bedford-Stuyvesant	7937
Williamsburg	7775
Harlem	5466
Bushwick	4982
Hell's Kitchen	3965
...	
Rossville	3
Gerritsen Beach	3
Fort Wadsworth	2
Glen Oaks	2
Chelsea, Staten Island	1

Name: count, Length: 224, dtype: int64

lat

40.76411	36
40.71813	32
40.76125	28
40.73756	27
40.76106	25
..	
40.73244	1
40.71740	1
40.85389	1
40.85924	1
40.79607	1

Name: count, Length: 21991, dtype: int64

long

-73.99371	44
-73.95350	40
-73.94791	37
-73.95427	37
-73.95677	34

..	
-73.86037	1
-73.82992	1
-73.84460	1
-73.85623	1
-73.81108	1

Name: count, Length: 17774, dtype: int64

country

United States	102067
---------------	--------

Name: count, dtype: int64

country_code

US	102468
----	--------

Name: count, dtype: int64

instant_bookable

False	51474
-------	-------

True	51020
------	-------

Name: count, dtype: int64

cancellation_policy

moderate	34343
----------	-------

strict	34106
--------	-------

flexible	34074
----------	-------

Name: count, dtype: int64

room_type

Entire home/apt	53701
-----------------	-------

Private room	46556
--------------	-------

Shared room	2226
-------------	------

Hotel room	116
------------	-----

Name: count, dtype: int64

Construction_year

2014.0	5243
--------	------

2008.0	5225
--------	------

2006.0	5223
--------	------

2019.0	5201
--------	------

2009.0	5166
--------	------

2020.0	5158
--------	------

2010.0	5155
--------	------

2022.0	5134
--------	------

```
2005.0    5132
2012.0    5131
2003.0    5125
2007.0    5106
2015.0    5094
2017.0    5066
2011.0    5058
2018.0    5057
2021.0    5039
2004.0    5037
2013.0    5018
2016.0    5017
Name: count, dtype: int64
```

```
-----
price
$206      137
$1,056    132
$481      129
$833      128
$573      127
...
$369       58
$923       57
$786       54
$89        53
$187       52
Name: count, Length: 1151, dtype: int64
```

```
-----
service_fee
$41       526
$216      524
$81       519
$177      518
$57       513
...
$73       386
$58       384
$67       381
$10       267
$240      248
Name: count, Length: 231, dtype: int64
```

```
-----
minimum_nights
1.0      25421
2.0      23604
3.0      16113
30.0     11653
4.0       6625
...
-125.0      1
125.0       1
```



```
34.0      1
458.0     1
825.0     1
Name: count, Length: 153, dtype: int64
```

```
-----
number_of_reviews
```

```
0.0      15734
1.0      10408
2.0       7175
3.0       5375
4.0       4151
```

```
...
1010.0     1
446.0      1
490.0      1
400.0      1
300.0      1
```

```
Name: count, Length: 476, dtype: int64
```

```
-----
last_review
```

```
6/23/2019   2443
6/30/2019   2232
7/1/2019    2218
6/24/2019   1510
7/7/2019    1151
```

```
...
2/4/2016     1
7/8/2015     1
2/18/2016    1
2/5/2015     1
5/21/2022    1
```

```
Name: count, Length: 2477, dtype: int64
```

```
-----
reviews_per_month
```

```
0.03      1666
0.05      1490
1.00      1455
0.04      1270
0.09      1269
```

```
...
11.58      1
7.80       1
7.73       1
24.49      1
33.08      1
```

```
Name: count, Length: 1016, dtype: int64
```

```
-----
review_rate_number
```

```
5.0      23369
4.0      23329
```

```
3.0    23265
2.0    23098
1.0     9212
Name: count, dtype: int64
```

```
-----
-----
calculated_host_listings_count
```

```
1.0    63429
2.0    14445
3.0     6577
4.0     3552
5.0     1995
```

```
...
86.0    21
22.0    20
55.0    15
46.0     9
80.0     7
```

```
Name: count, Length: 78, dtype: int64
```

```
-----
-----
availability_365
```

```
0.0    23544
365.0   2500
364.0   1168
89.0    751
1.0     734
```

```
...
366.0    33
-6.0     32
391.0    30
401.0    30
3677.0    1
```

```
Name: count, Length: 438, dtype: int64
```

```
-----
-----
house_rules
```

```
#NAME?
```

```
2712
```

House Rules 1. Check-in is 4 pm local time. If the unit is ready earlier, we'll let you know. Check-out is normally 11 am local time, but we'd be happy to extend it as long as we don't have a cleaning scheduled. Just let us know. 2. All bookings require a security deposit of at least \$300, which will be refunded within 7 days of your check-out. 3. For security measures we require all guests to provide proof of identification through ID verification on our own website. In order to check in we'll need a photo of your ID. 4. Our cancellation policy is as shown on our ad and defined by the site you are booking through. 5. Don't let \$300 go up in smoke. There's no smoking allowed in any Flatbook and a \$300 fine for breaking this rule. 6. Unfortunately we don't allow pets in any of our apartments. 7. Unless you're staying in one of our specialty apartments, we don't allow parties or excessive noise. 8. If we find the place very messy, we have to charge an extra \$40 for every extra hour o 904

Please remember that this is a residential building. The building rules, apartment occupancy restrictions and the year-round tenants must be respected.

881

No Smoking No Pets

814

Please remember that this is a residential building. The building rules, apartment occupancy restrictions and the year-round tenants must be respected. Parties, pets and smoking are not allowed under any circumstances and incur significant fees.

552

...

- The price is based on double occupancy. \$50. per person per night. - Check prices with host before booking - The price is based on double occupancy \$300 per person per week. And \$1000 monthly per person. - Single room \$64 per night. Weekly, \$350. Monthly \$1213

1

- Quiet time after 24

1

15 guest maximum. Smoking isn't permitted and unfortunately neither are pets. Sorry. Quiet hours 9p-8a.

1

15 nights minimum stay No pets at any time No smoking Please be mindful of the furniture and personal belongings, it was all furnished with designer pieces and hard to replace things.

1

16 nights minimum stay, no smoking or pets.

1

Name: count, Length: 1976, dtype: int64

license

41662/AL 2

Name: count, dtype: int64

```
In [9]: for col in Airbnb_Clean.columns:
        print(Airbnb_Clean[col].unique())
        print("--"*50)
```

[1001254 1002102 1002403 ... 57366313 57366865 57367417]

['Clean & quiet apt home by the park' 'Skylit Midtown Castle'
'THE VILLAGE OF HARLEM....NEW YORK !' ...
'Elmhurst 1st Fl Rightl Bedroom'
'Elmhurst 1st Floor BR w/ Pvt. Bath&Balcony'
'Parisian Style Apartment in Heart of Brooklyn']

[80014485718 52335172823 78829239556 ... 23801060917 15593031571
93578954226]

['unconfirmed' 'verified' nan]

['Madaline' 'Jenna' 'Elise' ... 'Anna B' 'Natalie And Dan' 'Apostle John']

['Brooklyn' 'Manhattan' 'brookln' 'manhatan' 'Queens' nan 'Staten Island'
'Bronx']

['Kensington' 'Midtown' 'Harlem' 'Clinton Hill' 'East Harlem'
'Murray Hill' 'Bedford-Stuyvesant' "Hell's Kitchen" 'Upper West Side'
'Chinatown' 'South Slope' 'West Village' 'Williamsburg' 'Fort Greene'
'Chelsea' 'Crown Heights' 'Park Slope' 'Windsor Terrace' 'Inwood'
'East Village' 'Greenpoint' 'Bushwick' 'Flatbush' 'Lower East Side'
'Prospect-Lefferts Gardens' 'Long Island City' 'Kips Bay' 'SoHo'
'Upper East Side' 'Prospect Heights' 'Washington Heights' 'Woodside'
'Brooklyn Heights' 'Carroll Gardens' 'Gowanus' 'Flatlands' 'Cobble Hill'
'Flushing' 'Boerum Hill' 'Sunnyside' 'DUMBO' 'St. George' 'Highbridge'
'Financial District' 'Ridgewood' 'Morningside Heights' 'Jamaica'
'Middle Village' 'NoHo' 'Ditmars Steinway' 'Flatiron District'
'Roosevelt Island' 'Greenwich Village' 'Little Italy' 'East Flatbush'
'Tompkinsville' 'Astoria' 'Clason Point' 'Eastchester' 'Kingsbridge'
'Two Bridges' 'Queens Village' 'Rockaway Beach' 'Forest Hills' 'Nolita'
'Woodlawn' 'University Heights' 'Gravesend' 'Gramercy' 'Allerton' nan
'East New York' 'Theater District' 'Concourse Village' 'Sheepshead Bay'
'Emerson Hill' 'Fort Hamilton' 'Bensonhurst' 'Tribeca' 'Shore Acres'
'Sunset Park' 'Concourse' 'Elmhurst' 'Brighton Beach' 'Jackson Heights'
'Cypress Hills' 'St. Albans' 'Arrochar' 'Rego Park' 'Wakefield' 'Clifton'
'Bay Ridge' 'Graniteville' 'Spuyten Duyvil' 'Stapleton' 'Briarwood'
'Ozone Park' 'Columbia St' 'Vinegar Hill' 'Mott Haven' 'Longwood'
'Canarsie' 'Battery Park City' 'Civic Center' 'East Elmhurst'
'New Springville' 'Morris Heights' 'Arverne' 'Cambria Heights'
'Tottenville' 'Mariners Harbor' 'Concord' 'Borough Park' 'Bayside'
'Downtown Brooklyn' 'Port Morris' 'Fieldston' 'Kew Gardens' 'Midwood'
'College Point' 'Mount Eden' 'City Island' 'Glendale' 'Port Richmond'
'Red Hook' 'Richmond Hill' 'Bellerose' 'Maspeth' 'Williamsbridge'
'Soundview' 'Woodhaven' 'Woodrow' 'Co-op City' 'Stuyvesant Town'
'Parkchester' 'North Riverdale' 'Dyker Heights' 'Bronxdale' 'Sea Gate'
'Riverdale' 'Kew Gardens Hills' 'Bay Terrace' 'Norwood']

'Claremont Village' 'Whitestone' 'Fordham' 'Bayswater' 'Navy Yard'
'Brownsville' 'Eltingville' 'Fresh Meadows' 'Mount Hope'
'Lighthouse Hill' 'Springfield Gardens' 'Howard Beach' 'Belle Harbor'
'Jamaica Estates' 'Van Nest' 'Morris Park' 'West Brighton' 'Far Rockaway'
'South Ozone Park' 'Tremont' 'Corona' 'Great Kills' 'Manhattan Beach'
'Marble Hill' 'Dongan Hills' 'Castleton Corners' 'East Morrisania'
'Hunts Point' 'Neponsit' 'Pelham Bay' 'Randall Manor' 'Throgs Neck'
'Todt Hill' 'West Farms' 'Silver Lake' 'Morrisania' 'Laurelton'
'Grymes Hill' 'Holliswood' 'Pelham Gardens' 'Belmont' 'Rosedale'
'Edgemere' 'New Brighton' 'Midland Beach' 'Baychester' 'Melrose'
'Bergen Beach' 'Richmondtown' 'Howland Hook' 'Schuylerville'
'Coney Island' 'New Dorp Beach' "Prince's Bay" 'South Beach' 'Bath Beach'
'Jamaica Hills' 'Oakwood' 'Castle Hill' 'Hollis' 'Douglaston' 'Huguenot'
'Olinville' 'Edenwald' 'Grant City' 'Westerleigh'
'Bay Terrace, Staten Island' 'Westchester Square' 'Little Neck'
'Fort Wadsworth' 'Rosebank' 'Unionport' 'Mill Basin' 'Arden Heights'
"Bull's Head" 'New Dorp' 'Rossville' 'Breezy Point' 'Willowbrook'
'Glen Oaks' 'Gerritsen Beach' 'Chelsea, Staten Island']

[40.64749 40.75362 40.80902 ... 40.63724 40.7669 40.84753]

[-73.97237 -73.98377 -73.9419 ... -73.86641 -73.89963 -73.89887]

['United States' nan]

['US' nan]

[False True nan]

['strict' 'moderate' 'flexible' nan]

['Private room' 'Entire home/apt' 'Shared room' 'Hotel room']

[2020. 2007. 2005. 2009. 2013. 2015. 2004. 2008. 2010. 2019. 2018. 2006.
2016. 2017. 2021. 2003. 2011. 2012. 2022. 2014. nan]

['\$966 ' '\$142 ' '\$620 ' ... '\$1,184 ' '\$905 ' '\$309 ']

['\$193 ' '\$28 ' '\$124 ' '\$74 ' '\$41 ' '\$115 ' '\$14 ' '\$212 ' '\$204 '
 '\$58 ' '\$64 ' '\$121 ' '\$143 ' '\$116 ' '\$30 ' nan '\$56 ' '\$95 ' '\$27 '
 '\$210 ' '\$163 ' '\$235 ' '\$106 ' '\$55 ' '\$42 ' '\$86 ' '\$133 ' '\$154 '
 '\$102 ' '\$172 ' '\$109 ' '\$38 ' '\$222 ' '\$219 ' '\$202 ' '\$113 ' '\$137 '
 '\$171 ' '\$12 ' '\$87 ' '\$183 ' '\$92 ' '\$73 ' '\$146 ' '\$180 ' '\$67 ' '\$51 '
 '\$181 ' '\$88 ' '\$205 ' '\$132 ' '\$59 ' '\$156 ' '\$131 ' '\$61 ' '\$165 ']

```
'$129 ' '$207 ' '$17 ' '$63 ' '$147 ' '$105 ' '$48 ' '$114 ' '$80 '
'$23 ' '$11 ' '$149 ' '$213 ' '$211 ' '$182 ' '$117 ' '$39 ' '$25 '
'$216 ' '$53 ' '$10 ' '$118 ' '$72 ' '$76 ' '$69 ' '$32 ' '$148 ' '$101 '
'$177 ' '$49 ' '$100 ' '$168 ' '$13 ' '$152 ' '$126 ' '$135 ' '$46 '
'$231 ' '$91 ' '$199 ' '$158 ' '$221 ' '$189 ' '$90 ' '$186 ' '$66 '
'$225 ' '$220 ' '$19 ' '$238 ' '$119 ' '$98 ' '$34 ' '$227 ' '$162 '
'$203 ' '$93 ' '$153 ' '$71 ' '$77 ' '$190 ' '$75 ' '$185 ' '$97 ' '$35 '
'$37 ' '$82 ' '$120 ' '$21 ' '$145 ' '$65 ' '$144 ' '$108 ' '$194 '
'$16 ' '$217 ' '$188 ' '$81 ' '$139 ' '$22 ' '$187 ' '$150 ' '$50 '
'$176 ' '$151 ' '$26 ' '$89 ' '$218 ' '$104 ' '$44 ' '$79 ' '$240 '
'$175 ' '$192 ' '$215 ' '$191 ' '$94 ' '$206 ' '$157 ' '$159 ' '$233 '
'$15 ' '$229 ' '$179 ' '$85 ' '$136 ' '$170 ' '$201 ' '$141 ' '$123 '
'$36 ' '$130 ' '$198 ' '$54 ' '$122 ' '$184 ' '$208 ' '$140 ' '$68 '
'$196 ' '$70 ' '$228 ' '$209 ' '$230 ' '$134 ' '$20 ' '$110 ' '$234 '
'$107 ' '$138 ' '$45 ' '$24 ' '$31 ' '$29 ' '$232 ' '$47 ' '$239 '
'$161 ' '$200 ' '$33 ' '$103 ' '$83 ' '$128 ' '$166 ' '$125 ' '$43 '
'$18 ' '$167 ' '$174 ' '$236 ' '$111 ' '$142 ' '$223 ' '$169 ' '$178 '
'$195 ' '$40 ' '$52 ' '$96 ' '$224 ' '$197 ' '$164 ' '$127 ' '$62 '
'$99 ' '$84 ' '$237 ' '$155 ' '$57 ' '$226 ' '$160 ' '$173 ' '$60 '
'$214 ' '$78 ' '$112 ' ']
```

```
[ 1.000e+01 3.000e+01 3.000e+00 4.500e+01 2.000e+00 1.000e+00
5.000e+00 4.000e+00 9.000e+01 7.000e+00 1.400e+01 6.000e+01
nan 1.800e+02 9.000e+00 3.100e+01 6.000e+00 1.500e+01
2.900e+01 8.000e+00 4.700e+01 8.100e+01 1.440e+02 3.710e+02
1.490e+02 2.730e+02 1.310e+02 2.750e+02 3.230e+02 5.900e+01
9.900e+01 2.680e+02 -1.000e+01 1.890e+02 3.400e+02 1.280e+02
3.500e+02 1.660e+02 1.100e+02 5.700e+01 1.420e+02 3.660e+02
2.800e+01 2.000e+02 5.000e+01 3.990e+02 1.700e+01 1.100e+01
4.520e+02 2.500e+01 1.300e+01 3.455e+03 -5.000e+00 2.100e+01
2.700e+01 1.800e+01 3.650e+02 -1.000e+00 2.000e+01 -1.200e+01
4.000e+01 4.400e+01 3.980e+02 -2.000e+00 6.500e+01 -3.000e+00
5.500e+01 1.200e+02 1.220e+02 1.900e+01 2.400e+02 8.800e+01
1.150e+02 1.500e+02 3.700e+02 1.600e+01 8.000e+01 1.810e+02
2.600e+01 2.650e+02 3.000e+02 1.850e+02 3.600e+02 5.600e+01
1.200e+01 7.000e+01 3.900e+01 2.400e+01 3.200e+01 1.000e+03
2.700e+02 2.200e+01 7.500e+01 2.500e+02 6.200e+01 2.300e+01
1.250e+03 3.640e+02 7.400e+01 1.980e+02 1.000e+02 5.000e+02
4.300e+01 3.500e+01 9.100e+01 4.800e+02 5.300e+01 1.600e+02
9.990e+02 1.860e+02 6.800e+01 9.300e+01 8.700e+01 1.830e+02
2.990e+02 1.750e+02 9.800e+01 1.330e+02 3.540e+02 4.200e+01
3.300e+01 3.700e+01 2.250e+02 4.000e+02 1.050e+02 1.840e+02
1.530e+02 1.340e+02 2.220e+02 5.800e+01 2.100e+02 1.820e+02
9.540e+02 5.645e+03 2.645e+03 -1.223e+03 -3.650e+02 -2.000e+02
1.140e+02 8.500e+01 3.600e+01 4.800e+01 1.550e+02 3.070e+02
1.290e+02 1.450e+02 6.400e+01 3.330e+02 1.190e+02 -1.250e+02
1.250e+02 3.400e+01 4.580e+02 8.250e+02]
```

```
[9.000e+00 4.500e+01 0.000e+00 2.700e+02 7.400e+01 4.900e+01 4.300e+02
1.180e+02 1.600e+02 5.300e+01 1.880e+02 1.670e+02 1.130e+02 2.700e+01
1.480e+02 1.980e+02 2.600e+02 1.300e+02 3.900e+01 7.100e+01 8.800e+01
1.900e+01 5.800e+01 1.080e+02 2.900e+01 2.420e+02 1.970e+02 2.730e+02]
```

1.680e+02	2.310e+02	2.140e+02	2.450e+02	1.500e+01	2.500e+01	8.100e+01
9.700e+01	1.100e+01	2.480e+02	6.100e+01	1.350e+02	1.120e+02	7.300e+01
8.200e+01	3.280e+02	1.050e+02	2.890e+02	1.380e+02	2.100e+01	4.200e+01
5.000e+00	6.600e+01	1.430e+02	1.910e+02	4.000e+00	3.380e+02	1.060e+02
1.900e+02	2.300e+01	1.420e+02	5.400e+01	7.000e+01	1.600e+01	9.400e+01
1.940e+02	2.000e+00	1.740e+02	2.400e+01	1.660e+02	nan	1.000e+00
3.000e+01	1.390e+02	2.330e+02	6.800e+01	4.600e+01	3.350e+02	1.620e+02
1.700e+02	3.340e+02	1.200e+01	4.670e+02	7.000e+00	3.800e+01	3.240e+02
1.150e+02	3.540e+02	1.950e+02	1.300e+01	3.600e+01	6.300e+01	1.550e+02
1.930e+02	3.200e+01	5.000e+01	2.600e+01	4.260e+02	2.270e+02	8.400e+01
1.000e+01	1.240e+02	2.400e+02	2.000e+02	7.900e+01	3.400e+01	1.340e+02
1.260e+02	2.340e+02	2.020e+02	2.800e+01	3.090e+02	1.400e+01	8.000e+01
2.940e+02	1.500e+02	4.700e+01	2.190e+02	1.140e+02	2.130e+02	8.600e+01
1.800e+01	2.060e+02	1.220e+02	3.300e+01	5.200e+01	5.100e+01	1.990e+02
3.000e+00	4.100e+01	1.090e+02	1.510e+02	2.850e+02	3.750e+02	6.000e+00
3.580e+02	2.260e+02	1.040e+02	2.040e+02	2.530e+02	1.290e+02	3.700e+01
6.900e+01	1.920e+02	1.700e+01	2.220e+02	2.050e+02	4.580e+02	1.830e+02
1.890e+02	1.270e+02	3.500e+01	1.710e+02	9.200e+01	2.380e+02	5.600e+01
1.110e+02	1.470e+02	1.770e+02	1.850e+02	6.200e+01	1.810e+02	3.330e+02
4.410e+02	2.790e+02	2.030e+02	2.100e+02	6.400e+01	1.320e+02	2.000e+01
6.700e+01	1.870e+02	2.200e+01	9.300e+01	4.800e+01	1.200e+02	4.300e+01
5.900e+01	3.100e+01	2.200e+02	2.860e+02	3.980e+02	7.600e+01	1.820e+02
8.000e+00	2.280e+02	3.880e+02	2.230e+02	2.180e+02	7.500e+01	3.700e+02
1.310e+02	1.360e+02	9.800e+01	3.800e+02	1.630e+02	2.470e+02	1.160e+02
3.200e+02	2.250e+02	9.500e+01	4.030e+02	1.750e+02	2.630e+02	2.320e+02
7.200e+01	4.140e+02	1.190e+02	1.170e+02	3.850e+02	1.030e+02	2.800e+02
5.700e+01	4.800e+02	1.010e+02	8.700e+01	2.410e+02	9.000e+01	9.900e+01
2.710e+02	8.900e+01	1.790e+02	4.010e+02	8.300e+01	2.110e+02	2.950e+02
2.070e+02	1.720e+02	3.300e+02	1.230e+02	5.500e+01	2.580e+02	1.280e+02
1.520e+02	1.540e+02	1.640e+02	2.780e+02	2.350e+02	3.050e+02	1.020e+02
6.500e+01	4.000e+01	1.000e+02	1.250e+02	2.290e+02	6.000e+01	1.070e+02
1.760e+02	1.780e+02	2.160e+02	3.140e+02	1.530e+02	1.100e+02	2.440e+02
2.080e+02	1.210e+02	1.570e+02	2.090e+02	4.400e+01	1.580e+02	2.360e+02
2.150e+02	7.800e+01	2.970e+02	7.700e+01	2.650e+02	1.560e+02	2.720e+02
2.680e+02	1.330e+02	1.650e+02	2.690e+02	2.430e+02	1.370e+02	1.450e+02
3.390e+02	4.540e+02	2.300e+02	1.440e+02	4.390e+02	2.610e+02	3.190e+02
1.840e+02	4.510e+02	1.690e+02	2.660e+02	2.510e+02	1.410e+02	3.040e+02
4.740e+02	8.500e+01	2.210e+02	2.010e+02	3.210e+02	2.500e+02	1.490e+02
2.540e+02	1.610e+02	3.230e+02	3.470e+02	2.960e+02	9.600e+01	1.730e+02
2.390e+02	4.040e+02	3.260e+02	3.630e+02	1.460e+02	3.270e+02	2.370e+02
3.530e+02	9.100e+01	3.020e+02	2.760e+02	1.860e+02	3.220e+02	2.670e+02
5.400e+02	2.240e+02	2.910e+02	3.150e+02	3.130e+02	5.940e+02	5.970e+02
6.070e+02	2.870e+02	1.590e+02	3.950e+02	4.470e+02	2.810e+02	3.560e+02
3.600e+02	3.490e+02	2.460e+02	3.080e+02	2.920e+02	1.400e+02	2.840e+02
3.740e+02	1.960e+02	2.120e+02	2.740e+02	3.400e+02	3.860e+02	3.360e+02
2.490e+02	2.750e+02	2.590e+02	2.520e+02	2.980e+02	2.880e+02	2.550e+02
3.650e+02	2.570e+02	2.170e+02	4.880e+02	3.320e+02	1.800e+02	2.770e+02
3.890e+02	3.720e+02	2.990e+02	3.940e+02	3.570e+02	3.500e+02	3.520e+02
3.510e+02	2.620e+02	3.180e+02	3.070e+02	3.250e+02	2.820e+02	3.960e+02
4.660e+02	3.970e+02	2.900e+02	4.480e+02	4.240e+02	4.080e+02	5.430e+02
2.640e+02	6.290e+02	5.760e+02	2.560e+02	4.590e+02	4.120e+02	3.780e+02
4.110e+02	4.340e+02	5.100e+02	3.060e+02	3.920e+02	4.170e+02	3.680e+02
4.360e+02	3.100e+02	3.410e+02	3.480e+02	7.760e+02	8.490e+02	6.180e+02
4.850e+02	5.500e+02	3.640e+02	4.150e+02	4.430e+02	3.770e+02	3.790e+02

```
3.290e+02 4.750e+02 9.660e+02 7.970e+02 5.920e+02 5.670e+02 6.480e+02
3.670e+02 4.830e+02 3.460e+02 4.060e+02 4.690e+02 3.420e+02 3.110e+02
3.450e+02 6.380e+02 3.620e+02 5.600e+02 6.980e+02 3.990e+02 4.320e+02
3.440e+02 4.490e+02 5.420e+02 3.710e+02 2.930e+02 3.610e+02 4.700e+02
4.730e+02 3.870e+02 4.960e+02 4.380e+02 4.130e+02 1.024e+03 3.010e+02
3.030e+02 4.550e+02 8.840e+02 4.180e+02 5.200e+02 5.660e+02 2.830e+02
4.400e+02 3.120e+02 4.200e+02 4.330e+02 3.370e+02 4.500e+02 5.830e+02
3.160e+02 4.910e+02 3.660e+02 3.430e+02 6.790e+02 5.630e+02 3.170e+02
3.730e+02 5.560e+02 4.290e+02 3.310e+02 4.680e+02 7.380e+02 4.770e+02
4.840e+02 5.010e+02 4.210e+02 1.010e+03 4.460e+02 4.900e+02 4.000e+02
3.000e+02]
```

```
-----
-----
['10/19/2021' '5/21/2022' nan ... '2/26/2020' '2/25/2020' '2/24/2020']
-----
```

```
-----
[ 0.21 0.38 nan ... 7.73 24.49 33.08]
-----
```

```
-----
[ 4. 5. 3. nan 2. 1.]
-----
```

```
-----
[ 6. 2. 1. 4. 3. nan 5. 7. 13. 28. 11. 8. 9. 52.
 18. 15. 19. 10. 39. 26. 29. 12. 21. 96. 14. 34. 43. 121.
 37. 49. 31. 91. 16. 87. 33. 23. 50. 20. 25. 232. 17. 47.
103. 65. 30. 27. 327. 32. 45. 53. 208. 171. 44. 48. 108. 51.
161. 152. 83. 24. 22. 126. 54. 66. 59. 62. 102. 186. 125. 55.
 63. 67. 58. 56. 80. 46. 218. 86. 332.]
-----
```

```
-----
[ 2.860e+02 2.280e+02 3.520e+02 3.220e+02 2.890e+02 3.740e+02
 2.240e+02 2.190e+02 1.800e+02 3.750e+02 1.000e+00 1.630e+02
 2.580e+02 4.700e+01 6.800e+01 1.000e+02 1.970e+02 9.600e+01
 3.250e+02 3.450e+02 3.470e+02 1.930e+02 5.400e+01 9.000e+00
 3.440e+02 3.720e+02 1.030e+02 1.720e+02 5.600e+01 1.050e+02
 1.690e+02 3.830e+02 4.110e+02 7.600e+01 4.160e+02 2.940e+02
 2.140e+02 3.560e+02 1.100e+02 3.540e+02 3.300e+01 4.050e+02
 8.600e+01 1.520e+02 3.530e+02 2.600e+01 3.200e+01 8.500e+01
      nan 8.200e+01 9.200e+01 5.500e+01 1.580e+02 6.400e+01
 1.840e+02 2.010e+02 3.280e+02 3.930e+02 6.100e+01 2.670e+02
 1.890e+02 8.300e+01 4.000e+02 3.400e+01 2.030e+02 3.500e+01
 2.800e+02 4.170e+02 1.620e+02 2.370e+02 2.420e+02 2.270e+02
 1.730e+02 1.460e+02 2.430e+02 4.250e+02 1.130e+02 2.600e+02
 3.950e+02 2.660e+02 5.800e+01 4.100e+02 3.880e+02 1.300e+01
 6.200e+01 3.660e+02 3.380e+02 3.340e+02 -1.000e+01 3.780e+02
 -2.000e+00 3.510e+02 3.890e+02 1.600e+01 3.370e+02 1.660e+02
 1.800e+01 1.300e+02 4.300e+01 0.000e+00 3.730e+02 4.260e+02
 3.960e+02 4.190e+02 3.770e+02 1.900e+02 3.350e+02 5.200e+01
 7.000e+01 3.110e+02 1.410e+02 -1.000e+00 2.630e+02 2.000e+02
 1.440e+02 3.200e+02 1.900e+01 2.930e+02 2.400e+02 3.000e+02
 4.210e+02 3.320e+02 2.710e+02 1.560e+02 4.200e+02 3.990e+02
 1.200e+01 1.500e+01 1.380e+02 3.270e+02 1.240e+02 1.370e+02
 1.920e+02 8.100e+01 3.610e+02 3.410e+02 2.130e+02 3.360e+02
 2.300e+01 7.800e+01 1.880e+02 6.600e+01 8.000e+01 3.820e+02]
```


3.290e+02	2.560e+02	-6.000e+00	3.860e+02	1.090e+02	3.190e+02
5.000e+01	1.180e+02	3.500e+02	1.500e+02	3.130e+02	2.480e+02
2.300e+02	1.450e+02	4.070e+02	4.220e+02	4.600e+01	1.010e+02
2.100e+02	1.040e+02	3.230e+02	2.060e+02	2.950e+02	3.970e+02
2.760e+02	3.980e+02	1.350e+02	4.120e+02	8.700e+01	-9.000e+00
3.150e+02	1.320e+02	4.200e+01	7.300e+01	1.770e+02	2.450e+02
2.720e+02	2.470e+02	3.940e+02	1.790e+02	3.100e+01	9.500e+01
1.540e+02	2.040e+02	3.710e+02	3.120e+02	3.800e+02	2.700e+02
4.500e+01	1.760e+02	3.790e+02	2.640e+02	4.020e+02	5.300e+01
2.520e+02	5.000e+00	3.620e+02	2.380e+02	4.140e+02	2.090e+02
4.060e+02	1.100e+01	2.350e+02	3.180e+02	1.400e+02	1.070e+02
1.310e+02	1.170e+02	3.160e+02	1.190e+02	4.240e+02	1.550e+02
2.870e+02	8.400e+01	2.910e+02	2.070e+02	3.490e+02	1.980e+02
1.430e+02	2.850e+02	3.580e+02	3.310e+02	2.830e+02	1.140e+02
7.700e+01	3.760e+02	2.110e+02	1.120e+02	2.960e+02	1.810e+02
1.060e+02	9.700e+01	3.680e+02	2.310e+02	2.970e+02	-4.000e+00
1.990e+02	1.000e+01	9.400e+01	7.500e+01	1.820e+02	1.640e+02
3.810e+02	2.530e+02	1.910e+02	7.100e+01	9.800e+01	4.080e+02
1.590e+02	3.260e+02	1.470e+02	1.860e+02	1.870e+02	2.680e+02
8.800e+01	4.040e+02	3.690e+02	2.920e+02	1.710e+02	3.010e+02
1.290e+02	3.000e+01	3.650e+02	2.690e+02	6.700e+01	3.400e+02
6.300e+01	2.360e+02	-8.000e+00	2.510e+02	3.240e+02	9.300e+01
9.100e+01	2.590e+02	4.000e+00	3.840e+02	2.650e+02	2.210e+02
1.220e+02	7.000e+00	2.900e+01	1.160e+02	3.920e+02	1.750e+02
3.030e+02	2.330e+02	6.900e+01	3.670e+02	2.390e+02	3.600e+01
1.490e+02	3.900e+01	3.870e+02	3.210e+02	3.020e+02	3.000e+00
1.830e+02	8.000e+00	2.440e+02	6.000e+01	1.530e+02	3.550e+02
2.500e+02	3.640e+02	-5.000e+00	2.160e+02	4.180e+02	1.670e+02
1.780e+02	1.400e+01	3.700e+02	2.540e+02	2.200e+01	3.330e+02
2.400e+01	2.460e+02	2.610e+02	1.420e+02	2.120e+02	3.390e+02
2.800e+01	3.090e+02	1.740e+02	2.320e+02	2.000e+01	1.330e+02
1.700e+01	2.700e+01	4.100e+01	2.260e+02	1.080e+02	1.680e+02
2.550e+02	2.780e+02	1.250e+02	5.700e+01	1.390e+02	6.500e+01
2.250e+02	2.050e+02	2.810e+02	2.500e+01	1.950e+02	3.700e+01
2.790e+02	1.150e+02	2.740e+02	2.150e+02	1.340e+02	6.000e+00
1.940e+02	1.110e+02	2.170e+02	1.960e+02	2.200e+02	2.180e+02
4.230e+02	3.300e+02	7.900e+01	1.850e+02	9.000e+01	2.000e+00
2.750e+02	3.480e+02	3.140e+02	3.100e+02	1.600e+02	3.570e+02
4.400e+01	3.590e+02	2.990e+02	4.130e+02	2.820e+02	2.900e+02
1.260e+02	3.800e+01	2.490e+02	2.020e+02	7.200e+01	2.770e+02
2.100e+01	1.650e+02	7.400e+01	2.220e+02	4.090e+02	1.700e+02
2.410e+02	8.900e+01	1.280e+02	1.210e+02	2.230e+02	2.730e+02
5.900e+01	1.230e+02	2.290e+02	-3.000e+00	3.850e+02	3.050e+02
3.070e+02	9.900e+01	2.340e+02	2.570e+02	3.060e+02	5.100e+01
2.080e+02	2.840e+02	1.360e+02	3.910e+02	4.900e+01	1.510e+02
1.270e+02	3.430e+02	3.080e+02	1.480e+02	4.010e+02	3.170e+02
3.040e+02	1.610e+02	3.600e+02	2.980e+02	4.000e+01	3.630e+02
4.030e+02	1.570e+02	3.420e+02	2.620e+02	3.900e+02	3.460e+02
1.020e+02	4.800e+01	1.200e+02	2.880e+02	-7.000e+00	4.150e+02
3.677e+03]					

 ["Clean up and treat the home the way you'd like your home to be treated. No s
 moking."]

'Pet friendly but please confirm with me if the pet you are planning on bringing with you is OK. I have a cute and quiet mixed chihuahua. I could accept more guests (for an extra fee) but this also needs to be confirmed beforehand. Also friends traveling together could sleep in separate beds for an extra fee (the second bed is either a sofa bed or inflatable bed). Smoking is only allowed on the porch.'

"I encourage you to use my kitchen, cooking and laundry facilities. There is no additional charge to use the washer/dryer in the basement. No smoking, inside or outside. Come home as late as you want. If you come home stumbling drunk, it's OK the first time. If you do it again, and you wake up me or the neighbors downstairs, we will be annoyed. (Just so you know . . .)"

'''

'15 nights minimum stay No pets at any time No smoking Please be mindful of the furniture and personal belongings, it was all furnished with designer pieces and hard to replace things.'

'House rules: Guests agree to the following terms and conditions 1.Guest(s) agree to NO PARTIES anywhere on the property. Legal action will be taken for parties that cause damages to the unit, property, reputation, or relationship with building staff or other residents. 2.Smoking of any kind is not permitted in the unit or on the premises. 3.Guests agree to abide by maximum occupancy of unit, and observe quiet hours between 9pm and 9am 4.Guests are responsible for action of ALL GUESTS occupying their unit at all times. 5.Guests agree to return all keys, remotes, and parking passes. Replacement fees may apply for unreturned items 6.Guests understand that onsite building staff (maintenance and concierge) are not affiliated with Jordan or AIRBNB. Please contact me directly for any needed items. 7.Guests understand that no pets are to be in the unit. 8.Guests agree to pay replacement cost for any unreturned items, and \$125 per hour fee for unauthorized early checkin or late checkout. 22'

'16 nights minimum stay, no smoking or pets.']

[nan '41662/AL']

```
In [10]: Airbnb_Clean["neighbourhood_group"].value_counts()
Airbnb_Clean["neighbourhood_group"].replace({"brookln":"Brooklyn",
"manhatan":"Manhattan"},inplace=True)
```

```
In [11]: # check duplicate
DUBplicate = Airbnb_Clean[Airbnb_Clean.duplicated()].shape[0]
print(f"The Total Numbers of Duplicate Values are {DUBplicate}")
```

The Total Numbers of Duplicate Values are 541

```
In [12]: # Remove Duplicate
Airbnb_Clean = Airbnb_Clean.drop_duplicates(keep="first")
```

```
In [13]: DUBplicate = Airbnb_Clean[Airbnb_Clean.duplicated()].shape[0]
print(f"The Total Numbers of Duplicate Values are {DUBplicate}")
```

The Total Numbers of Duplicate Values are 0

```
In [14]: # Check missing Value
```

```
Null_Summary = pd.DataFrame({"Number_of_Null":Airbnb_Clean.isnull().sum(),
"Percentage": round(((Airbnb_Clean.isnull().sum())/len(Airbnb_Clean))*100,2),
"DataType": Airbnb_Clean.dtypes})
Null_Summary
```

Out[14]:

	Number_of_Null	Percentage	DataType
id	0	0.00	int64
NAME	250	0.24	object
host_id	0	0.00	int64
host_identity_verified	289	0.28	object
host_name	404	0.40	object
neighbourhood_group	29	0.03	object
neighbourhood	16	0.02	object
lat	8	0.01	float64
long	8	0.01	float64
country	532	0.52	object
country_code	131	0.13	object
instant_bookable	105	0.10	object
cancellation_policy	76	0.07	object
room_type	0	0.00	object
Construction_year	214	0.21	float64
price	247	0.24	object
service_fee	273	0.27	object
minimum_nights	400	0.39	float64
number_of_reviews	183	0.18	float64
last_review	15832	15.51	object
reviews_per_month	15818	15.50	float64
review_rate_number	319	0.31	float64
calculated_host_listings_count	319	0.31	float64
availability_365	448	0.44	float64
house_rules	51842	50.80	object
license	102056	100.00	object

In [15]:

```
Overwise_Null = ((Airbnb_Clean.isnull().sum().sum())/ (Airbnb_Clean.shape[0]*Ai
print(f"Overall null Percentage {Overwise_Null:.2f}")
```

Overall null Percentage 7.15

```
In [16]: # Remove $ from Price & Service Fees
columns = ["price", "service_fee"]
for col in columns:
    Airbnb_Clean[col] = Airbnb_Clean[col].str.replace(r"[$,]", "", regex=True).a

In [17]: # fill Price & Service Fee
for col in columns:
    Airbnb_Clean[col].fillna(Airbnb_Clean[col].median(), inplace=True)

In [18]: # fill object columns
Columns = ["NAME", "host_identity_verified", "host_name", "neighbourhood_group",
"neighbourhood", "country", "country_code", "instant_bookable",
"cancellation_policy"]
for col in Columns:
    Airbnb_Clean[col].fillna(Airbnb_Clean[col].mode()[0], inplace=True)

In [19]: # Handling Date columns
Airbnb_Clean.rename(columns={"last_review": "Date"}, inplace=True)

In [20]: Airbnb_Clean["Date"] = pd.to_datetime(Airbnb_Clean["Date"], errors="coerce")

In [21]: Airbnb_Clean.dropna(subset=["Date"], inplace=True)

In [22]: Airbnb_Clean = Airbnb_Clean[Airbnb_Clean["Date"] <= pd.Timestamp.today()]

In [23]: Airbnb_Clean["house_rules"].fillna("No Rules Applied", inplace=True)

In [24]: # Remove Outlier
# minimum_nights
Airbnb_Clean = Airbnb_Clean[
    (Airbnb_Clean["minimum_nights"] > 1) &
    (Airbnb_Clean["minimum_nights"] <= 365)
]

# availability_365
Airbnb_Clean = Airbnb_Clean[
    Airbnb_Clean["availability_365"].between(0, 365)
]

In [25]: Airbnb_Clean.drop(columns=["license"], axis=1, inplace=True)
Airbnb_Clean.dropna(subset=["lat", "long"], inplace=True)

In [26]: num_cols = [
    "price",
    "service_fee",
    "reviews_per_month",
    "review_rate_number",
    "calculated_host_listings_count",
```

```

    "number_of_reviews",
    "Construction_year"
]

for col in num_cols:
    Airbnb_Clean[col].fillna(Airbnb_Clean[col].median(), inplace=True)

```

In [27]: `Airbnb_Clean["Construction_year"] = Airbnb_Clean["Construction_year"].astype(i`

Load Clean Data in MySQL

```

In [28]: import urllib.parse
from sqlalchemy import create_engine
import time
username = "root"
raw_password = "ANku7970@!#"
password = urllib.parse.quote_plus(raw_password)

host = "localhost"
port = 3306
Database = "Airbnb_Database"
engine = create_engine(f"mysql+pymysql://{username}:{password}@{host}:{port}/{

try:
    print("Attempting to connect to database...")

    for x in reversed(range(1,4)):
        print(f"Connecting in {x}...")
        time.sleep(1)

    Airbnb_Clean.to_sql(
        "Airbnb_Clean",
        engine,
        if_exists="replace",
        index=False
    )

    print("Connected ✓ Data Loaded Successfully")

except Exception as e:
    print("Not Connected ✗")
    print("Reason:", e)

```

```

Attempting to connect to database...
Connecting in 3...
Connecting in 2...
Connecting in 1...
Connected ✓ Data Loaded Successfully

```

Exploratory Data Analysis

```
In [29]: Airbnb_Clean.describe().T
```

```
Out[29]:
```

		count	mean	min	
	id	62712.0	30125513.933761	1001254.0	1606
	host_id	62712.0	49182756900.83828	123600518.0	2455585
	lat	62712.0	40.726856	40.50456	40
	long	62712.0	-73.951428	-74.24984	-73
	Construction_year	62712.0	2012.469352	2003.0	
	price	62712.0	626.122241	50.0	
	service_fee	62712.0	125.199675	10.0	
	minimum_nights	62712.0	9.289769	2.0	
	number_of_reviews	62712.0	28.785942	1.0	
	Date	62712	2019-06-29 23:14:00.413318144	2012-07-11 00:00:00	2018 00
	reviews_per_month	62712.0	1.13748	0.01	
	review_rate_number	62712.0	3.28296	1.0	
	calculated_host_listings_count	62712.0	8.063433	1.0	
	availability_365	62712.0	132.10081	0.0	

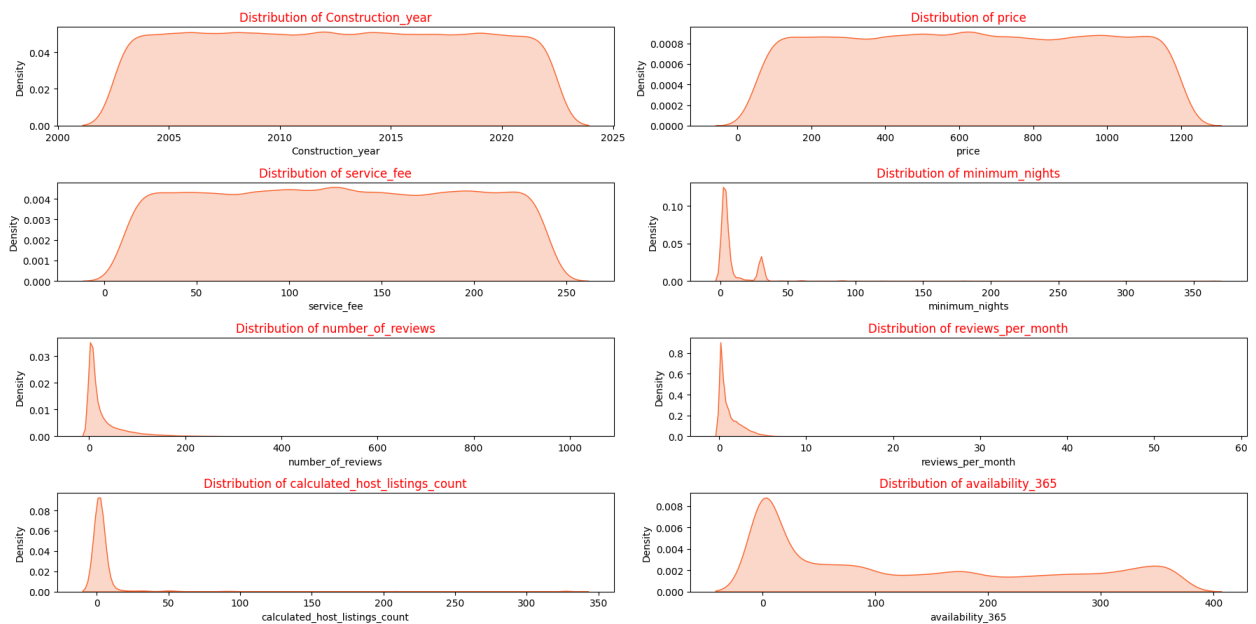
```
In [30]: Airbnb_Clean.describe(include="object").T
```

```
Out[30]:
```

	count	unique	top	freq
NAME	62712	37919	Home away from home	143
host_identity_verified	62712	2	unconfirmed	31530
host_name	62712	9526	Michael	768
neighbourhood_group	62712	5	Brooklyn	27085
neighbourhood	62712	221	Bedford-Stuyvesant	5322
country	62712	1	United States	62712
country_code	62712	1	US	62712
cancellation_policy	62712	3	moderate	21016
room_type	62712	4	Entire home/apt	37320
house_rules	62712	1923	No Rules Applied	33772

1. Distribution of Numerical Column

```
In [31]: Numerical_col = ["Construction_year", "price", "service_fee",  
    "minimum_nights", "number_of_reviews", "reviews_per_month",  
    "calculated_host_listings_count", "availability_365"]  
plt.figure(figsize=(18,9))  
for x, col in enumerate(Numerical_col):  
    plt.subplot(4,2,x+1)  
    sns.kdeplot(x = Airbnb_Clean[col], fill=True, color="#FC642D")  
    plt.title(f"Distribution of {col}", fontsize = 12,  
        color = "red")  
plt.tight_layout()  
plt.show()
```



Insights

1. Construction Year

- Most properties were constructed between 2000 and 2022, indicating that Airbnb listings largely consist of relatively modern buildings rather than very old properties.

2. Price

- Listing prices are widely distributed, ranging from budget to premium segments, showing that Airbnb caters to a diverse customer base with varying affordability levels.

3. Service Fee

- Service fees follow a similar distribution to prices, suggesting a strong relationship between listing price and the associated service fee.

4.Minimum Nights

- The majority of listings require a low minimum stay (around 2–5 nights), highlighting Airbnb’s focus on short-term accommodations, with fewer listings targeting long-term stays.

5.Number of Reviews

- The number of reviews is highly right-skewed, indicating that while most listings receive limited engagement, a small number of listings accumulate a very high volume of reviews.

6.Reviews per Month

- Most listings receive fewer than two reviews per month, showing that demand is uneven and concentrated among a limited set of highly active listings.

7.Calculated Host Listings Count

- Most hosts manage only one or two listings, while a small proportion of hosts operate a large number of properties, reflecting the presence of both individual and professional hosts on the platform.

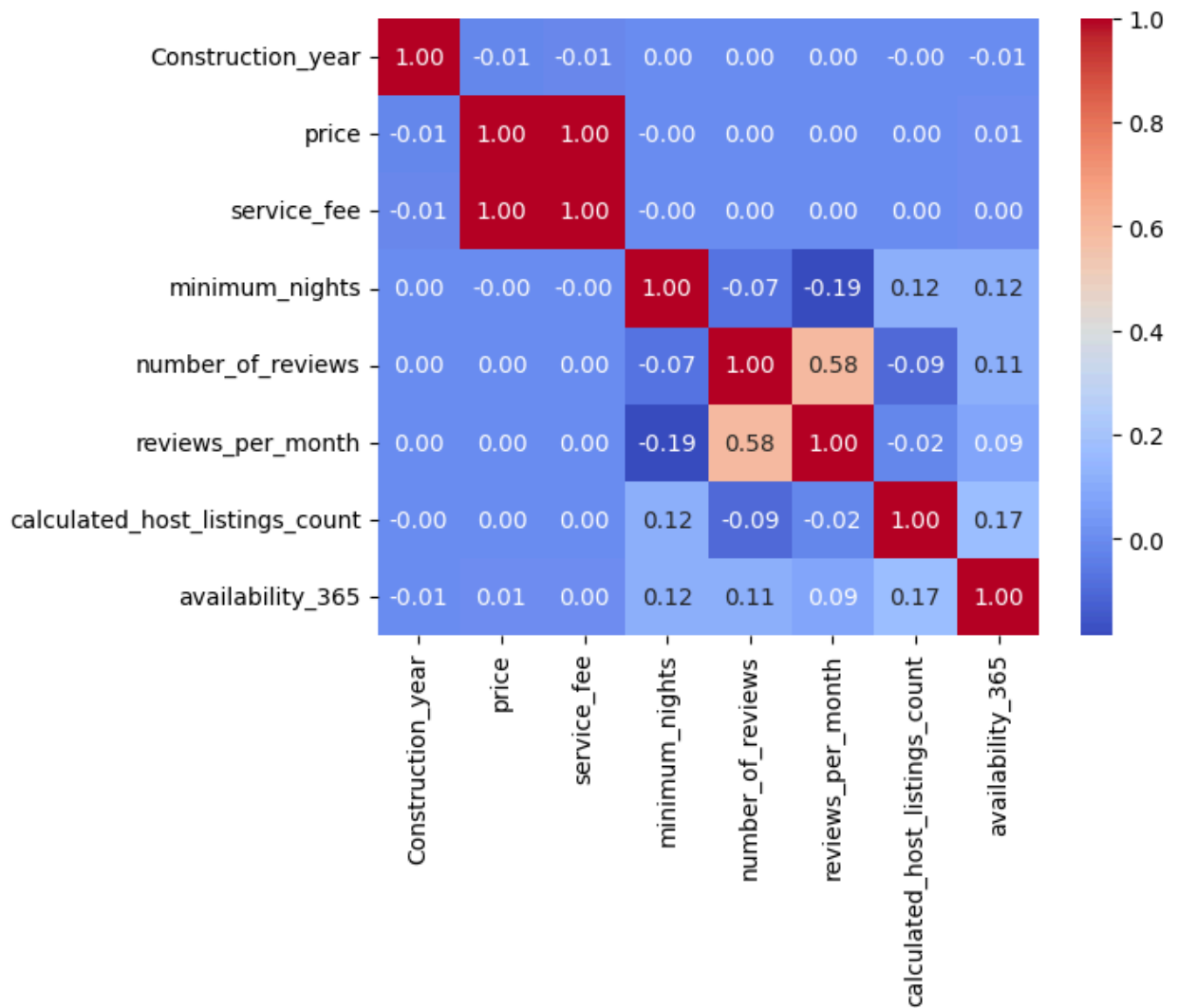
8.Availability (365 Days)

- Availability varies widely across listings, suggesting different hosting strategies, ranging from full-time rentals to part-time or seasonal availability.

2.Heatmap correlation of Numerical

```
In [32]: sns.heatmap(Airbnb_Clean[Numerical_col].corr(),  
                    annot=True,fmt=".2f",cmap="coolwarm")
```

```
Out[32]: <Axes: >
```

Insights

1.Price and Service Fee

- Price and service fee show a perfect positive correlation, indicating that service fees increase directly with listing prices.

2.Number of Reviews and Reviews per Month

- There is a moderate positive correlation (~ 0.58) between number of reviews and reviews per month, suggesting that listings with higher overall engagement also receive reviews more frequently.

3.Minimum Nights and Reviews per Month

- Minimum nights have a weak negative correlation (~ -0.19) with reviews per month, implying that listings requiring longer stays tend to receive

reviews less frequently.

4.Availability and Host Listings Count

- Availability shows a weak positive correlation (~ 0.17) with the number of listings managed by a host, indicating that hosts with multiple properties tend to keep listings available for more days.

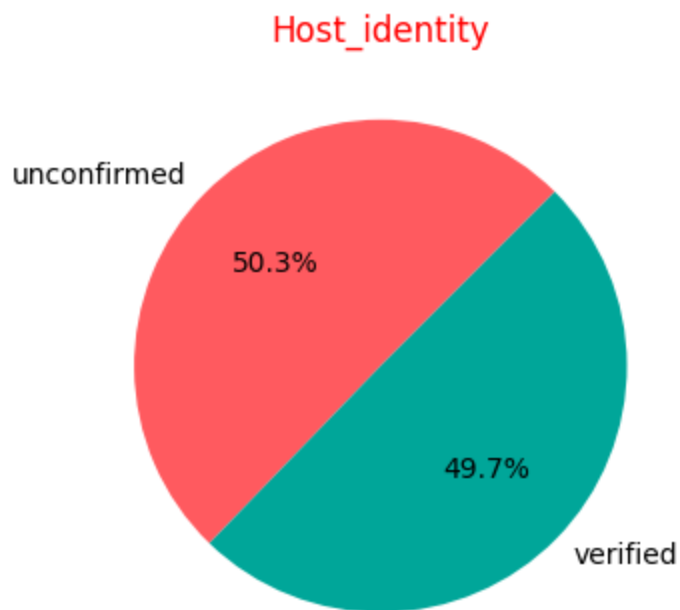
5.Minimum Nights and Availability A weak positive correlation (~ 0.12) suggests that listings with higher minimum stay requirements are often available for more days in a year.

6.Construction Year and Other Variables Construction year has negligible correlation with price, reviews, and availability, indicating that property age does not strongly influence pricing or demand.

7.Price and Demand Metrics Price shows almost no correlation with number of reviews or reviews per month, suggesting that higher prices do not necessarily reduce or increase customer engagement.

3.Host Identity Analysis

```
In [33]: Host_identity = pd.read_sql_query("Select host_identity_verified, count(host_i
as count from Airbnb_Clean group by host_identity_verified", engine)
plt.figure(figsize=(6,4))
plt.pie(Host_identity["count"],labels=Host_identity["host_identity_verified"]
,autopct="%1.1f%%",colors=["#FF5A5F","#00A699"],startangle=45)
plt.title("Host_identity", color = "Red", fontsize = 12)
plt.show()
```

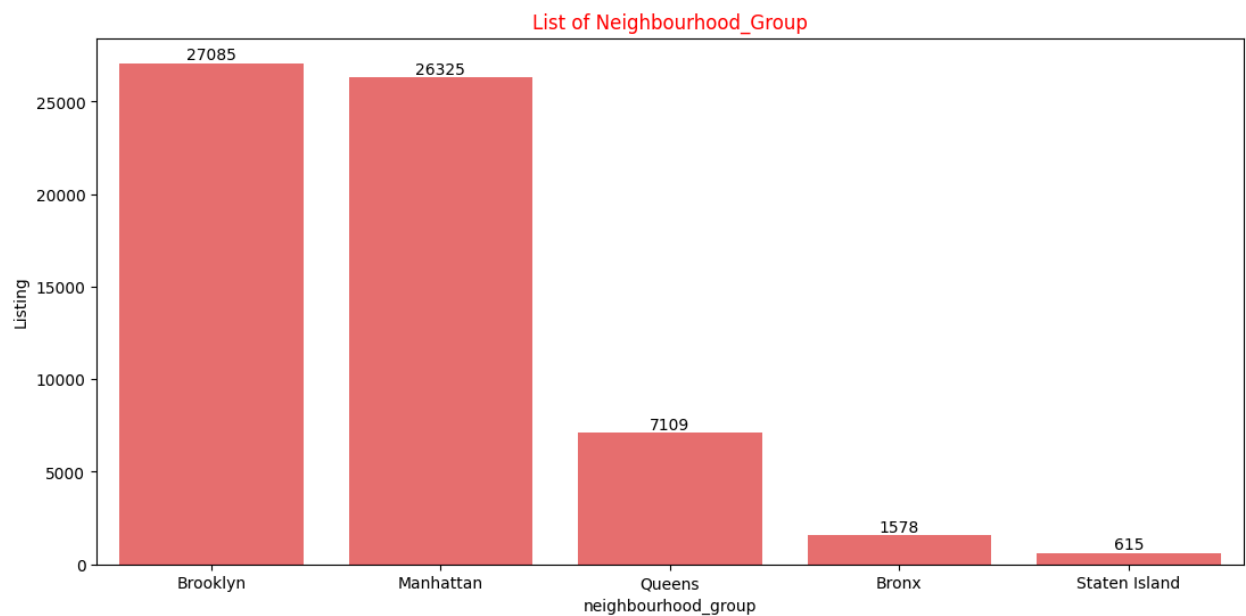


Insights

About 50.3% of hosts are not identity verified, while 49.7% of hosts are verified, showing an almost equal distribution between verified and unverified hosts.

4. Neighbourhood group Analysis

```
In [34]: Neighbourhood_group = pd.read_sql_query("Select neighbourhood_group ,count(Lei
         \"from AIrbnb_clean group by neighbourhood_group order by Listing desc\", engine
         plt.figure(figsize=(13,6))
         ax = sns.barplot(x = \"neighbourhood_group\", y=\"Listing\",
         data = Neighbourhood_group,color=\"#FF5A5F\")
         for bar in ax.containers:
             ax.bar_label(bar)
         plt.title(\"List of Neighbourhood_Group\",color = \"Red\", fontsize = 12)
         plt.show()
```

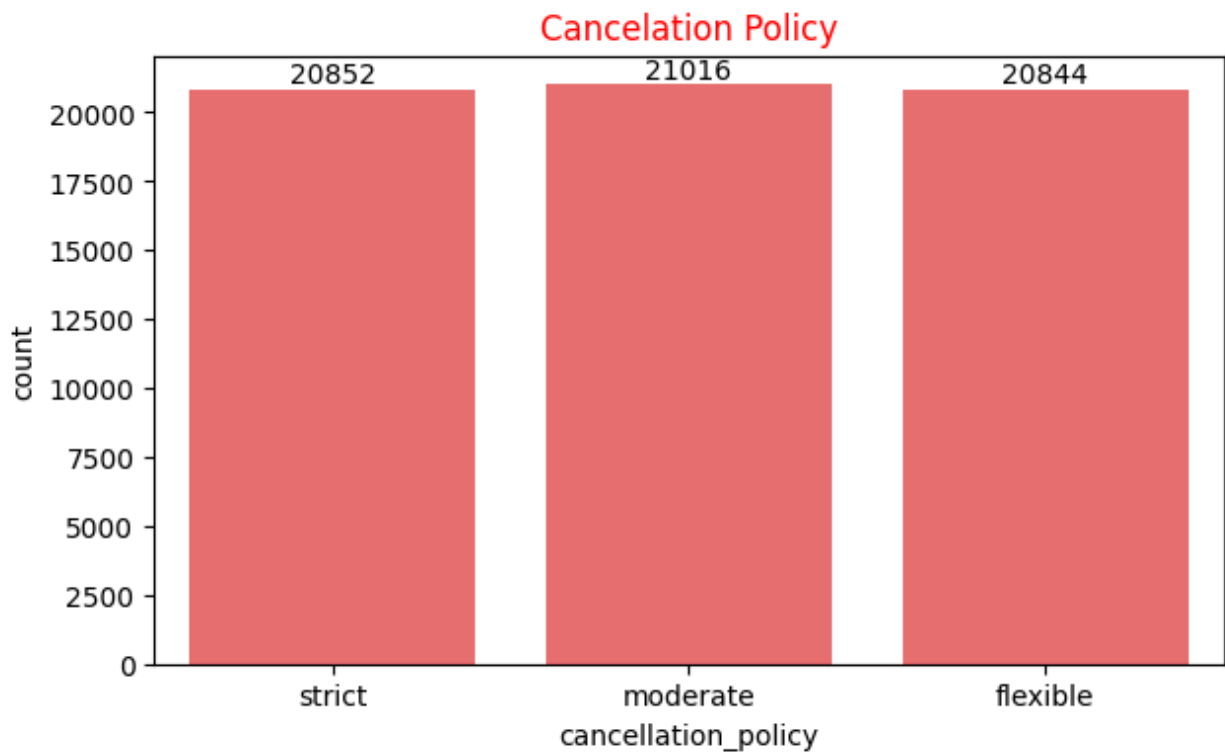


Insight

Brooklyn and Manhattan contribute the highest total price because they have the largest number of listings.

5.Cancelation policy Analysis

```
In [ ]: plt.figure(figsize=(7,4))
ax = sns.countplot(x = "cancellation_policy", data=Airbnb_Clean,color="#FF5A5F")
for bar in ax.containers:
    ax.bar_label(bar)
plt.title("Cancellation Policy",color = "Red", fontsize = 12)
plt.show()
```

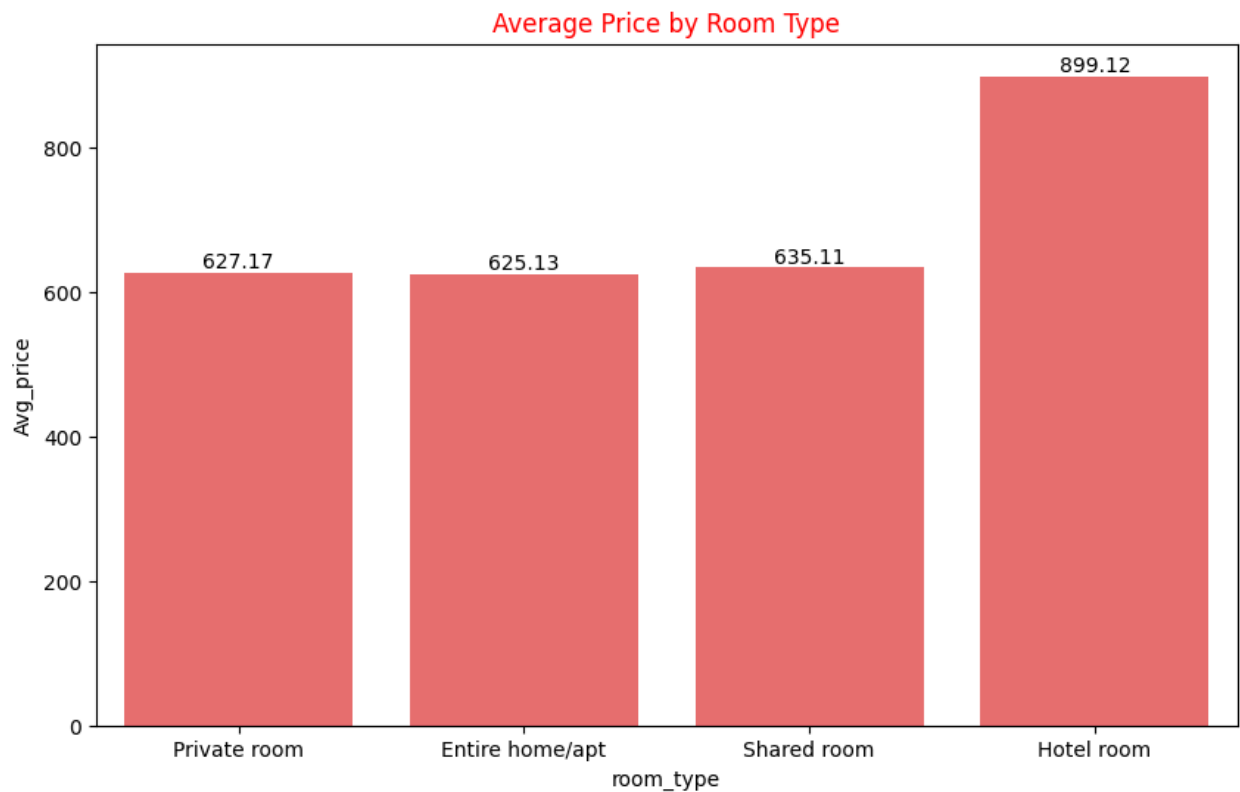


Insights

- The distribution of cancellation policies is almost evenly balanced across listings.
- The **moderate** cancellation policy is used slightly more frequently than strict and flexible policies.
- The difference in counts between the three policies is minimal, indicating no dominant cancellation preference among hosts

6.Avg Price by Room Types

```
In [43]: AVg_Price_Room = pd.read_sql_query("Select room_type , round(avg(price),2) as  
plt.figure(figsize=(10,6))  
ax = sns.barplot(x = "room_type", y="Avg_price",  
data= AVg_Price_Room, color="#FF5A5F")  
for bar in ax.containers:  
    ax.bar_label(bar)  
plt.title("Average Price by Room Type",  
color = "Red", fontsize = 12)  
plt.show()
```

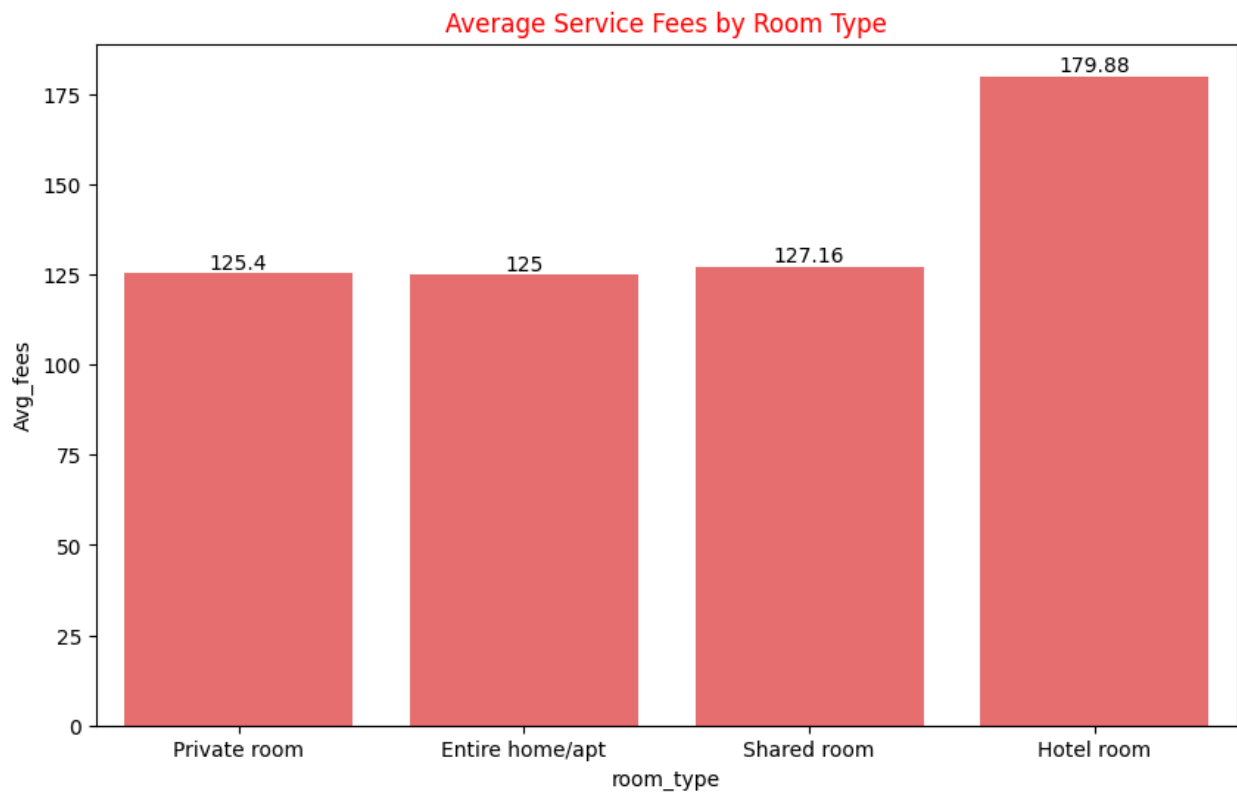


Insights.

- Hotel rooms have the highest average price at approximately 899.12, making them the most expensive room type on the platform.
- Shared rooms have an average price of about 635.11, slightly higher than private rooms and entire homes/apartments.
- Private rooms and entire homes/apartments have similar average prices, at approximately 627.17 and 625.13 respectively.
- Overall, hotel rooms are priced significantly higher than other room types, while the remaining categories fall within a similar mid-price range.

7.Service Fees by Room Type

```
In [45]: AVg_Price_Room = pd.read_sql_query("Select room_type , round(avg(service_fee),
plt.figure(figsize=(10,6))
ax = sns.barplot(x = "room_type", y="Avg_fees",
data= AVg_Price_Room, color="#FF5A5F")
for bar in ax.containers:
    ax.bar_label(bar)
plt.title("Average Service Fees by Room Type",
color = "Red", fontsize = 12)
plt.show()
```

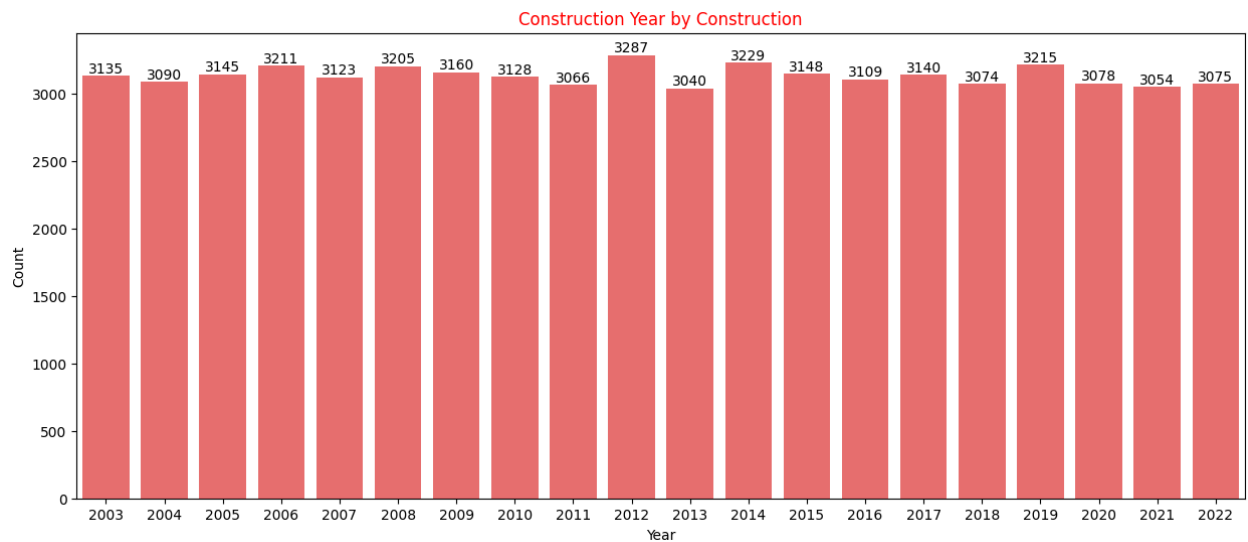


Insights

- Hotel rooms have the highest average service fee at approximately 179.88, making them the most expensive in terms of additional charges.
- Shared rooms have an average service fee of about 127.16, slightly higher than private rooms and entire homes/apartments.
- Private rooms and entire homes/apartments have very similar service fees, at approximately 125.40 and 125.00 respectively.
- Overall, service fees follow the same pricing pattern as room prices, with hotel rooms clearly positioned as a premium category.

8.Construction Year Analysis

```
In [67]: Construction = pd.read_sql_query("Select Construction_year as Year, count(*) as  
Count from Airbnb_clean group by Year", engine)  
plt.figure(figsize=(15,6))  
ax = sns.barplot(x = "Year", y= "Count",  
data= Construction, color = "#FF5A5F")  
for bar in ax.containers:  
    ax.bar_label(bar)  
plt.title("Construction Year by Construction", color = "Red",  
fontsize =12)  
plt.show()
```

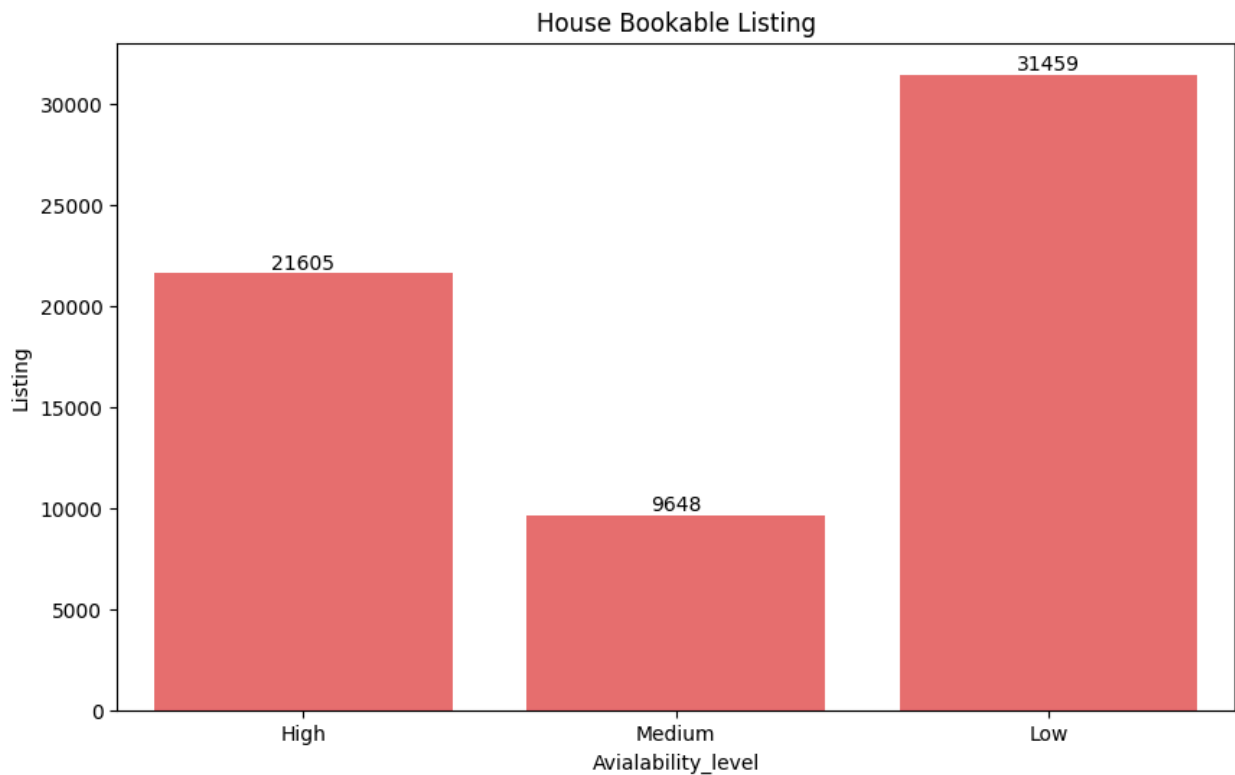


Insights

- Property construction is fairly evenly distributed across the years from 2003 to 2022.
- The highest number of properties were constructed in 2012, making it the peak construction year in the dataset.
- After 2012, construction activity remains relatively stable with no extreme rise or drop in subsequent years.
- This indicates consistent property development over time rather than sudden construction spikes.

9.House Bookable Analysis

```
In [74]: Availability = pd.read_sql_query("Select Case when availability_365<= 90 Then " \
    " 'Low' when availability_365<= 180 Then 'Medium' else 'High' end as " \
    "Avialability_level, count(*) as Listing from Airbnb_clean group by Avialabili
plt.figure(figsize=(10,6))
ax = sns.barplot(x = "Avialability_level", y = "Listing",
    data=Availability,color="#FF5A5F")
for bar in ax.containers:
    ax.bar_label(bar)
plt.title("House Bookable Listing")
plt.show()
```

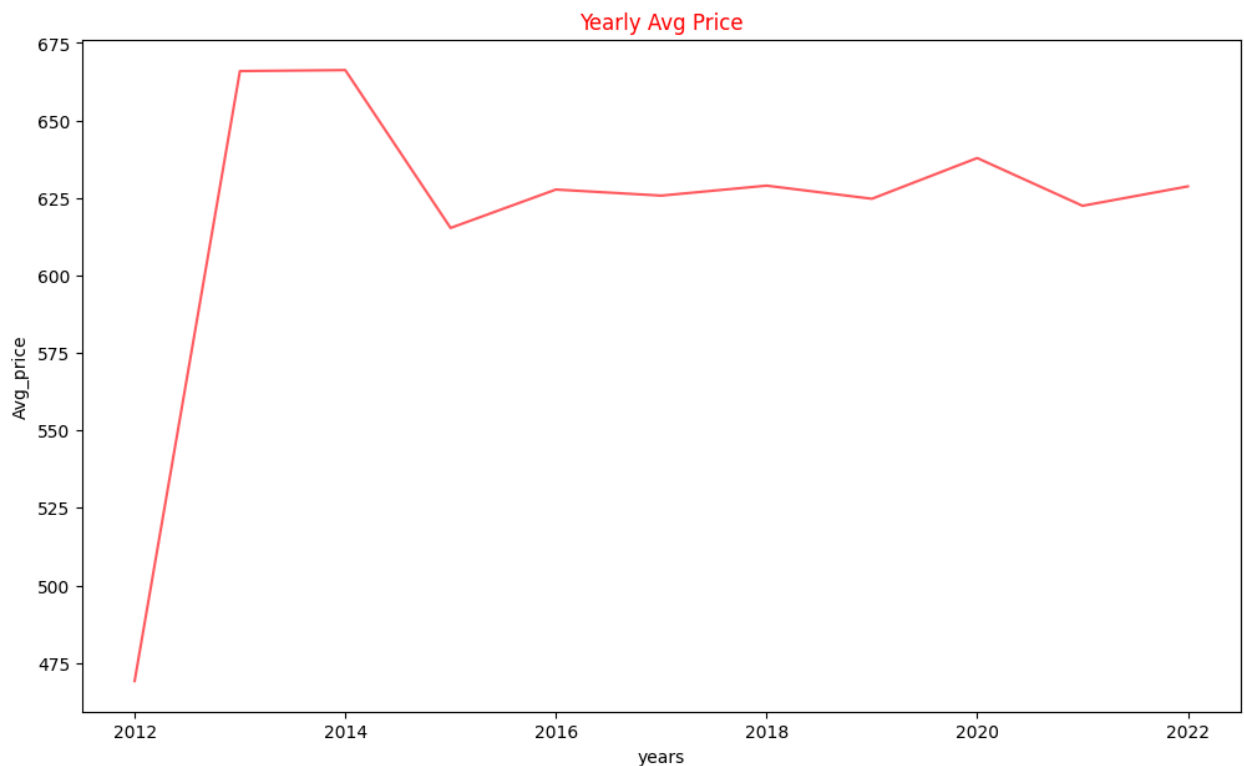



Insights

- Listings with low availability are the most common, with approximately 31,459 listings, indicating that many properties are available for a limited number of days in a year.
- High availability listings account for around 21,605 listings, showing that a significant portion of properties are available for most of the year.- -
- Medium availability listings are the least common, with about 9,648 listings.
- Overall, the distribution suggests that many hosts operate listings with limited yearly availability rather than maintaining consistent availability throughout the year.

10. Yearly Price

```
In [89]: Year_price = pd.read_sql_query("Select Year(Date) as years, Avg(price) " \
" as Avg_price from Airbnb_Clean group by Years", engine)
plt.figure(figsize=(12,7))
sns.lineplot(x = "years", y="Avg_price",
data=Year_price,color= "#FF5A5F")
plt.title("Yearly Avg Price",color = "Red", fontsize = 12)
plt.show()
```

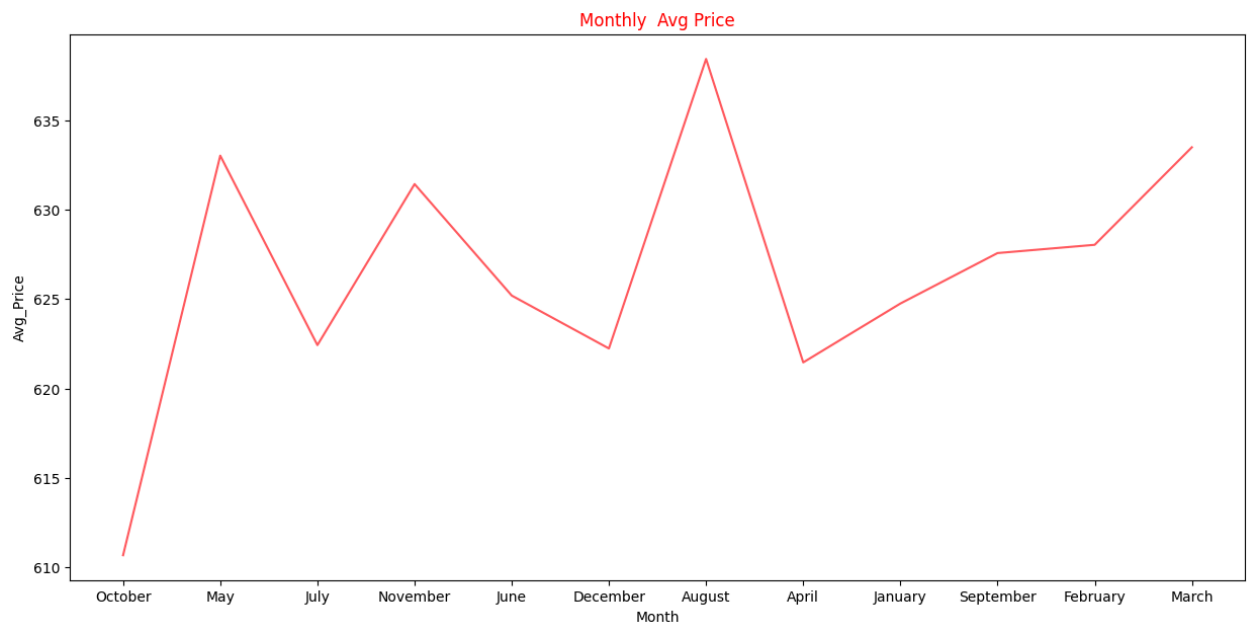


Insights

- Average price increased sharply from 2012 to 2013–2014, indicating an initial pricing correction or market adjustment phase.
- After 2015, average prices stabilized, fluctuating within a narrow range around 620–630, showing pricing consistency over time.
- A slight increase is observed around 2020, followed by a minor dip and recovery in later years.
- Overall, the trend suggests that pricing remained relatively stable after early growth, with no extreme volatility in recent years.

11.Monthly Price

```
In [88]: Month_price = pd.read_sql_query("Select MonthName(Date) as Month, avg(price) "
" as Avg_Price from Airbnb_Clean group by Month", engine)
plt.figure(figsize=(15,7))
sns.lineplot(x = "Month", y="Avg_Price",
data=Month_price,color= "#FF5A5F")
plt.title("Monthly Avg Price",color = "Red", fontsize = 12)
plt.show()
```



Insights

- Average prices remain relatively stable across months, mostly ranging between ~610 and ~640.
- August records the highest average price, indicating peak seasonal pricing during late summer.
- October shows the lowest average price, suggesting relatively lower pricing in that period.
- The overall variation across months is moderate, indicating limited seasonal fluctuation in average prices.