DS804 - Assignment 1

- By Dikshant Joshi

Q1)

- **Summary column**

Step 1 : Cleansing of Data.

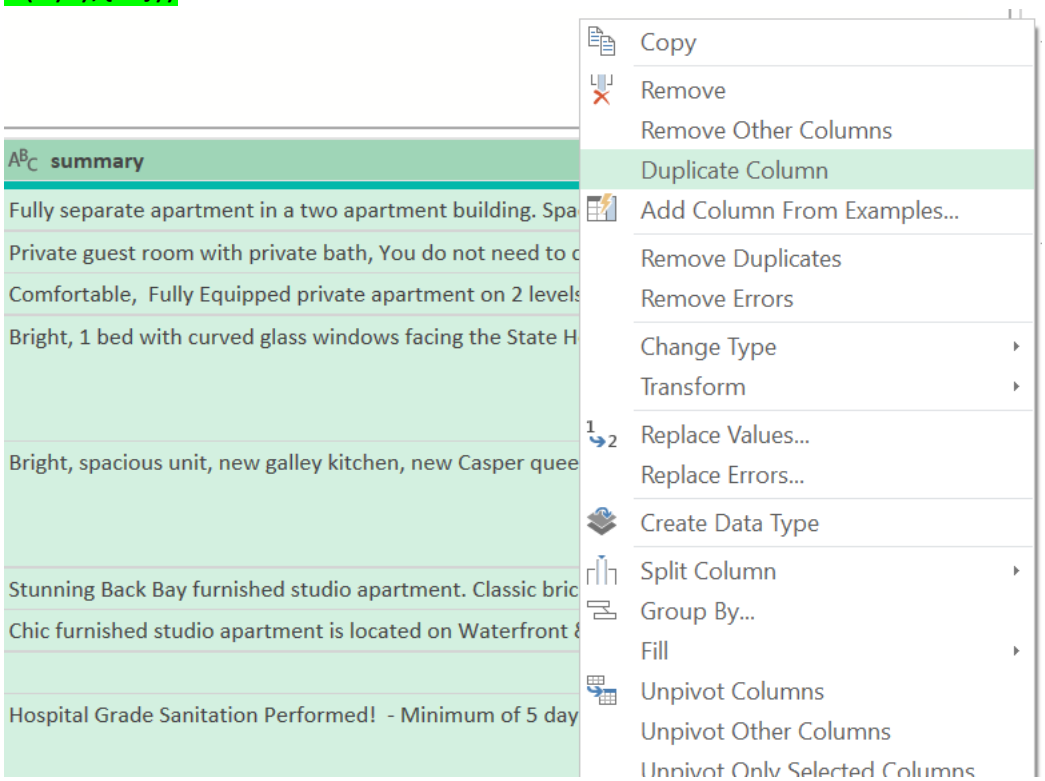
Step 2 : Duplicated summary column.

Step 3: Renamed duplicated Summary Column from Summary-copy to summary_count.

Step 4: Replaced null values with empty space in summary_count column.

Step 5: Added custom column with summary_count_number_words name and function :

```
List.Count(List.RemoveItems(Text.SplitAny([#"summary_count"],"#(tab)#(lf)"),{""}))
```



```
= Table.RenameColumns(#"Duplicated Column",{{"summary - Copy", "summary_count"}})
```

stings_count_shared_rooms	1.2 reviews_per_month	ABC summary_count
0	0.27	Fully separate apartment in a two apartment building. Space is perfect...
0	0.78	Private guest room with private bath, You do not need to cut through ...
0	0.87	Comfortable, Fully Equipped private apartment on 2 levels. Sleeps up...
0	0.35	Bright, 1 bed with curved glass windows facing the State House! High c...
0	0.25	Bright, spacious unit, new galley kitchen, new Casper queen size bed, ...
0	2.57	Stunning Back Bay furnished studio apartment. Classic brick brownsto...
0	0.04	Chic furnished studio apartment is located on Waterfront & North End...
0	null	null
0	0.42	Hospital Grade Sanitation Performed! - Minimum of 5 days between o...

```
= Table.AddColumn(#"Replaced Value4", "summary_count_number_words", each List.Count(List.RemoveItems
    (Text.SplitAny([#"summary_count"],"#(tab) #(lf)"),{""})))
```

eviews_per_month	ABC summary_count	ABC 123 summary_count_number_wo...
0.27	Fully separate apartment in a two apartment building. Space is perfect...	44
0.78	Private guest room with private bath, You do not need to cut through ...	40
0.87	Comfortable, Fully Equipped private apartment on 2 levels. Sleeps up...	22
0.35	Bright, 1 bed with curved glass windows facing the State House! High c...	31
0.25	Bright, spacious unit, new galley kitchen, new Casper queen size bed, ...	35
2.57	Stunning Back Bay furnished studio apartment. Classic brick brownsto...	63
0.04	Chic furnished studio apartment is located on Waterfront & North End...	69
null		0
0.42	Hospital Grade Sanitation Performed! - Minimum of 5 days between o...	130

➢ Space column

Followed the same above steps .

```
= Table.AddColumn(#"Replaced Value5", "space_count_number_words", each List.Count(List.RemoveItems
    (Text.SplitAny([#"space_Count"],"#(tab) #(lf)"),{""})))
```

ry_count_number_wo...	space_Count	space_count_number_words
44	This is a totally separate apartment located on the first floor of a 3 sto...	136
40	**THE BEST Value in BOSTON!!*** PRIVATE GUEST ROOM WITH PRIV...	160
22	** WELCOME *** FULL PRIVATE APARTMENT In a Historic Victorian Br...	159
31	Fully Furnished studio with enclosed bedroom. Curved glass windows ...	118
35	Bright one bed facing the golden dome of the State House. Great close...	77
63	Back Bay Studio Apt - Private bath, A/C, Cable TV, Private phone, High ...	153
69	Waterfront Studio apt on Commercial St. Totally renovated with gran...	33
0	No Frills Accommodations in Cambridge - Kendall Square, Tech Square...	17
130	Kennedy Library - Umass Boston - Castle Beach The Dorset is a fully fur...	152

➢ Description

Followed the same above steps.

```
= Table.RenameColumns(#"Added Custom2",{{"Custom", "description_count_number_words"}})
```

t_number_words	description_count	description_count_number_words
136	Fully separate apartment in a two apartment building. Space is perfect...	170
160	Private guest room with private bath, You do not need to cut through ...	163
159	Comfortable, Fully Equipped private apartment on 2 levels. Sleeps up...	157
118	Bright, 1 bed with curved glass windows facing the State House! High c...	151
77	Bright, spacious unit, new galley kitchen, new Casper queen size bed, ...	154
153	Stunning Back Bay furnished studio apartment. Classic brick brownsto...	154
33	Chic furnished studio apartment is located on Waterfront & North End...	107
17	No Frills Accommodations in Cambridge - Kendall Square, Tech Square...	17
152	Hospital Grade Sanitation Performed! - Minimum of 5 days between o...	161

- **Neighbourhood Overview**

Followed the same above steps.

```
= Table.AddColumn(#"Replaced Value7", "neighborhood_overview_count_number_words", each List.Count
    (List.RemoveItems(Text.SplitAny([#"neighborhood_overview_count"],"#(tab) #(lf)"),{""})))
```

_words	neighborhood_overview_count	neighborhood_overview_count_number_w...
170	Mostly quiet (no loud music, no crowed sidewalks) area with resident...	53
163	Peacful, Architecturally interesting, historic, diverse, and quiet residen...	14
157	Peaceful, Architecturally interesting, historic, diverse, and quiet. Racial...	13
151	Beacon Hill is a historic neighborhood filled with tradition, brownstone...	27
154	Beacon Hill is located downtown and is conveniently located to multipl...	23
154	Wander around this quintessential neighborhood in the heart of Bosto...	159
107		0
17		0
161	Once its own city, Dorchester is now Bostonâ€™s largest and most div...	67

Q2)

- **Calendar_updated**

Step 1: Duplicated Calendar_updated column.

Step 2: Replaced null values with empty space in Calendar_updated-copy column.

Step 3: Added custom column with Calendar_updated_weeks name and function :

```
if Text.Contains([#"calendar_updated"], "months")
then
Number.FromText(
   Text.Select(
      [#"calendar_updated"],{"0".."9"})
      )*4
```

else if Text.Contains([#"calendar_updated"], "weeks")

then Number.FromText(

Text.Select([#"calendar_updated"],{"0".."9"})

)

else if Text.Contains([#"calendar_updated"], "a week ago")then 1

else 0

```
= Table.RenameColumns(#"Added Custom4",{{"Custom", "calendar_updated - weeks"}})
```

	neighborhood_overview_count_number_w...	calendar_updated - Copy	calendar_updated - weeks
ident...	53	4 months ago	16
iden...	14	3 months ago	12
acial...	13	3 months ago	12
stone...	27	9 months ago	36
ultipl...	23	6 weeks ago	6
Bosto...	159	7 weeks ago	7
	0	2 months ago	8
	0	never	0
t div...	67	a week ago	1

Q3)

- **Zipcode**

Step 1: Duplicated Zipcode column.

Step 2: Changed Zipcode data type to text

Step 3: Added custom column with Zipcode_corrected name and function :

if Text.Length([#"zipcode - Copy"]) = 4

then Text.PadStart([#"zipcode - Copy"],5,"0")

else [#"zipcode - Copy"]

```
= Table.AddColumn(#"Changed Type1", "Zipcode_correct", each if Text.Length([#"zipcode - Copy"]) = 4
then Text.PadStart([#"zipcode - Copy"],5,"0")
else [#"zipcode - Copy"])
```

	calendar_updated - Copy	calendar_updated - weeks	zipcode - Copy	Zipcode_correct
3	4 months ago	16	2128	02128
4	3 months ago	12	2119	02119
3	3 months ago	12	2119	02119
7	9 months ago	36	2108	02108
3	6 weeks ago	6	2108	02108
9	7 weeks ago	7	2115	02115
0	2 months ago	8	2109	02109
0	never	0	2114	02114
7	a week ago	1	2125	02125

Q4)

- **Host Since**

Step 1: Added custom column with host_since_day name and function:

Date.DayOfWeekName([#"host_since"],"en-US")

Step 2: Added custom column with host_since_month name and function:

Date.MonthName([#"host_since"],"en-US")

Step 3: Added custom column with host_since_year name and function :

Date.Year([#"host_since"])

host_since_day	host_since_month	host_since_year
Wednesday	December	2008
Thursday	February	2009
Thursday	February	2009
Wednesday	July	2009
Wednesday	July	2009
Wednesday	September	2009
Wednesday	September	2009
Wednesday	June	2009
Tuesday	January	2010

- **Host_Since_Duration_Days**

Step 1: Created a new column with function :

Host_Since_Duration_Days = DATEDIFF(listings[host_since],TODAY(),DAY)

```
1 Host_Since_Duration_Days = DATEDIFF(listings[host_since],TODAY(),DAY)
```

irm mattress	Private bathroom	Bathtub with bath chair	Shower chair	Neighbourhood_join	Zipcode_corrected	price (bins)	Host_Since_Duration_Days
0	0	0	0	Roxbury	02119	$150	5008
0	0	0	0	Beacon Hill	02114	$50	4883
0	0	0	0	Dorchester	02125	$100	4674
0	0	0	0	Back Bay	02116	$100	4638
0	0	0	0	South End	02116	$100	4636
0	0	0	0	Charlestown	02129	$100	4463
0	0	0	0	Roxbury	02119	$50	4414
0	0	0	0	Roxbury	02119	$50	4414
0	0	0	0	Charlestown	02129	$100	4268
0	0	0	0	Roxbury	02119	$200	5008
0	0	0	0	Roxbury	02119	$200	4136
0	0	0	0	South Boston	02127	$200	4024
0	0	0	0	Roxbury	02118	$200	4134
0	0	0	0	West Roxbury	02132	$50	4045
0	0	0	0	Roxbury	02118	$200	4134
0	0	0	0	Dorchester	02125	$200	4674
0	0	0	0	Dorchester	02125	$150	4674
0	0	0	0	Dorchester	02125	$100	4674
0	0	0	0	Dorchester	02125	$100	4674
0	0	0	0	Jamaica Plain	02130	$50	3780

Q5)

- **Host Location**

Step 1: Duplicated host_location column

Step 2: Added custom column with Custom.1 name and function, to complete the given location so that data is clean:

```
if [#"host_location - Copy"] = "Boston, MA" then "Boston, Massachusetts, United States"

else if [#"host_location - Copy"] = "Boston, Massachusetts" then "Boston, Massachusetts, United States"

else if [#"host_location - Copy"] = "PK" then "Pakistan"

else if [#"host_location - Copy"] = "ID" then "Indonesia"

else if [#"host_location - Copy"] = "TN" then "Tennessee, United States"

else if [#"host_location - Copy"] = "Roslindale (part of Boston), MA" then "Boston, Massachusetts, United States"

else if [#"host_location - Copy"] = "Massachusetts" then "Massachusetts, United States"

else if [#"host_location - Copy"] = "Texas" then "Texas, United States"

else if [#"host_location - Copy"] = "US" then "United States"

else if [#"host_location - Copy"] = "CN" then "China"

else if [#"host_location -Copy"] = "Boston, From Jamaica " then "Boston, Massachusetts, United States "

else [#"host_location - Copy"]
```

Step 3: Added custom column with Custom.2 name and function, to count number of commas in column Custom.1:

```
List.Count( Text.ToList(Text.Select([Custom.1], ",")))
```

Step 4: Added custom column with Custom.3 name and function, to make the number of delimiters equal in column Custom.1 with the help of column Custom.2:

```
if [Custom.2]=0 then ",,"& [Custom.1]
```

else if [Custom.2]=1 then ","& [Custom.1]

else [Custom.1]

Step 5: Split the column Custom.3 and rename the columns generated after splitting.

```
= Table.DuplicateColumn(#"Added Custom8", "host_location", "host_location - Copy")
```

host_since_day	host_since_month	host_since_year	host_location - Copy
dnesday	December	2008	Massachusetts
ırsday	February	2009	Boston, Massachusetts, United States
ırsday	February	2009	Boston, Massachusetts, United States
dnesday	July	2009	Boston, Massachusetts, United States
dnesday	July	2009	Boston, Massachusetts, United States
dnesday	September	2009	US
dnesday	September	2009	US
dnesday	June	2009	Cambridge, Massachusetts, United States
ısday	January	2010	Boston, Massachusetts, United States

```
= Table.AddColumn(#"Duplicated Column6", "Custom.1", each if [#"host_location - Copy"] = "Boston, MA"
    then "Boston, Massachusetts, United States" else if [#"host_location - Copy"] = "Boston,
    Massachusetts" then "Boston, Massachusetts, United States" else if [#"host_location - Copy"] =
    "PK" then "Pakistan" else if [#"host_location - Copy"] = "ID" then "Indonesia" else if
    [#"host_location - Copy"] = "TN" then "Tennessee, United States" else if [#"host_location - Copy"]
```

host_since_month	host_since_year	host_location - Copy	Custom.1
cember	2008	Massachusetts	Massachusetts, United States
ɔruary	2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...
ɔruary	2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...
y	2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...
y	2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...
ɔtember	2009	US	United States
ɔtember	2009	US	United States
ıe	2009	Cambridge, Massachusetts, United States	Cambridge, Massachusetts, U...
ıuary	2010	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...

```
= Table.AddColumn(#"Replaced Value9", "Custom.2", each List.Count(Text.ToList(Text.Select([Custom.1],
    ","))))
```

host_since_year	host_location - Copy	Custom.1	Custom.2
2008	Massachusetts	Massachusetts, United States	1
2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...	2
2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...	2
2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...	2
2009	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...	2
2009	US	United States	0
2009	US	United States	0
2009	Cambridge, Massachusetts, United States	Cambridge, Massachusetts, U...	2
2010	Boston, Massachusetts, United States	Boston, Massachusetts, Unite...	2

```
= Table.AddColumn(#"Added Custom10", "Custom.3", each if [Custom.2]=0 then ",,"& [Custom.1] else if
    [Custom.2]=1 then ","& [Custom.1] else [Custom.1])
```

n - Copy	Custom.1	Custom.2	Custom.3
	Massachusetts, United States	1	,Massachusetts, United States
usetts, United States	Boston, Massachusetts, Unite...	2	Boston, Massachusetts, United States
usetts, United States	Boston, Massachusetts, Unite...	2	Boston, Massachusetts, United States
usetts, United States	Boston, Massachusetts, Unite...	2	Boston, Massachusetts, United States
usetts, United States	Boston, Massachusetts, Unite...	2	Boston, Massachusetts, United States
	United States	0	,,United States
	United States	0	,,United States
achusetts, United States	Cambridge, Massachusetts, U...	2	Cambridge, Massachusetts, United States
usetts, United States	Boston, Massachusetts, Unite...	2	Boston, Massachusetts, United States

```
= Table.RenameColumns(#"Changed Type2",{{"Custom.3.1", "host_location_city"}, {"Custom.3.2",
    "host_location_state"}, {"Custom.3.3", "host_location_country"}})
```

	Custom.2	host_location_city	host_location_state	host_location_country
tes	1		Massachusetts	United States
ite...	2	Boston	Massachusetts	United States
ite...	2	Boston	Massachusetts	United States
ite...	2	Boston	Massachusetts	United States
ite...	2	Boston	Massachusetts	United States
	0			United States
	0			United States
, U...	2	Cambridge	Massachusetts	United States
ite...	2	Boston	Massachusetts	United States

Q6)

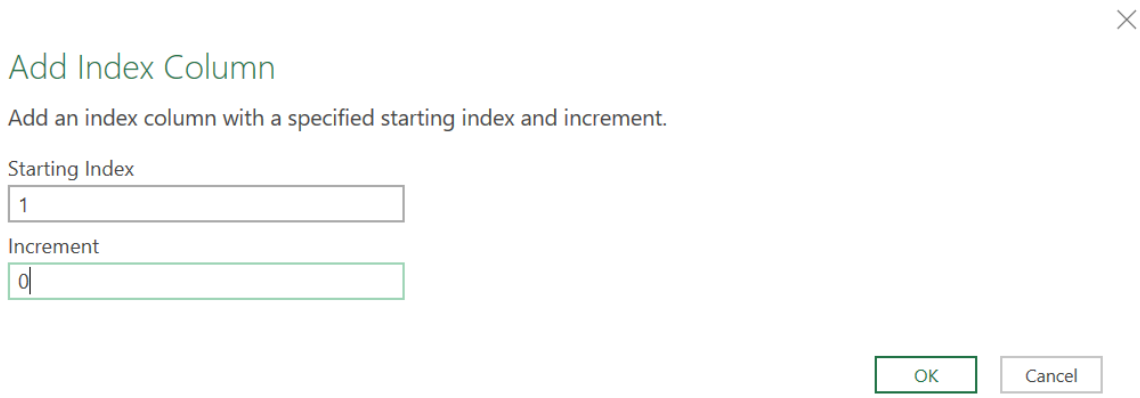
- **Host Response Time**

Step 1: Duplicated host_response_time column.

Step 2: Removed null values in host_response_time – Copy column with Host_Response_Time_Null.

Step 3: Created an Index column

Step 4: Pivot the host_response_time – Copy, replaced null values with 0 and renamed all the columns appropriately.



host_response_time - Copy	Index
within a day	1
within an hour	1
within an hour	1
within a few hours	1
within a few hours	1
within a few hours	1
within a few hours	1
within a day	1
within a few hours	1
a few days or more	1
within an hour	1
N/A	1

	host_reponse_time_within a day	host_reponse_time_within an hour	host_reponse_time_within a few hours	host_reponse_time_a few days or more	host_reponse_time_N/A	host_reponse_
1	1	0	0	0	0	
2	0	1	0	0	0	
3	0	1	0	0	0	
4	0	0	1	0	0	
5	0	0	1	0	0	
6	0	0	1	0	0	
7	0	0	1	0	0	
8	1	0	0	0	0	
9	0	0	1	0	0	
10	0	0	0	1	0	
11	0	1	0	0	0	
12	0	0	0	0	1	

- **Host is Superhost**

Followed the same above steps as in Host Response Time

1.2 host_is_superhost_t	1.2 host_is_superhost_f	1.2 host_is_superhost_null
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0

- **Host Identity Verified**

Followed the same above steps as in Host Response Time

1.2 host_identity_verified_f	1.2 host_identity_verified_t	1.2 host_identity_verified_null
1	0	0
0	1	0
0	1	0
1	0	0
1	0	0
1	0	0
1	0	0
0	1	0
0	1	0
0	1	0
0	1	0
0	1	0

- **Host has profile pic**

Followed the same above steps as in Host Response Time

1.2 host_has_profile_pic_t	1.2 host_has_profile_pic_f	1.2 host_has_profile_pic_null
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0

- **Host Verification**

Step 1: Duplicated host_verification column.

Step 2: Removed [,],’ in host_verification – Copy column.

Step 3: Replaced null value in host_verification – Copy column with host_verification_null.

Step 4: Created an index column

Step 5 : Pivot the host_verification – Copy, replaced null values with 0.

1.2 email	1.2 phone	1.2 reviews	1.2 kba	1.2 jumio
1	1	1	0	
1	1	1	1	
1	1	1	1	
1	1	1	0	
1	1	1	0	
1	1	1	0	
1	1	1	0	
1	1	1	0	
1	1	1	1	
1	1	1	0	
1	1	1	1	
1	1	1	1	

➢ Amenities

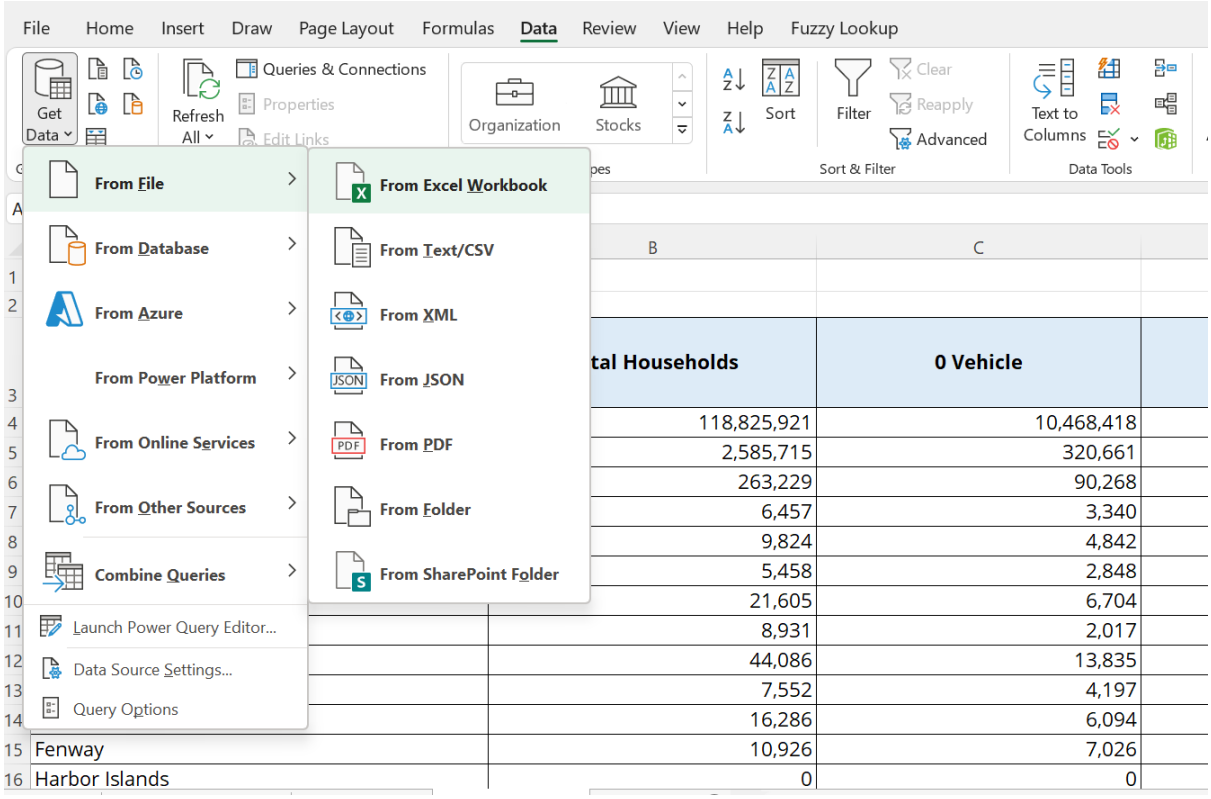
Followed same steps as that in Host Verififcation.

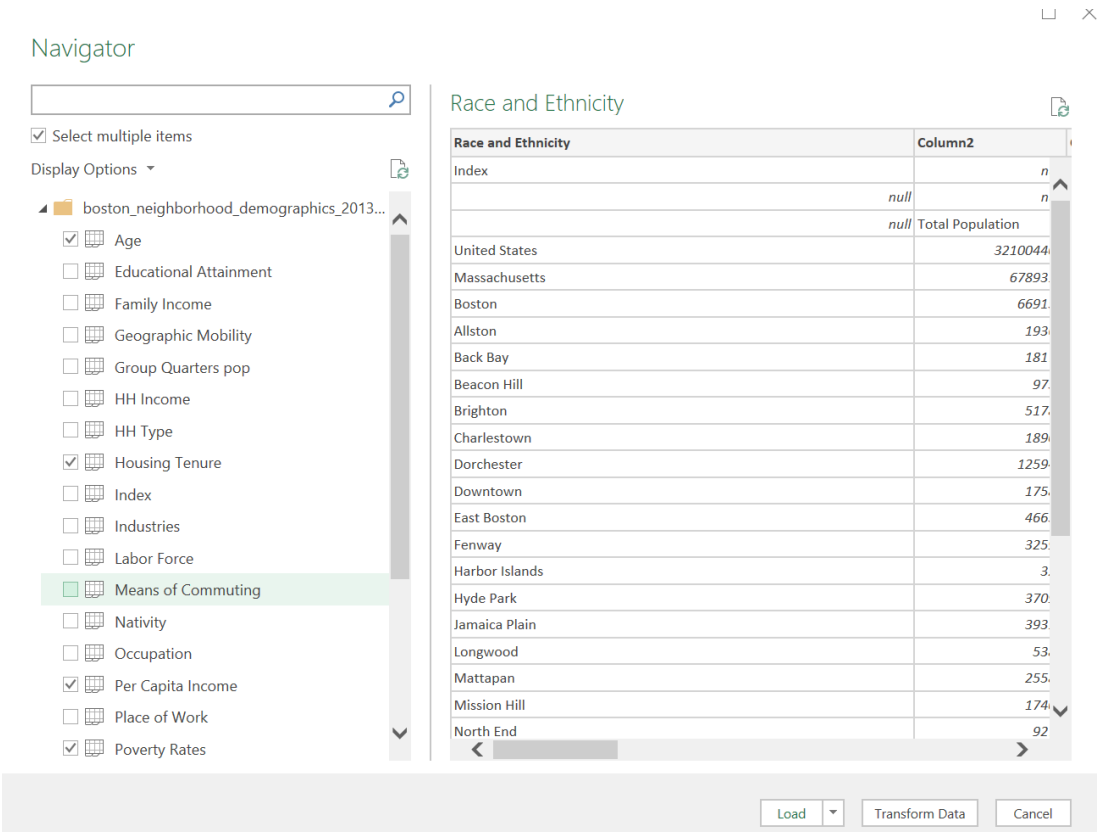
1.2 TV	1.2 Wifi	1.2 Air conditioning	1.2 Kitchen	1.2 Paid parking off premises
1	1	1	1	
1	1	1	0	
1	1	1	1	
1	1	1	1	
1	1	1	1	
1	1	1	1	
1	1	1	1	
0	1	0	0	
1	1	1	1	
1	1	1	1	
0	1	1	0	
1	1	1	1	

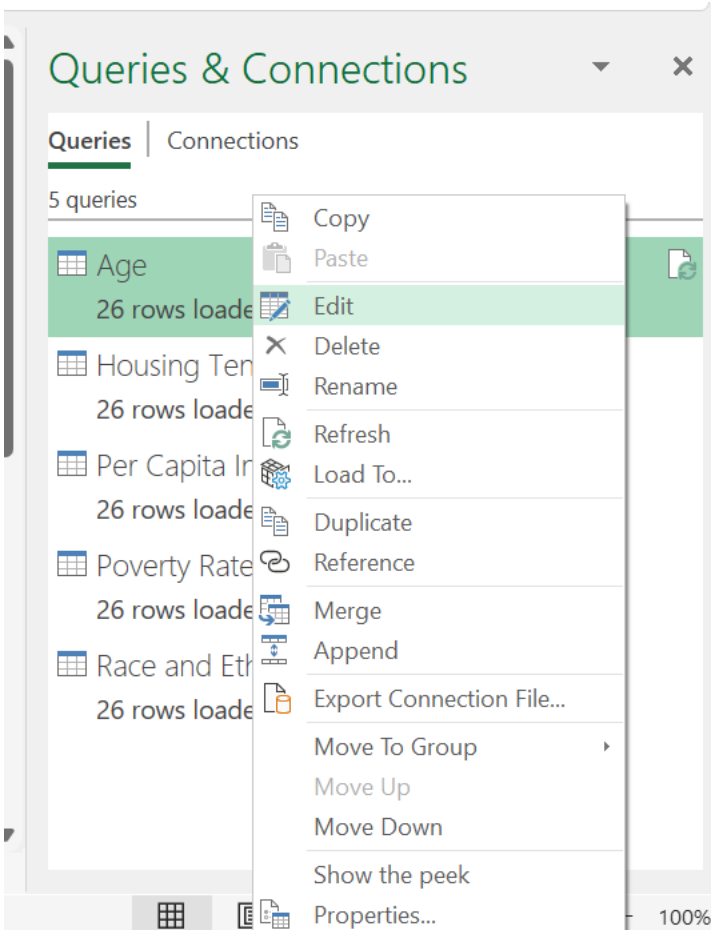
Q7)

- **Boston Neighbourhood Data**

Step 1: Imported the Boston Neighbourhood Data into Listings.

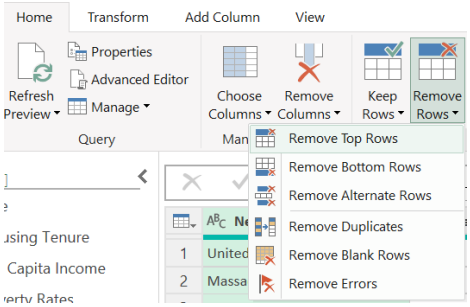


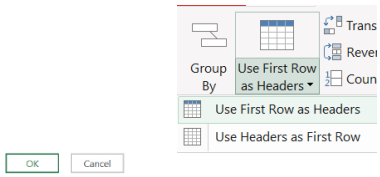




Step 2: Cleaned the data

- Removed unwanted top rows and bottom rows from each table i.e.; Age, Per Capita Income, Housing Tenure, Poverty Rates, Race and Ethnicity
- Made the top row as header
- Final cleaned data looked like the following:





Age

	Neighborhood	Total Population	Median Age	0-9 years	%
1	United States	321004407	38	40298637	0.125539.
2	Massachusetts	6789319	39	734048	0.10811
3	Boston	669158	32	63428	0.094787
4	Allston	19363	26	550	0.028404
5	Back Bay	18176	33	798	0.043904
6	Beacon Hill	9751	32	728	0.074659
7	Brighton	51785	29	3271	0.063165
8	Charlestown	18901	35	2424	0.128247.
9	Dorchester	125947	33	15270	0.121241
10	Downtown	17581	33	929	0.052841.
11	East Boston	46655	34	5666	0.121444
12	Fenway	32598	23	637	0.019541
13	Harbor Islands	322	52	0	
14	Hyde Park	37094	39	3755	0.101229.
15	Jamaica Plain	39314	34	3968	0.100930
16	Longwood	5389	20	6	0.001113.
17	Mattapan	25586	37	3115	0.121746.
18	Mission Hill	17406	26	1035	0.059462.
19	North End	9271	30	271	0.029230
20	Roslindale	29206	39	3528	0.120797

Housing Tenure

	Neighborhood	Total Housing Units:	Total Occupied	% of Total Housing Units	Owner Occupi
1	United States	135393564	118825921	0.877633452	
2	Massachusetts	2864989	2585715	0.90252179	
3	Boston	285660	263229	0.921476581	
4	Allston	7110	6457	0.908	
5	Back Bay	11773	9824	0.834	
6	Beacon Hill	6014	5458	0.908	
7	Brighton	23214	21605	0.931	
8	Charlestown	9407	8931	0.949	
9	Dorchester	47891	44086	0.921	
10	Downtown	9468	7552	0.798	
11	East Boston	17368	16286	0.938	
12	Fenway	12359	10926	0.884	
13	Harbor Islands	0	0	0	
14	Hyde Park	13419	12891	0.961	
15	Jamaica Plain	16810	16092	0.957	
16	Longwood	297	280	0.943	
17	Mattapan	9638	8866	0.92	
18	Mission Hill	6571	6270	0.954	
19	North End	5863	5338	0.91	

Per Capita Income

Neighborhood	Total population	Aggregate income in the past 12 months (in 2017 Inflation-ad...	Per Capita Inco
United States	321004407	1.00081E+13	
Massachusetts	6789319	2.70981E+11	
Boston	669158	26555981200	
Allston	19363	561260800	
Back Bay	18176	1790244400	
Beacon Hill	9751	879805600	
Brighton	51785	1857817200	
Charlestown	18901	1308312700	
Dorchester	125947	3311346200	
Downtown	17581	1195404800	
East Boston	46655	1239576900	
Fenway	32598	814861100	
Harbor Islands	322	2519300	
Hyde Park	37094	1171066500	
Jamaica Plain	39314	1836787300	
Longwood	5389	37902100	
Mattapan	25586	607122400	
Mission Hill	17406	372241900	
North End	9271	752336300	
Roslindale	29206	1132030000	

Poverty Rates

Neighborhood	Total population for whom poverty status is determined	Total in poverty	Poverty rate
United States	313048563	45650345	0.14
Massachusetts	6552347	727546	0.11
Boston	626118	128618	0.20
Allston	13892	4326	0.31
Back Bay	16661	1958	0.11
Beacon Hill	9751	907	0.09
Brighton	48366	9627	0.19
Charlestown	18816	3378	0.17
Dorchester	125342	29905	0.23
Downtown	14372	3043	0.21
East Boston	46517	9431	0.20
Fenway	18822	7672	0.40
Harbor Islands	317	247	0.77
Hyde Park	36737	4166	0.11
Jamaica Plain	38440	5961	0.15
Longwood	433	100	0.23
Mattapan	25148	5199	0.20
Mission Hill	15234	6158	0.40
North End	9211	872	0.09
Roslindale	28796	3055	0.10

Race and Ethnicity

Neighborhood	Total Population	Non-Hispanic White Alone	%	Non-Hispanic Bla
United States	321004407	197277789	0.614564114	
Massachusetts	6789319	4952367	0.729435014	
Boston	669158	300491	0.449058369	
Allston	19363	10494	0.541961473	
Back Bay	18176	13731	0.755446743	
Beacon Hill	9751	8137	0.834478515	
Brighton	51785	33674	0.650265521	
Charlestown	18901	13835	0.731971853	
Dorchester	125947	27110	0.215249272	
Downtown	17581	9914	0.563904215	
East Boston	46655	15194	0.325667131	
Fenway	32598	19598	0.601202528	
Harbor Islands	322	173	0.537267081	
Hyde Park	37094	9071	0.244540896	
Jamaica Plain	39314	21644	0.550541792	
Longwood	5389	3783	0.701985526	
Mattapan	25586	1739	0.067966857	
Mission Hill	17406	7637	0.438756751	
North End	9271	8196	0.884047028	
Roslindale	29206	14982	0.512976786	

Step 3: Resolved the differences between neighbourhood column in the listings table and the Boston data by duplicating the neighbourhood_cleansed column in listings data and making changes as following:

• Chinatown & Leather District = Downtown

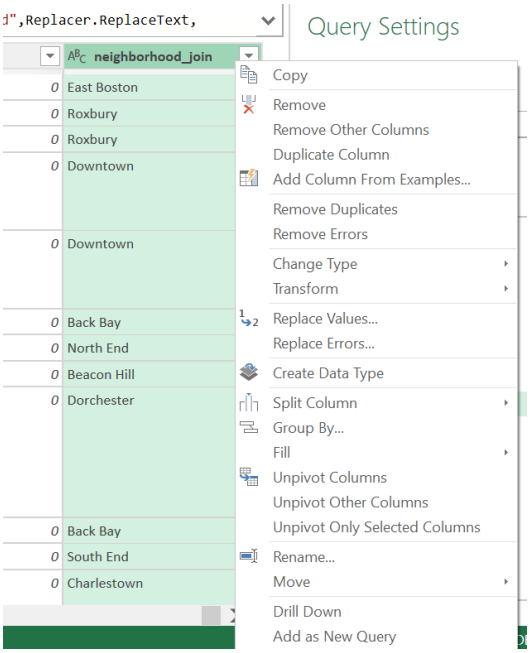
• Bay Village=South End

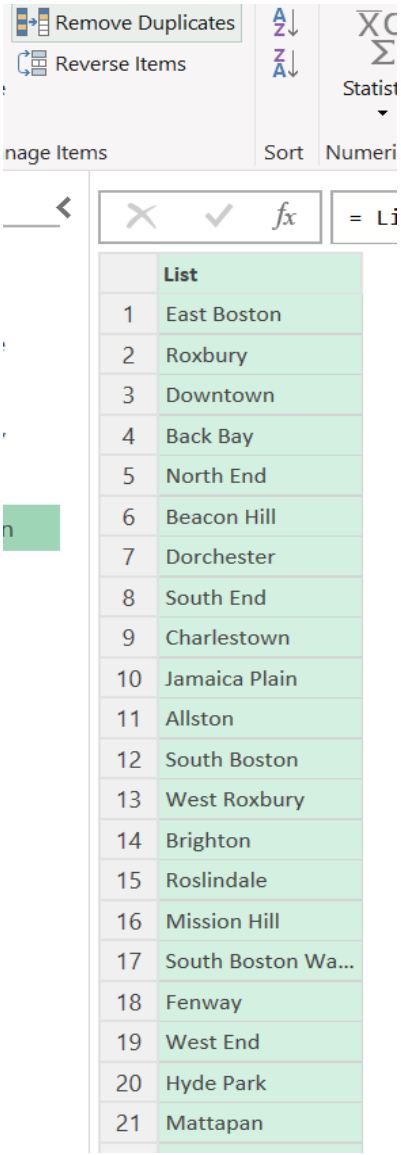
• Longwood Medical Area=Longwood

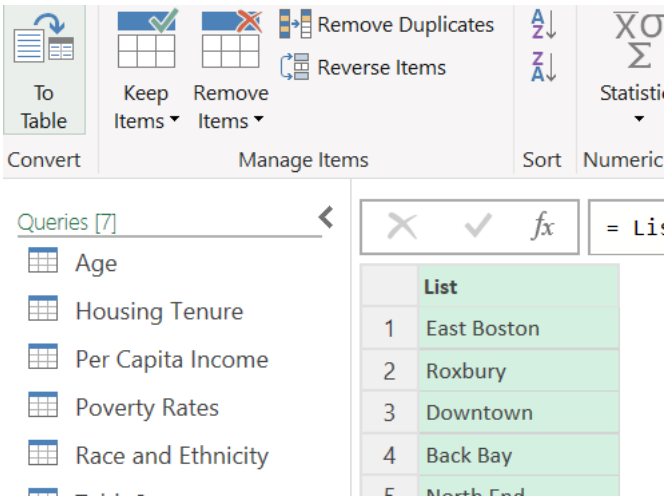
Renaming the column as Neighbourhood_join

Step 4: Creating a fact table to link Boston data to listings data

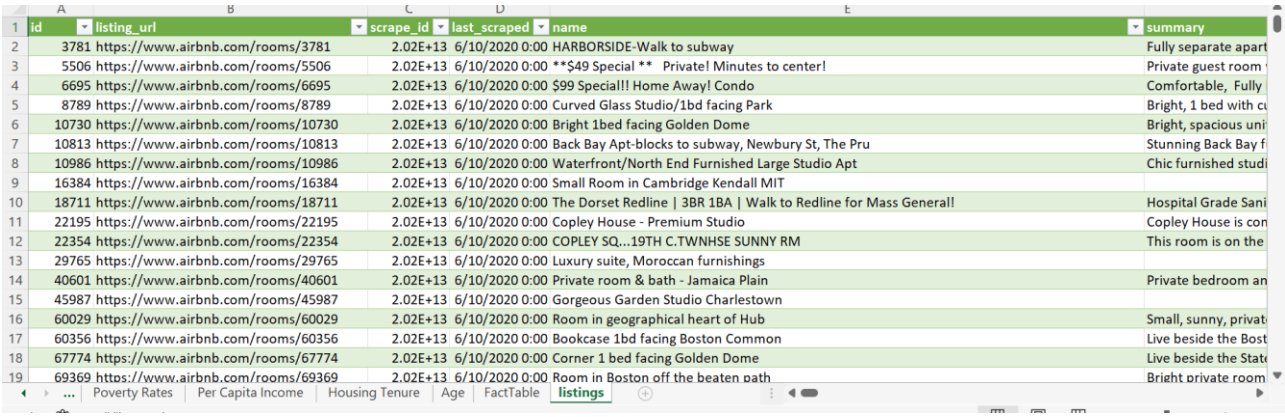
- Added the Neighbourhood_join column as a new query.
- Removed duplicates from the column.
- Transformed query into the table using To Table tool.
- Renamed the table as Fact Table with column as Neighbourhood_join
- Saved everything and closed the power query.







Final listings workbook will look like the following.

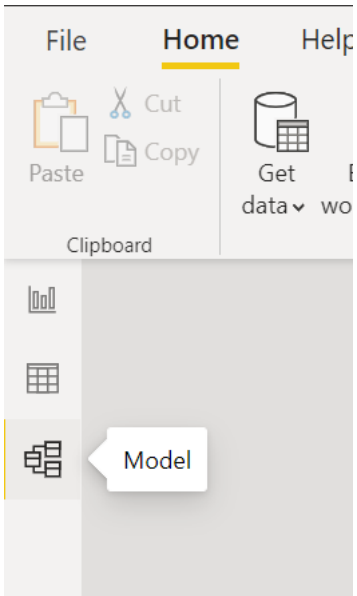


id	listing_url	scrape_id	last_scraped	name	summary
3781	https://www.airbnb.com/rooms/3781	2.02E+13	6/10/2020 0:00	HARBORSIDE-Walk to subway	Fully separate apart
5506	https://www.airbnb.com/rooms/5506	2.02E+13	6/10/2020 0:00	**$49 Special ** Private! Minutes to center!	Private guest room
6695	https://www.airbnb.com/rooms/6695	2.02E+13	6/10/2020 0:00	$99 Special!! Home Away! Condo	Comfortable, Fully
8789	https://www.airbnb.com/rooms/8789	2.02E+13	6/10/2020 0:00	Curved Glass Studio/1bd facing Park	Bright, 1 bed with cu
10730	https://www.airbnb.com/rooms/10730	2.02E+13	6/10/2020 0:00	Bright 1bed facing Golden Dome	Bright, spacious uni
10813	https://www.airbnb.com/rooms/10813	2.02E+13	6/10/2020 0:00	Back Bay Apt-blocks to subway, Newbury St, The Pru	Stunning Back Bay f
10986	https://www.airbnb.com/rooms/10986	2.02E+13	6/10/2020 0:00	Waterfront/North End Furnished Large Studio Apt	Chic furnished studi
16384	https://www.airbnb.com/rooms/16384	2.02E+13	6/10/2020 0:00	Small Room in Cambridge Kendall MIT	
18711	https://www.airbnb.com/rooms/18711	2.02E+13	6/10/2020 0:00	The Dorset Redline | 3BR 1BA | Walk to Redline for Mass General!	Hospital Grade Sani
22195	https://www.airbnb.com/rooms/22195	2.02E+13	6/10/2020 0:00	Copley House - Premium Studio	Copley House is con
22354	https://www.airbnb.com/rooms/22354	2.02E+13	6/10/2020 0:00	COPLEY SQ...19TH C.TWNHSE SUNNY RM	This room is on the
29765	https://www.airbnb.com/rooms/29765	2.02E+13	6/10/2020 0:00	Luxury suite, Moroccan furnishings	
40601	https://www.airbnb.com/rooms/40601	2.02E+13	6/10/2020 0:00	Private room & bath - Jamaica Plain	Private bedroom an
45987	https://www.airbnb.com/rooms/45987	2.02E+13	6/10/2020 0:00	Gorgeous Garden Studio Charlestown	
60029	https://www.airbnb.com/rooms/60029	2.02E+13	6/10/2020 0:00	Room in geographical heart of Hub	Small, sunny, privat
60356	https://www.airbnb.com/rooms/60356	2.02E+13	6/10/2020 0:00	Bookcase 1bd facing Boston Common	Live beside the Bost
67774	https://www.airbnb.com/rooms/67774	2.02E+13	6/10/2020 0:00	Corner 1 bed facing Golden Dome	Live beside the State
69369	https://www.airbnb.com/rooms/69369	2.02E+13	6/10/2020 0:00	Room in Boston off the beaten path	Bright private room

Q8)

- **Creating visualizations in Power BI**

Step 1: Open Powe BI and select model on the left pane.



Step 2: Click on get Data and load the listings workbook into the Power BI.

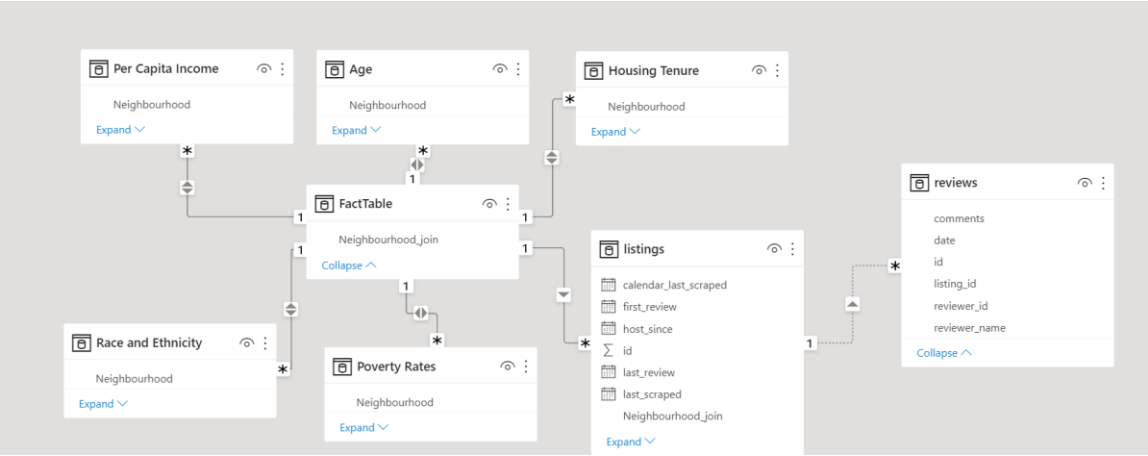
Step 3: Update the relationship between fact table and all the other tables as following:

- Listings[Neighbourhood_join] -> FactTable[Neighbourhood_join]
- Per Capita Income[Neighbourhood] -> FactTable[Neighbourhood_join]
- Age[Neighbourhood] -> FactTable[Neighbourhood_join]
- Housing Tenure[Neighbourhood] -> FactTable[Neighbourhood_join]
- Race and Ethnicity[Neighbourhood] -> FactTable[Neighbourhood_join]
- Poverty Rates[Neighbourhood] -> FactTable[Neighbourhood_join]

Step 4: Add reviews data set and create relationship with that as follows:

- Listings[id] -> reviews[listing_id]

The final model will look like the following:

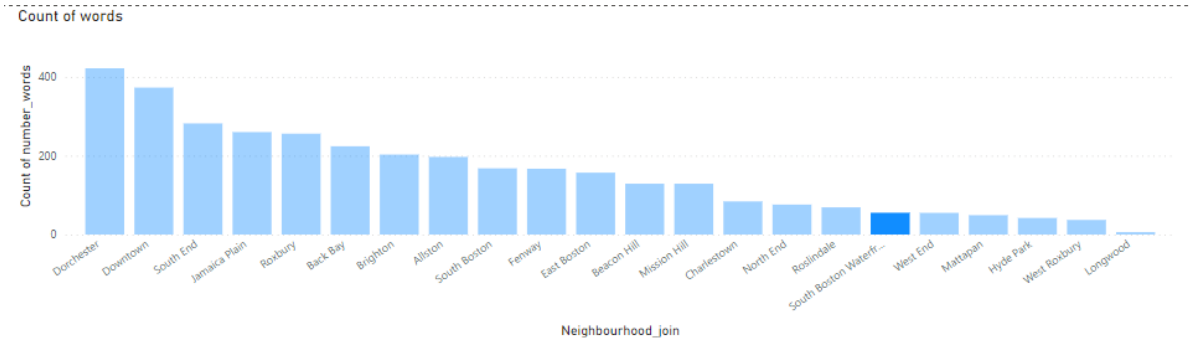


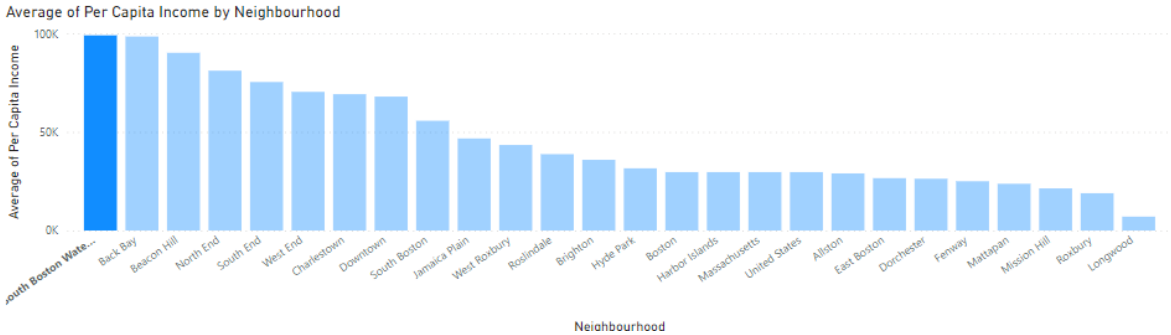
a) Do owners from more affluent neighbourhoods use more words to describe their properties?

Ans- To answer this question I used two clustered column charts as follows:

Count of words vs Neighbourhood

Average of Per Capita Income vs Neighbourhood





This clearly depicts that more affluent neighbourhood does not use more words to describe their properties.

b) What percentage of listings were updated between 1-10 weeks?

Ans – For this I created two tables with columns count of id, calendar_updated_weeks. In one table I applied filter for weeks less than equal to 10. In other table I calculated total listings updated.

For 1- 10 weeks

weeks	listings
2	513
1	419
8	210
3	164
4	127
5	115
6	86
7	53
0	42
Total	**1729**

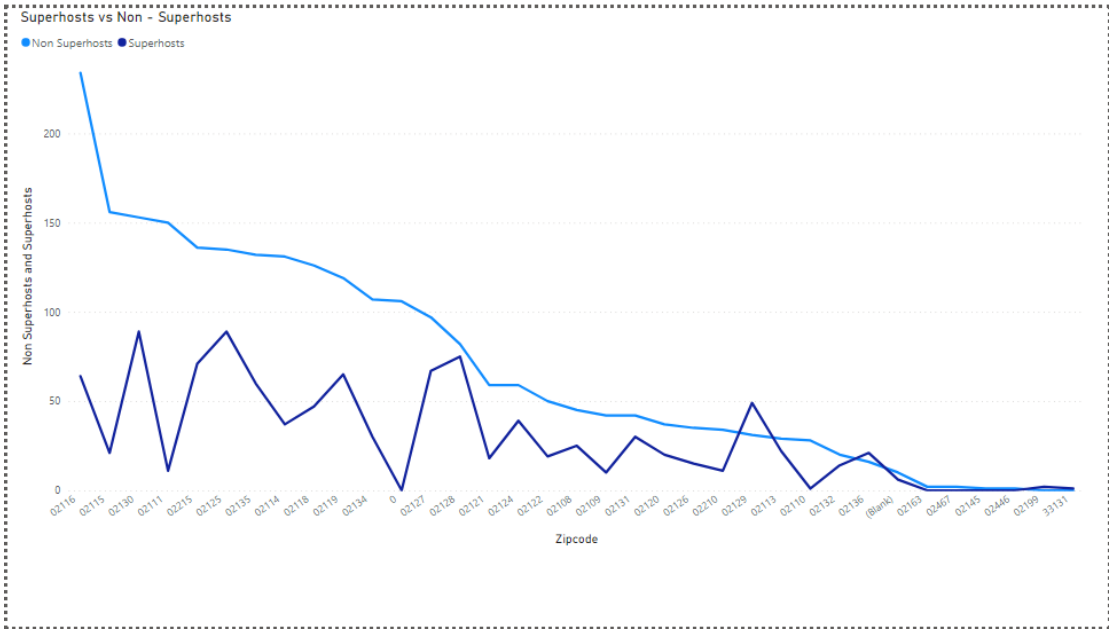
For All weeks

weeks	listings
0	42
1	419
2	513
3	164
4	127
5	115
6	86
7	53
8	210
12	445
16	172
20	195
24	107
28	135
32	93
36	156
40	57
44	38
48	29
52	24
56	10
60	12
64	6
68	7
72	7
76	8
80	3
84	2
88	6
92	5
Total	**3440**

So according to this the percentage of listings that were updated between 1-10 weeks is (1729/3440)*100 = 50.2616%.

c) Do any zip codes have more Superhosts than non Superhosts? If so, which Zip codes are they?

Ans – For this I used a line chart with zipcode on X-axis and sum of host_is_superhost_t , sum of host_is_superhost_f on Y-axis.

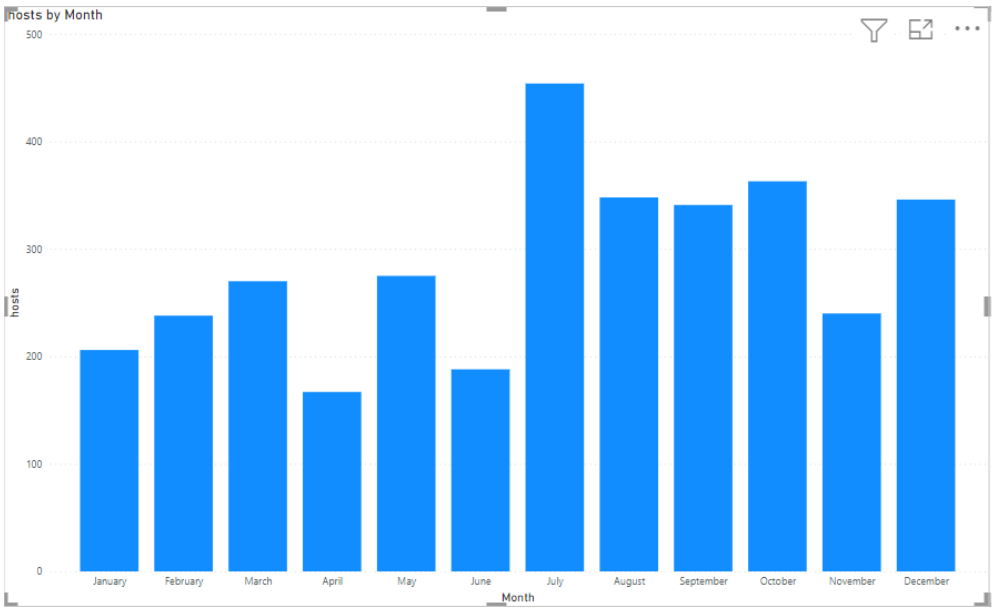


It clearly depicts that at 3 points number of Super hosts is greater than number of Non-super hosts. Zip codes for those points are:

02129, 02136, 02199

d) In which month of the year do the fewest number of people become hosts?

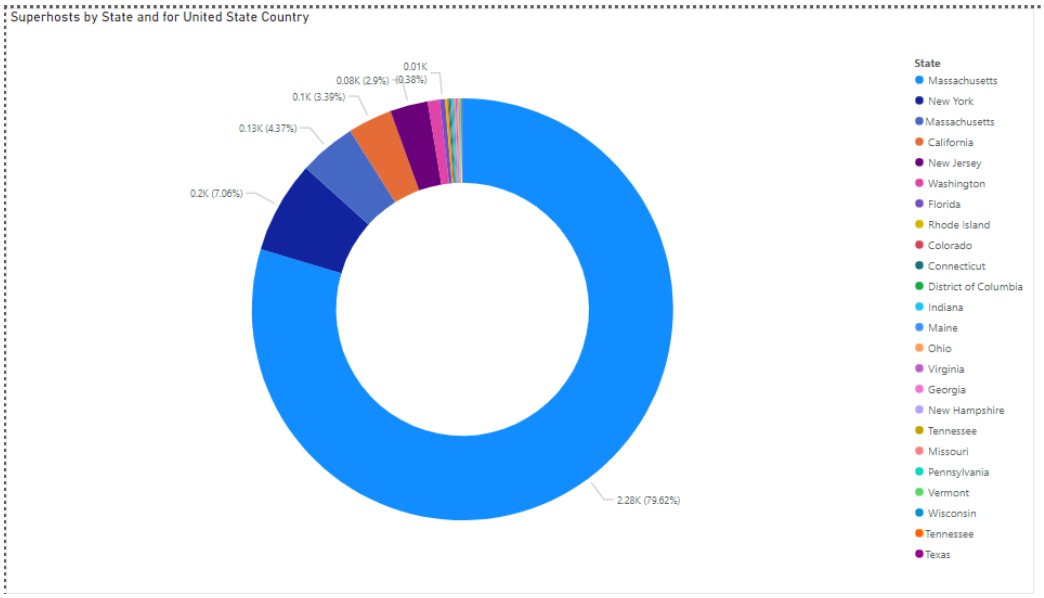
Ans – For this I used clustered column chart with count of host_since on Y-axis and host_since month on X-axis.



It clearly shows that in the month of April the fewest number of people became hosts (167).

e) What proportion of the Superhosts in the United States do host from California constitute?

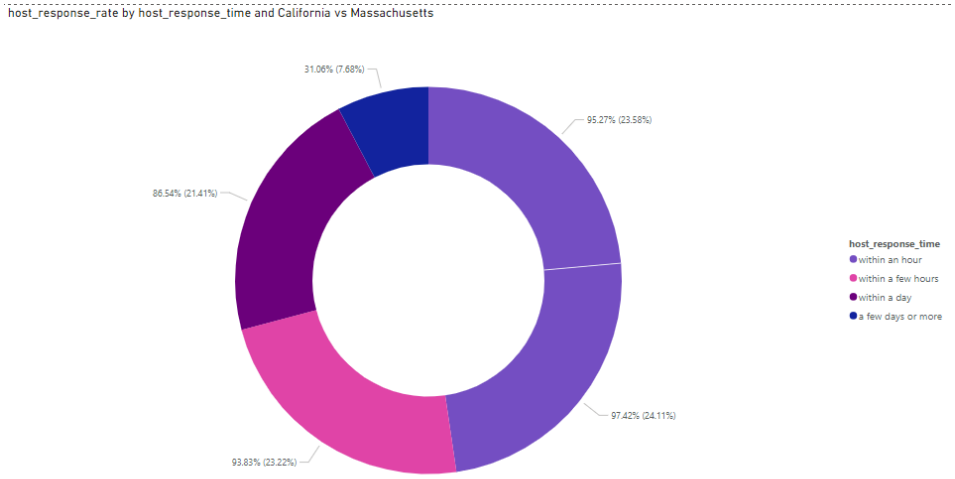
Ans – For this I used Donut Chart with legend as host_location_state, Values as Count of Host_is_superhost_t, and details as host_location_country and specified United States in filter.



This tells us that 3.39% (97 Superhosts) host from California state.

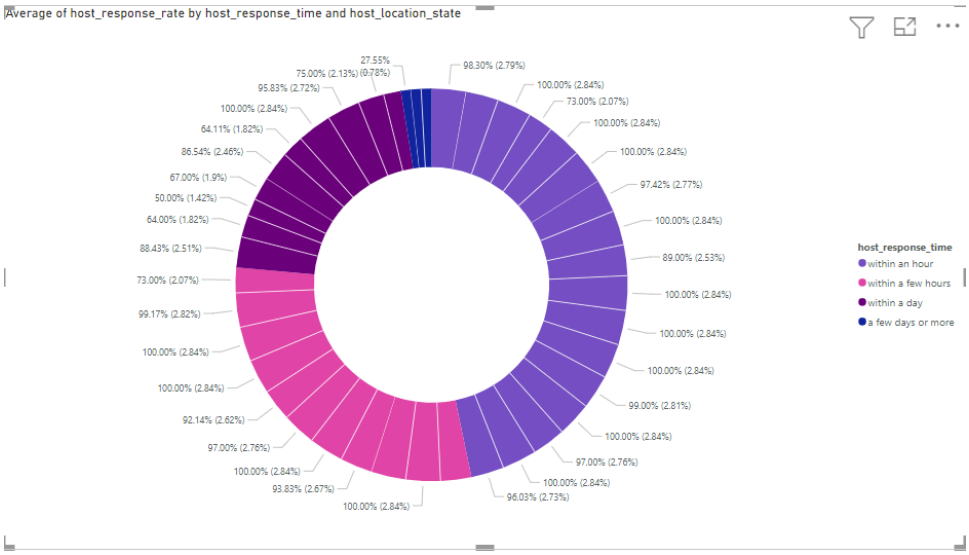
f) Is the proportion of hosts from Massachusetts who respond within an hour, larger the proportion of hosts from California who respond within an hour? Show different host response times across all the states.

Ans – For this I used donut chart with legend as host_response_time, Values as Average of Host_response_rate, and details as host_location_state and specified California, Massachusetts in filter.



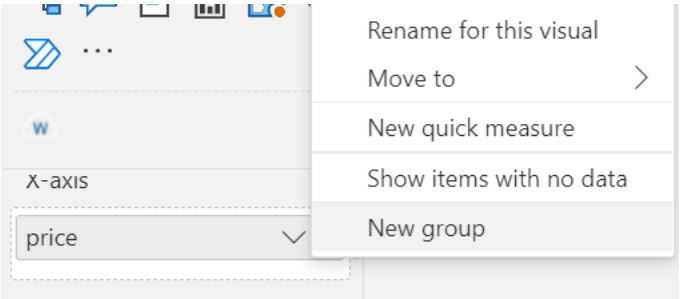
The chart tells us that proportion of hosts from Massachusetts who respond within an hour (97.42%) is larger the proportion of hosts from California who respond within an hour (95.27%).

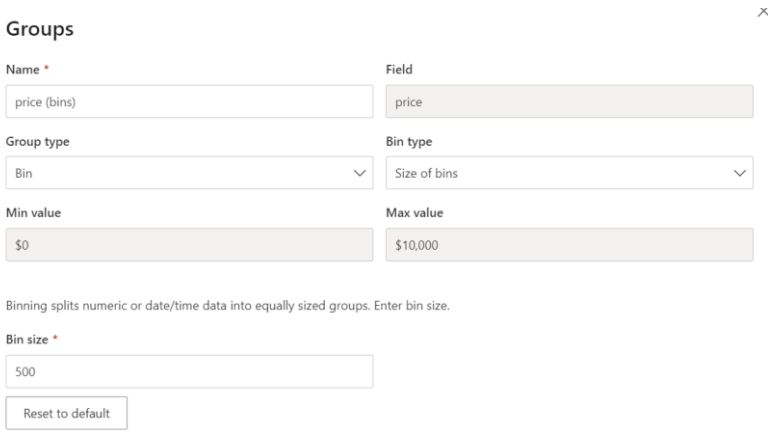
Below is the different host response times across all the states



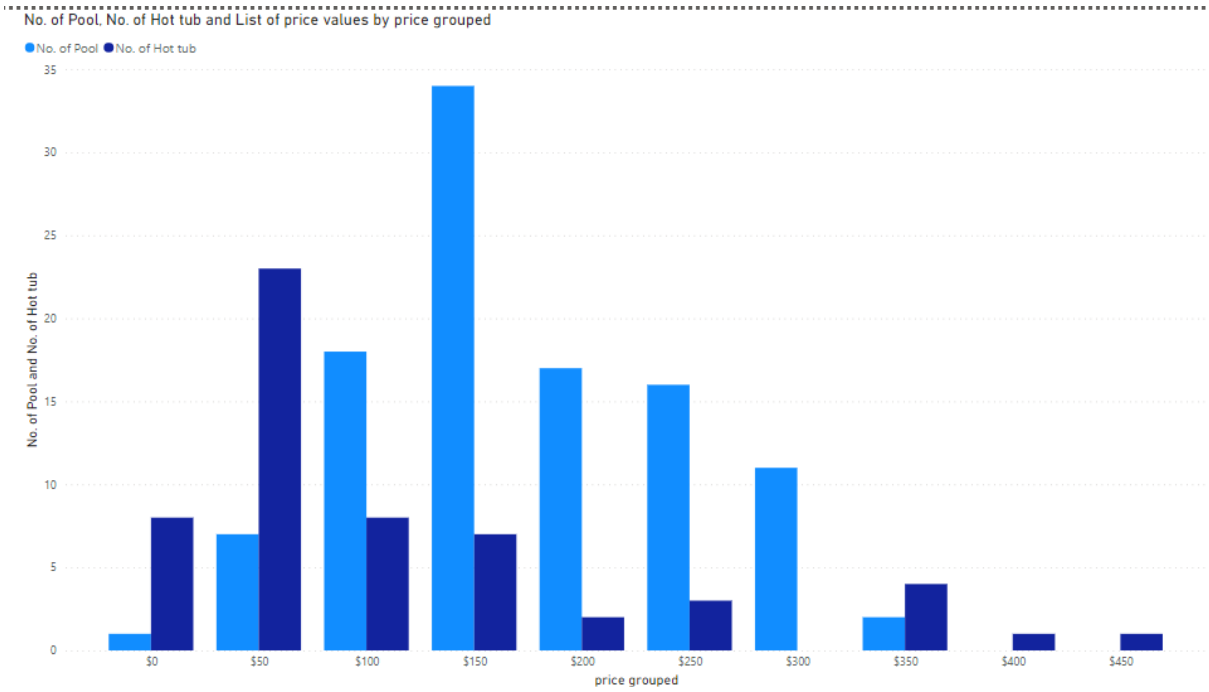
g) For listings whose price is between $1 and $500 show the number have pools and hot tubs. Property prices should be grouped shown in increments of $50

Ans – For this I used clustered column chart with price on X-axis and No. of Pool and No. of Hot tuns on Y-axis. To show the listings whose price is between $1 and $500 I applied a filter on price . To group the Property prices shown in increments of $50 I chose New group and specified the Bin Size as $50.





My final visualization looked like the following:



h) Generate summary statistic(Average, Minimum, Maximum, Standard deviation, variance, median, count) for the following columns and interpret your output [price],[review_scores_rating],[description_numberofWords],[number_of_reviews],[host_total_listings_count

Ans – For this I created different tables for different column Statistics as following visual shows.

Summary Statistic of host_total_listing_count

Average	Min	Max	Standard deviation	Variance	Median	Count
111.36	0	1920	302.71	91632.97	6	3436

Summary Statistic of Review Score Rating

Average	Min	Max	Standard deviation	Variance	Median	Count
9.29	2	10	0.92	0.85	9	2696

Summary Statistic of desciption_numberOfWords

Average	Min	Max	Standard deviation	Variance	Median	Count
143.70	0	202	45.22	2044.88	161	3440

Summary Statistic of Number of Reviews

Average	Min	Max	Standard deviation	Variance	Median	Count
40.96	0	615	69.70	4858.71	11	3440

Summary Statistic of Price

Average	Min	Max	Standard deviation	Variance	Median	Count
$181.7363372093023	$0	$10,000	$428.6615872153998	$183,750.7563540258	$130	3440

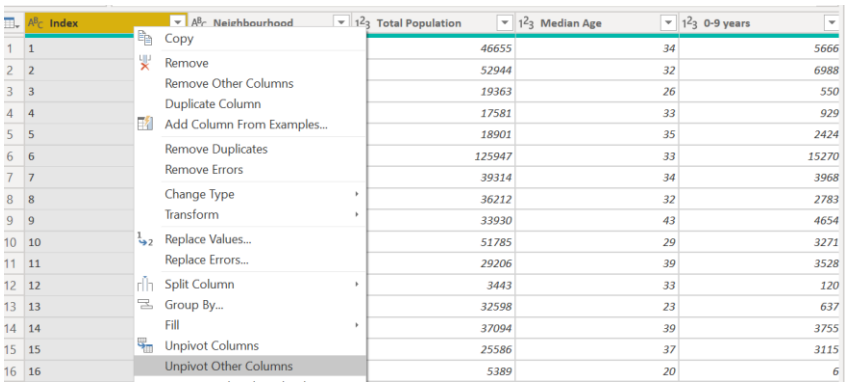
i) Generate a correlation matrix for each of the following columns and interpret your output [price],[review_scores_rating],[description_numberofWords],[Total Population],[Median Age],[Per Capita Income],[number_of_reviews],[host_total_listings_count],[host_response_time_within an hour].

Ans:

Step 1:Merged Age,listings and Per Capita Income table into one table merge.

Step 2: Created an index for the Merge table.

Step 3: Created a table Merge_pivot with source = merge and Unpivoted other columns.



Step 4: Filtered the rows for which correlation is required.

Step 5: Created 2 tables Attribute row, Attribute column with source as Merged_pivot. Select the attribute column and select remove other columns.

Step 6: Remove duplicates .

Step 7: Filtered the rows required for the correlation plot. Close and apply

Step 8: Created measures for correlation(n,X,XY,Y,X^2,Y^2 and correlation_coeff). Following are the code:

- XY =

```
VAR CurrentX = SELECTEDVALUE('Attribute row'[Attribute])
VAR CurrentY = SELECTEDVALUE('Attribute column'[Attribute])

VAR VIRTUAL =
SUMMARIZE(
    'Merge_pivot',
    'Merge_pivot'[Index],
    "X", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentX),
    "Y", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentY)

    )
    RETURN
    SUMX(
    VIRTUAL,
    [X] * [Y]
    )
```

- X =

```
VAR CurrentX = SELECTEDVALUE('Attribute row'[Attribute])
VAR CurrentY = SELECTEDVALUE('Attribute column'[Attribute])

VAR VIRTUAL =
SUMMARIZE(
    'Merge_pivot',
    'Merge_pivot'[Index],
    "X", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentX),
    "Y", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentY)

    )
    RETURN
    SUMX(
    VIRTUAL,
    [X]
    )
```

- X^2 =

```
VAR CurrentX = SELECTEDVALUE('Attribute row'[Attribute])
VAR CurrentY = SELECTEDVALUE('Attribute column'[Attribute])

VAR VIRTUAL =
SUMMARIZE(
    'Merge_pivot',
    'Merge_pivot'[Index],
    "X", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentX),
    "Y", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentY)

    )
    RETURN
    SUMX(
    VIRTUAL,
    [X] * [X]
    )
```

- Y^2 =

```
    VAR CurrentX = SELECTEDVALUE('Attribute row'[Attribute])
    VAR CurrentY = SELECTEDVALUE('Attribute column'[Attribute])

    VAR VIRTUAL =
    SUMMARIZE(
    'Merge_pivot',
    'Merge_pivot'[Index],
    "X", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentX),
    "Y", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentY)

    )
    RETURN
    SUMX(
    VIRTUAL,
    [Y] * [Y]
    )
```

- Y =

```
VAR CurrentX = SELECTEDVALUE('Attribute row'[Attribute])
VAR CurrentY = SELECTEDVALUE('Attribute column'[Attribute])

VAR VIRTUAL =
SUMMARIZE(
 'Merge_pivot',
 'Merge_pivot'[Index],
 "X", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentX),
 "Y", CALCULATE(MAX('Merge_pivot'[Value]),'Merge_pivot'[Attribute] = CurrentY)

    )
    RETURN
    SUMX(
    VIRTUAL,
```

```
    [Y]
    )
```

- CORRELATION COEFF =

```
DIVIDE(
    [N]*[XY] - [X]*[Y],
    SQRT(
            ([N]*[X^2] - [X]^2)*([N]*[Y^2]-[Y]^2)
    )
)
```

- n = DISTINCTCOUNT(Merge_pivot[Index])

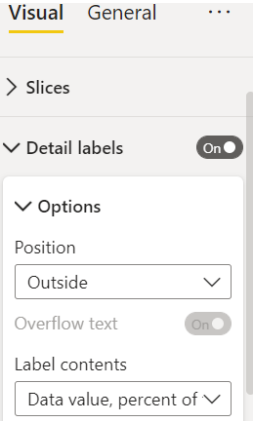
Step 9: Created a matrix with row as Attribute_row, column as Attribute_column and values as Correltion_coeff. Following is the Correlation table which we will get:

Attribute	Median Age	Sum of listings.description_count_number_words	Sum of listings.host_reponse_time_within an hour	Sum of listings.host_total_listings_count	Sum of listings.number_of_reviews	Sum of listings.price	Sum of listings.review_scores_rating	Sum of Per Capita Income.Per Capita Income	Total Population	**Total**
Median Age	1.00	-0.12	-0.09	-0.16	0.00	-0.13	-0.10	0.31	0.14	
Sum of listings.description_count_number_words	-0.12	1.00	0.97	0.35	0.88	0.89	0.98	-0.03	0.65	
Sum of listings.host_response_time_within an hour	-0.09	0.97	1.00	0.36	0.88	0.86	0.96	0.05	0.66	
Sum of listings.host_total_listings_count	-0.16	0.35	0.36	1.00	-0.02	0.70	0.19	0.57	-0.28	
Sum of listings.number_of_reviews	0.00	0.88	0.88	-0.02	1.00	0.63	0.93	-0.16	0.78	
Sum of listings.price	-0.13	0.89	0.86	0.70	0.63	1.00	0.79	0.20	0.35	
Sum of listings.review_scores_rating	-0.10	0.98	0.96	0.19	0.93	0.79	1.00	-0.11	0.73	
Sum of Per Capita Income.Per Capita Income	0.31	-0.03	0.05	0.57	-0.16	0.20	-0.11	1.00	-0.34	
Total Population	0.14	0.65	0.66	-0.28	0.78	0.35	0.73	-0.34	1.00	
Total										

Q9)

- **Creating a Dashboard**

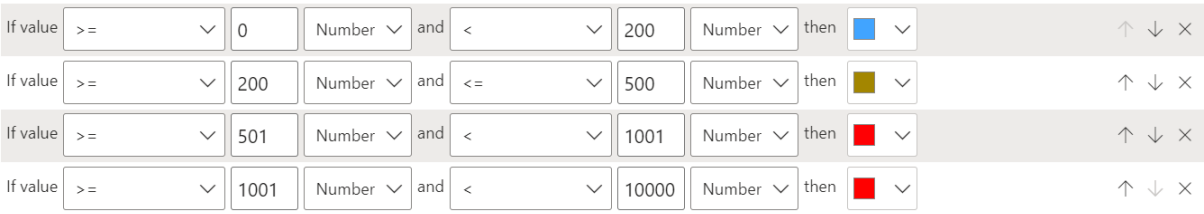
1)Superhost: Created a donut chart with **legend** as host_is_superhost and **values** as count_of_host_response_rate. Also selected label contents as Data values, Percent of total.



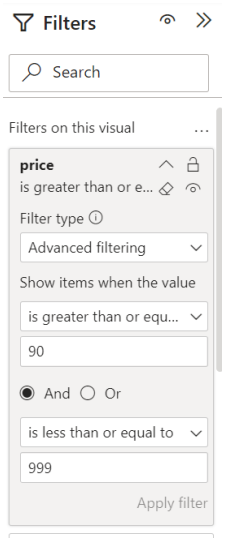
2)Count of comment vs month : Created a clustered column chart with **X-axis** as date-Month and **Y-axis** as Count of comments.

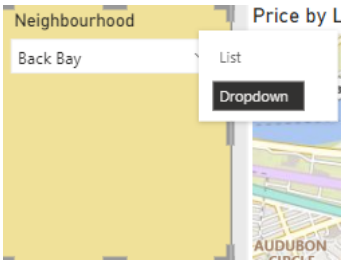
3)Comments :Created a word cloud with **category** as comments Also created a card with **fields** as count of comments.

4)Price by location : Created a Map with **latitude** as latitude and **longitude** as longitude and **Bubble size** as Average of Price. Also created a function for **bubble colors** as shown and created a card with **category** as Average of Review_Score_rating.



5)Neighbourhood, Review Score and Daily Price: Created a slicer for all three with **field** as Neighbourhood_join,price,review_score_rating respectively. Also added a **filter** for Daily price in price as follows to display only a range of price. For Neighbourhood selected dropdown as the **type** of slicer.





Finally did some text editing to display the dashboard as was displayed in the question. Final dashboard looked like the following: