USA HOUSING

Ιωάννης Καλαντζής (8200235)

Οδυσσέας Σπυρόπουλος (3200183)

Χειμερινό εξάμηνο
2023-2024

USA Listings Dataset

- Data from 385.000 Listings
- Dataset with 22 features describing the listings
- Source: https://www.kaggle.com/datasets/austinreese/usa-housing-listings?fbclid=IwAR3-tif1QU43u4R1AbmmuK9kZboVGdYfgAG3AGTJK_yEm2b3aK_ZLHbhF6U

Dataset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	id	url	region	region_url	price	type	sqfeet	beds	baths	cats_allov	dogs_allo	smoking_	wheelchai	electric_ve	comes_fu
2	7.05E+09	https://rer	reno / tah	https://rer	1148	apartmen	1078	3	2	1	1	0	0	0	0
3	7.05E+09	https://rer	reno / tah	https://rer	1200	condo	1001	2	2	0	0	0	0	0	0
4	7.04E+09	https://rer	reno / tah	https://rer	1813	apartmen	1683	2	2	1	1	1	0	0	0
5	7.05E+09	https://rer	reno / tah	https://rer	1095	apartmen	708	1	1	1	1	1	0	0	0
6	7.05E+09	https://rer	reno / tah	https://rer	289	apartmen	250	0	1	1	1	1	1	0	1
7	7.05E+09	https://rer	reno / tah	https://rer	1093	apartmen	720	1	1	1	1	1	0	0	0
8	7.05E+09	https://rer	reno / tah	https://rer	935	apartmen	661	1	1	1	1	1	0	0	0
9	7.05E+09	https://rer	reno / tah	https://rer	1095	apartmen	708	1	1	1	1	1	0	0	0
10	7.05E+09	https://rer	reno / tah	https://rer	1525	apartmen	1053	2	2	1	1	0	0	0	0
11	7.05E+09	https://rer	reno / tah	https://rer	1295	condo	930	2	2	0	0	0	0	0	0

P	Q	R	S	T	U	V
laundry_o	parking_o	image_url	descriptio	lat	long	state
w/d in uni	carport	https://im	Ridgeview	39.5483	-119.796	ca
w/d hooku	carport	https://im	Convenier	39.5026	-119.789	ca
w/d in uni	attached g	https://im	2BD | 2BA	39.6269	-119.708	ca
w/d in uni	carport	https://im	MOVE IN S	39.4477	-119.771	ca
laundry on site		https://im	Move In To	39.5357	-119.805	ca
laundry in bldg		https://im	1BD | 1BA	39.4572	-119.776	ca
laundry or	off-street p	https://im	Tucked av	39.5118	-119.802	ca
w/d in uni	carport	https://im	MOVE IN S	39.4477	-119.771	ca
w/d in uni	carport	https://im	BRAND NE	39.6185	-119.672	ca
w/d in uni	carport	https://im	6850 Shar	39.5193	-119.897	ca

Data types

- Boolean
- String
- Integer
- Float

Data Manipulation

Through Jupyter Notebook we manipulated and cleaned the data as follows:

- removal of Null elements
- deletion of columns with no analytical value(e.g. image url or region url)
- coordinates examination to classify points within the United States
- change of specific features

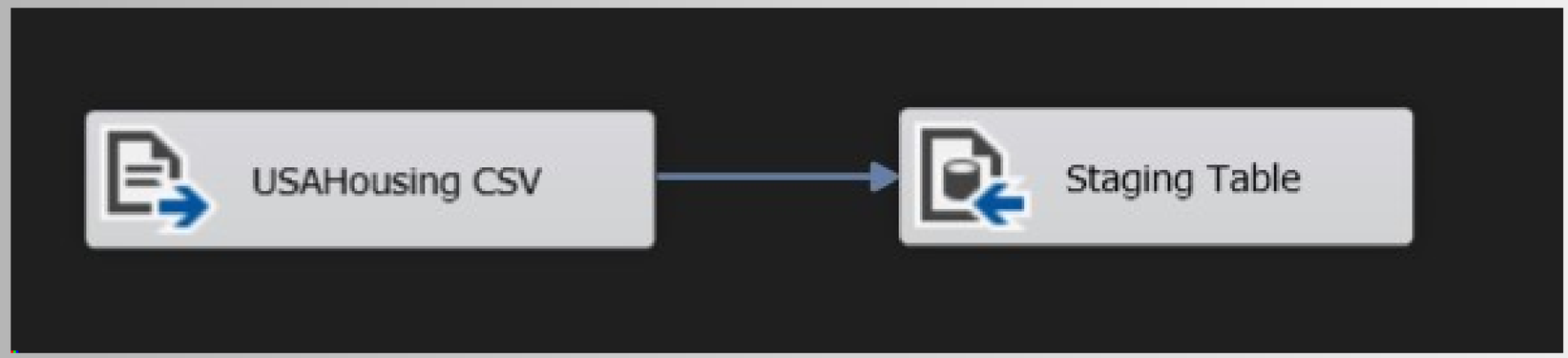
42.000 rows deleted!

Data Warehousing

We used Microsoft Visual Studios Integration Services Project to load our data to the SQL Server and to create staging, dimensions and Fact table.

Data Warehousing (1)

We started by creating our relational database in SSMS and introducing a Data Flow Task whose purpose is to create a table based on the datasets used.

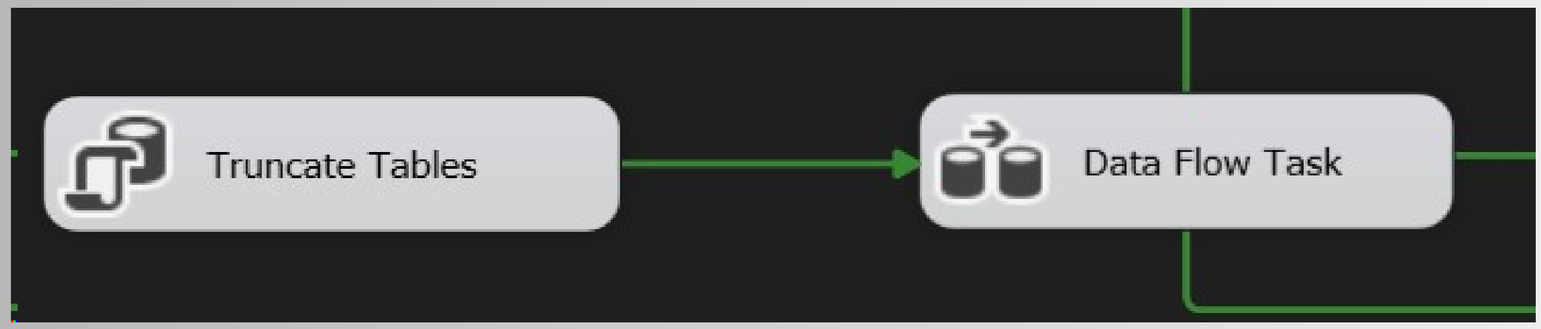


Staging Table

	id	region	price	type	sqfeet	beds	baths	cats_allowed	dogs_allowed	smoking_allowed	wheelchair_access	electric_vehicle_charge	comes_furnished	laundry_options	parking_options	lat	long	state
1	7047427478	washington DC	1170	apartment	720	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.8425	-76.9222	dc
2	7035993979	washington DC	1860	apartment	1005	2	1	1	1	1	0	0	0	w/d in unit	off-street parking	38.8372	-77.064	dc
3	7048606421	washington DC	1955	apartment	885	2	1	0	0	1	0	0	0	Unknown	Unknown	38.859	-77.0997	dc
4	7048659179	washington DC	1920	apartment	669	0	1	1	1	0	0	0	0	Unknown	Unknown	38.9025	-77.0028	dc
5	7050027023	washington DC	1448	apartment	867	1	1	1	0	1	0	0	0	laundry in bldg	Unknown	38.9719	-76.9554	dc
6	7042558238	washington DC	1705	apartment	825	2	2	0	0	1	0	0	0	Unknown	Unknown	38.9996	-76.884	dc
7	7031052315	daytona beach	875	apartment	600	1	1	0	0	0	0	0	0	Unknown	off-street parking	29.2833	-81.0882	fl
8	7050024991	washington DC	1705	apartment	1008	2	2	1	0	1	0	0	0	laundry in bldg	Unknown	38.9719	-76.9554	dc
9	7050016814	washington DC	1520	apartment	925	2	1	1	1	1	0	0	0	Unknown	Unknown	39.1821	-77.5359	dc
10	7037979176	washington DC	1310	apartment	564	1	1	1	1	1	0	0	0	laundry on site	Unknown	38.7596	-77.1485	dc
11	7050022780	washington DC	2155	apartment	589	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.9202	-77.0375	dc
12	7050022564	washington DC	1840	apartment	763	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.953	-77.2295	dc
13	7048624235	washington DC	1705	apartment	893	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.8458	-77.3242	dc
14	7050020926	washington DC	1421	apartment	773	1	1	1	0	1	0	0	0	laundry in bldg	Unknown	38.9719	-76.9554	dc
15	7047045467	washington DC	1741	apartment	1146	2	1	0	0	1	0	0	0	Unknown	Unknown	38.8307	-77.2142	dc
16	7050020222	washington DC	1200	apartment	680	1	1	0	0	0	0	0	0	no laundry on site	street parking	39.0005	-76.9723	dc
17	7049898058	washington DC	1479	apartment	950	2	1	1	1	1	0	0	0	laundry on site	off-street parking	38.7589	-77.0873	dc
18	7050015326	washington DC	1373	apartment	850	2	1.5	1	1	1	0	0	0	Unknown	Unknown	39.1821	-77.5359	dc
19	7050019172	washington DC	1957	apartment	1085	2	2	1	0	1	0	0	0	laundry in bldg	Unknown	38.9719	-76.9554	dc
20	7050019146	washington DC	2177	apartment	1118	2	2	1	1	0	1	0	0	w/d in unit	detached gara...	39.0003	-77.1022	dc
21	7050015767	washington DC	1299	apartment	730	1	1	1	1	1	0	0	0	Unknown	Unknown	39.1821	-77.5359	dc
22	7050016120	washington DC	1358	apartment	775	1	1	1	1	1	0	0	0	Unknown	Unknown	39.1821	-77.5359	dc
23	7050018288	washington DC	1587	apartment	640	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.8415	-77.0905	dc
24	7050018128	washington DC	1716	apartment	695	1	1	1	1	1	0	0	0	w/d in unit	Unknown	38.8935	-77.2532	dc
25	7050017507	washington DC	1850	apartment	487	0	1	1	1	1	0	0	0	w/d in unit	Unknown	38.9287	-77.0275	dc

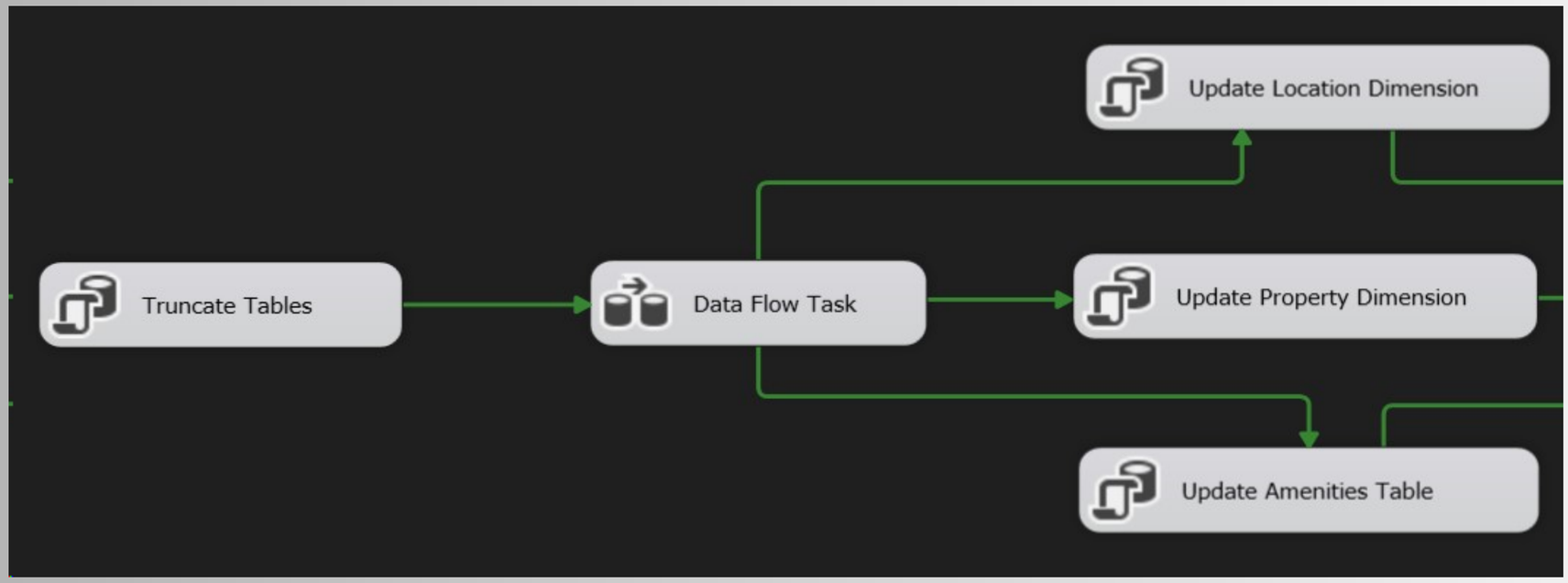
Data Warehousing (2)

The next step was to create an Execute SQL Task, connected to the Data Flow Task, and truncate the table before the data insertion in order to avoid stacking the data each time.



Data Warehousing (3)

Afterwards, we created our 3 dimension tables (Location, Property, Amenities) in SSMS. Then, we created 3 Execute SQL Tasks that update those tables by inserting the appropriate values. These tasks are connected to the Data Flow Task.



Dimensions

Location Dimension

Column Name	Data Type	Allow Nulls
location_id	int	☐
region	varchar(64)	☐
state	varchar(64)	☐
lat	float	☐
long	float	☐
		☐

Property Dimension

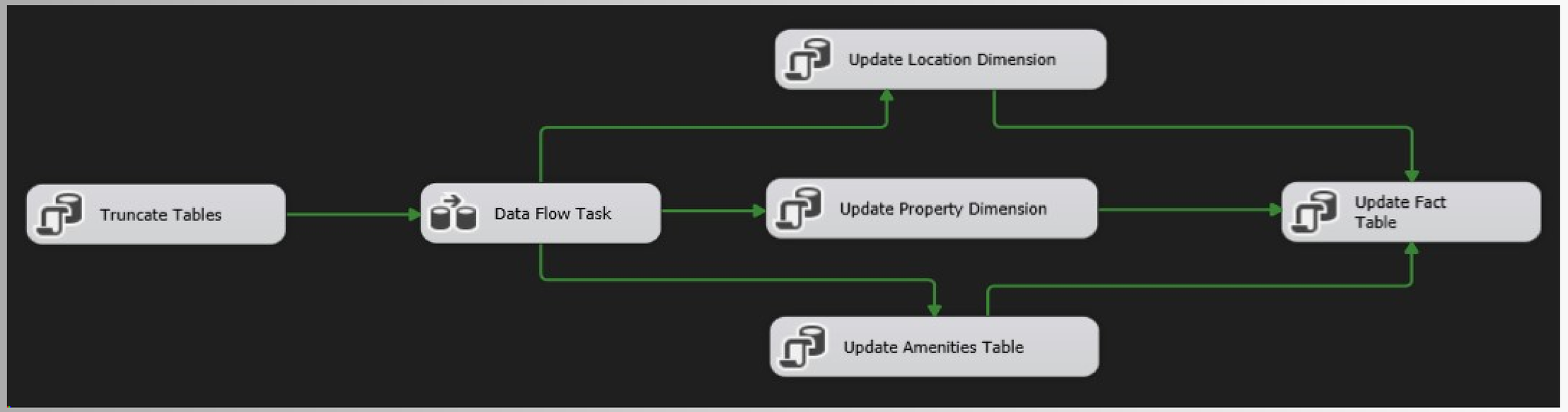
Column Name	Data Type	Allow Nulls
property_id	int	☐
type	varchar(64)	☐
beds	int	☐
baths	float	☐
		☐

Amenities Dimension

Column Name	Data Type	Allow Nulls
amenities_id	int	☐
cats_allowed	bit	☐
dogs_allowed	bit	☐
smoking_allowed	bit	☐
wheelchair_access	bit	☐
electric_vehicle_charge	bit	☐
comes_furnished	bit	☐
laundry_options	varchar(64)	☐
parking_options	varchar(64)	☐
		☐

Data Warehousing (4)

Afterwards, we created our Fact table in SSMS. Then, we created an Execute SQL Task that joins the dimension ids to the staging table and selects them into the Fact table. This task is connected to the Dimension Tasks.



Dimension Tables(1)

Property Dimension Table

	property_id	type	beds	baths
1	139	apartment	1	0
2	34	condo	1	0
3	80	cottage/cabin	1	0
4	102	duplex	1	0
5	19	house	1	0
6	114	in-law	1	0
7	105	loft	1	0
8	11	manufactured	1	0
9	61	townhouse	1	0
10	3	apartment	2	0
11	129	condo	2	0
12	24	cottage/cabin	2	0
13	146	duplex	2	0
14	179	flat	2	0
15	122	house	2	0
16	159	in-law	2	0
17	174	loft	2	0
18	110	manufactured	2	0
19	121	townhouse	2	0

Location Dimension Table

	location_id	region	state	lat	long
1	31255	hawaii	hi	19.1002	-155.726
2	238	hawaii	hi	19.1013	-155.769
3	12297	hawaii	hi	19.3085	-155.82
4	21066	hawaii	hi	19.3129	-155.887
5	44552	hawaii	hi	19.3206	-155.81
6	16662	hawaii	hi	19.3356	-155.028
7	48979	hawaii	hi	19.4273	-155.913
8	20408	hawaii	hi	19.4282	-155.021
9	45825	hawaii	hi	19.497	-155.912
10	12005	hawaii	hi	19.4975	-155.074
11	42240	hawaii	hi	19.5002	-155.101
12	3398	hawaii	hi	19.5295	-154.981
13	36012	hawaii	hi	19.5323	-154.914
14	60793	hawaii	hi	19.5345	-154.846
15	59569	hawaii	hi	19.5447	-155.924
16	61541	hawaii	hi	19.5506	-155.086
17	8070	hawaii	hi	19.5693	-155.035
18	30559	hawaii	hi	19.589	-154.949
19	20938	anchorage / mat-su	ak	19.5893	-154.993

Dimension Tables(2)

Amenities Dimension Table

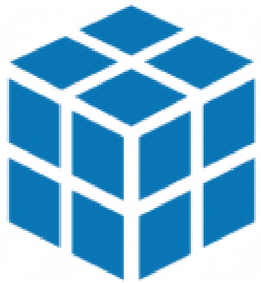
	amenities_id	cats_allowed	dogs_allowed	smoking_allowed	wheelchair_access	electric_vehicle_charge	comes_furnished	laundry_options	parking_options
1	284	0	0	0	0	0	0	laundry in bldg	attached garage
2	601	0	0	0	1	0	0	laundry in bldg	attached garage
3	1191	0	0	1	0	0	0	laundry in bldg	attached garage
4	1048	0	0	1	1	0	0	laundry in bldg	attached garage
5	280	0	0	0	0	0	0	laundry in bldg	carport
6	1121	0	0	0	1	0	0	laundry in bldg	carport
7	772	0	0	1	0	0	0	laundry in bldg	carport
8	210	0	0	1	1	0	0	laundry in bldg	carport
9	779	0	0	0	0	0	0	laundry in bldg	detached garage
10	298	0	0	0	1	0	0	laundry in bldg	detached garage
11	833	0	0	1	0	0	0	laundry in bldg	detached garage
12	908	0	0	1	1	0	0	laundry in bldg	detached garage
13	1	0	0	0	0	0	0	laundry in bldg	no parking
14	266	0	0	1	0	0	0	laundry in bldg	no parking
15	196	0	0	0	0	0	0	laundry in bldg	off-street parking
16	478	0	0	0	1	0	0	laundry in bldg	off-street parking
17	104	0	0	1	0	0	0	laundry in bldg	off-street parking
18	719	0	0	1	1	0	0	laundry in bldg	off-street parking
19	158	0	0	0	0	0	0	laundry in bldg	street parking

Fact Table

	listing_id	location	property	amenities	sqfeet	price
1	7003808130	42471	1	161	954	799
2	7004010416	58328	1	55	1117	1005
3	7004032234	30747	1	670	763	750
4	7004041631	57816	139	640	800	1000
5	7004048100	22021	16	902	641	530
6	7004059925	12370	17	157	1500	850
7	7004071335	57816	16	640	900	1000
8	7004183951	29315	156	246	1200	1500
9	7004199151	30747	27	670	440	545
10	7004529237	9450	171	902	815	570
11	7004546974	7456	48	902	900	550
12	7004554442	14536	16	996	635	600
13	7004558383	14536	26	873	858	690
14	7004561469	14536	172	873	958	740
15	7004567218	51839	1	902	1190	1279
16	7004570307	5327	26	72	1340	899
17	7004588837	39141	59	902	780	500
18	7004604504	40989	118	530	1540	995
19	7004638292	54545	107	1076	814	1310
20	7004645442	62150	48	902	700	620
21	7004655964	14536	16	873	635	600
22	7004686153	51839	16	6	786	1200
23	7004692207	63675	1	782	1232	1220

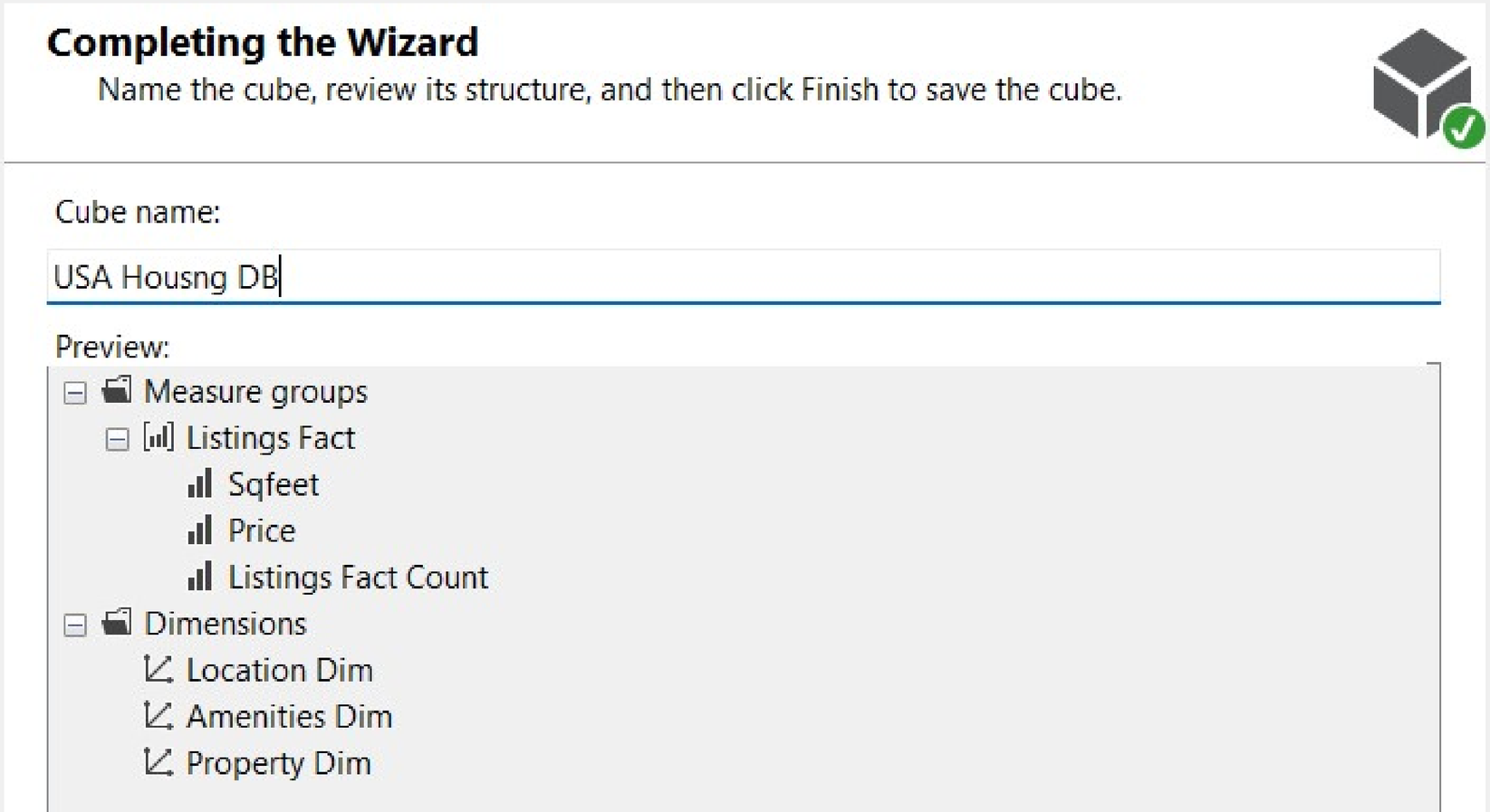
Cube

We used Microsoft Visual Studios Analysis Services Multidimensional Project to load our views from SQL Server, to create process and to deploy the cube of our data.

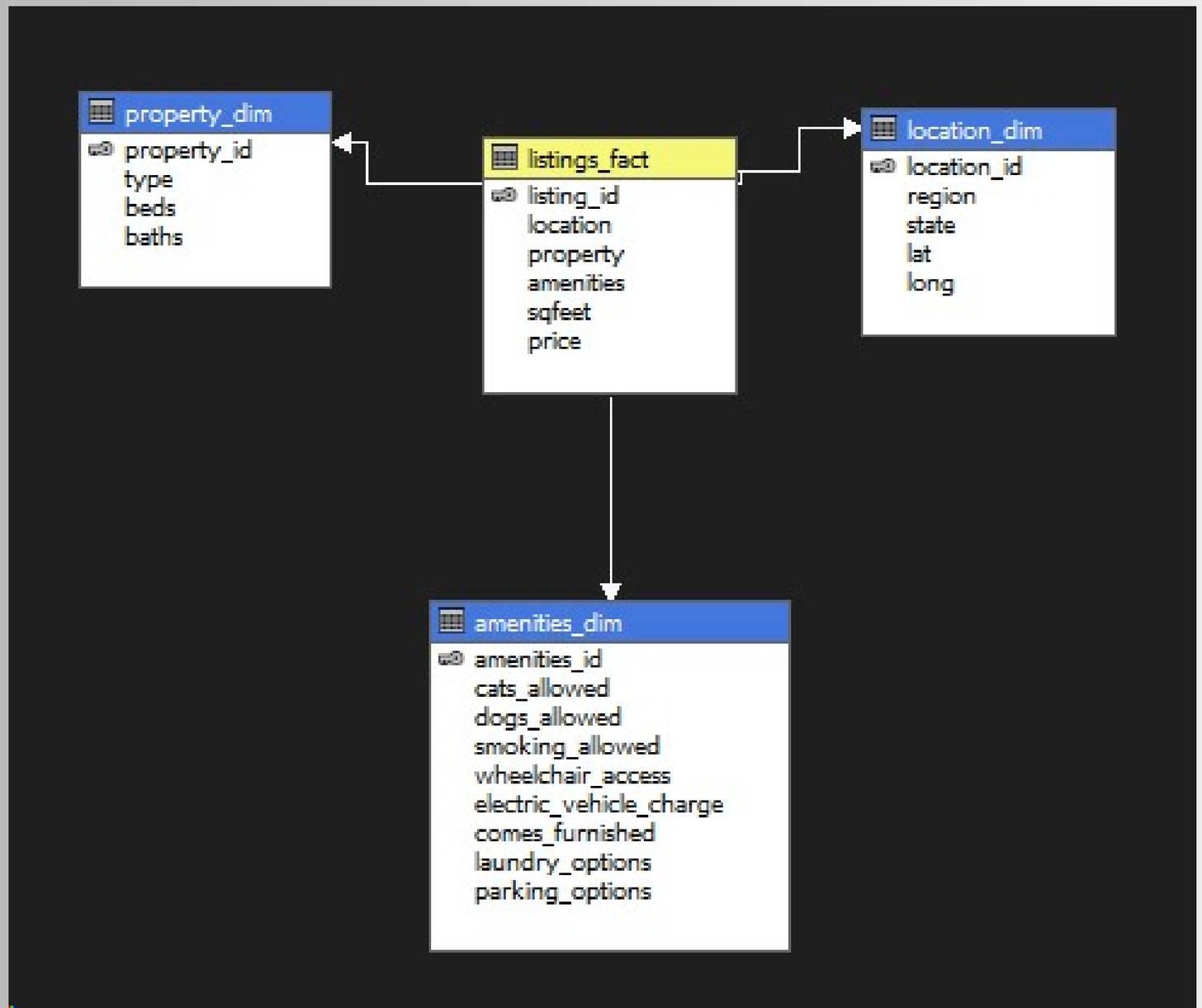


Cube(1)

We added our dimensions and Fact table to create the view from our SQL Server Data Source. We set as measures the Sqfeet, the Price and the Listings Fact Count.

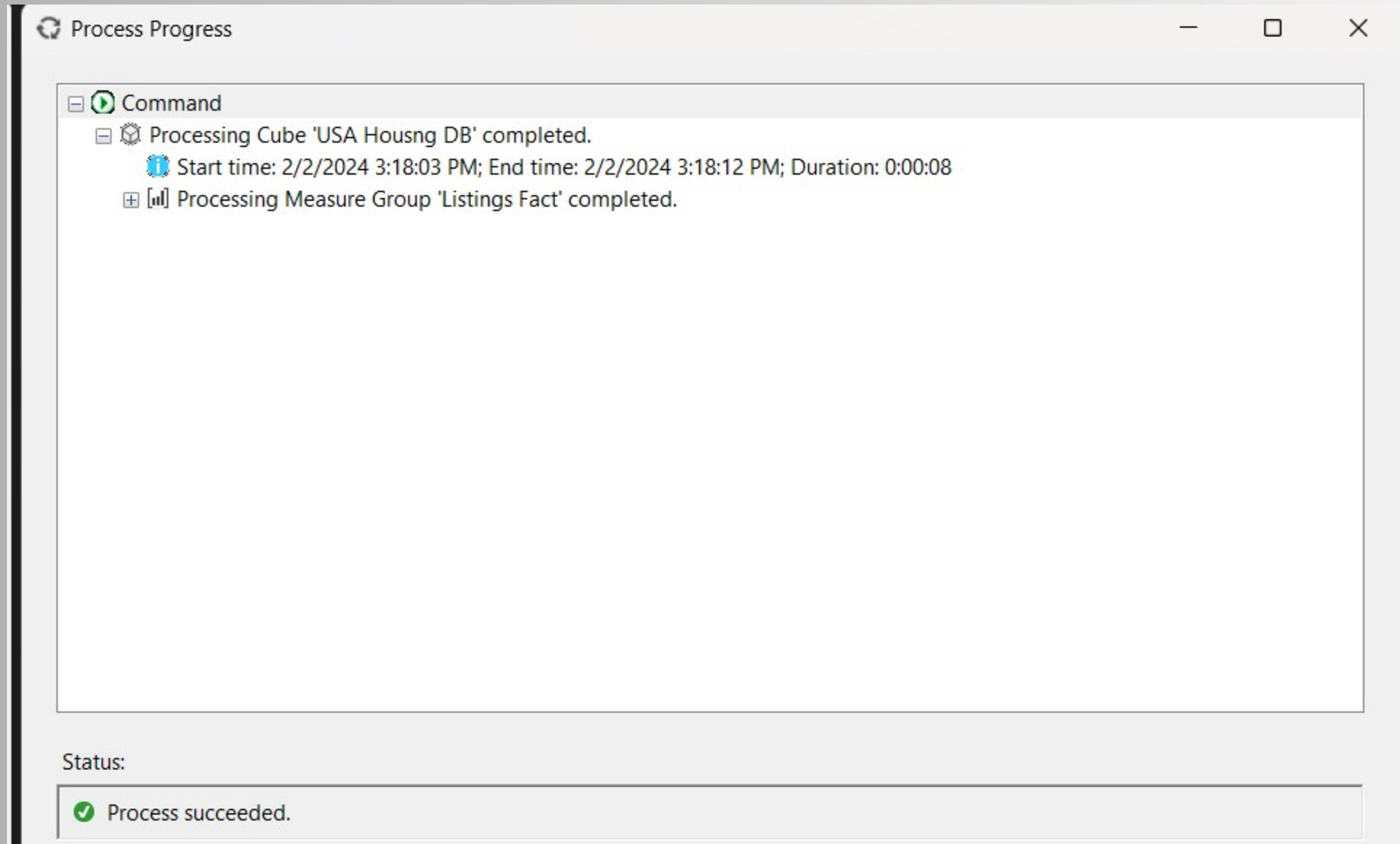


Star Schema


property_dim
property_id
type
beds
baths
listings_fact
listing_id
location
property
amenities
sqfeet
price
location_dim
location_id
region
state
lat
long
amenities_dim
amenities_id
cats_allowed
dogs_allowed
smoking_allowed
wheelchair_access
electric_vehicle_charge
comes_furnished
laundry_options
parking_options

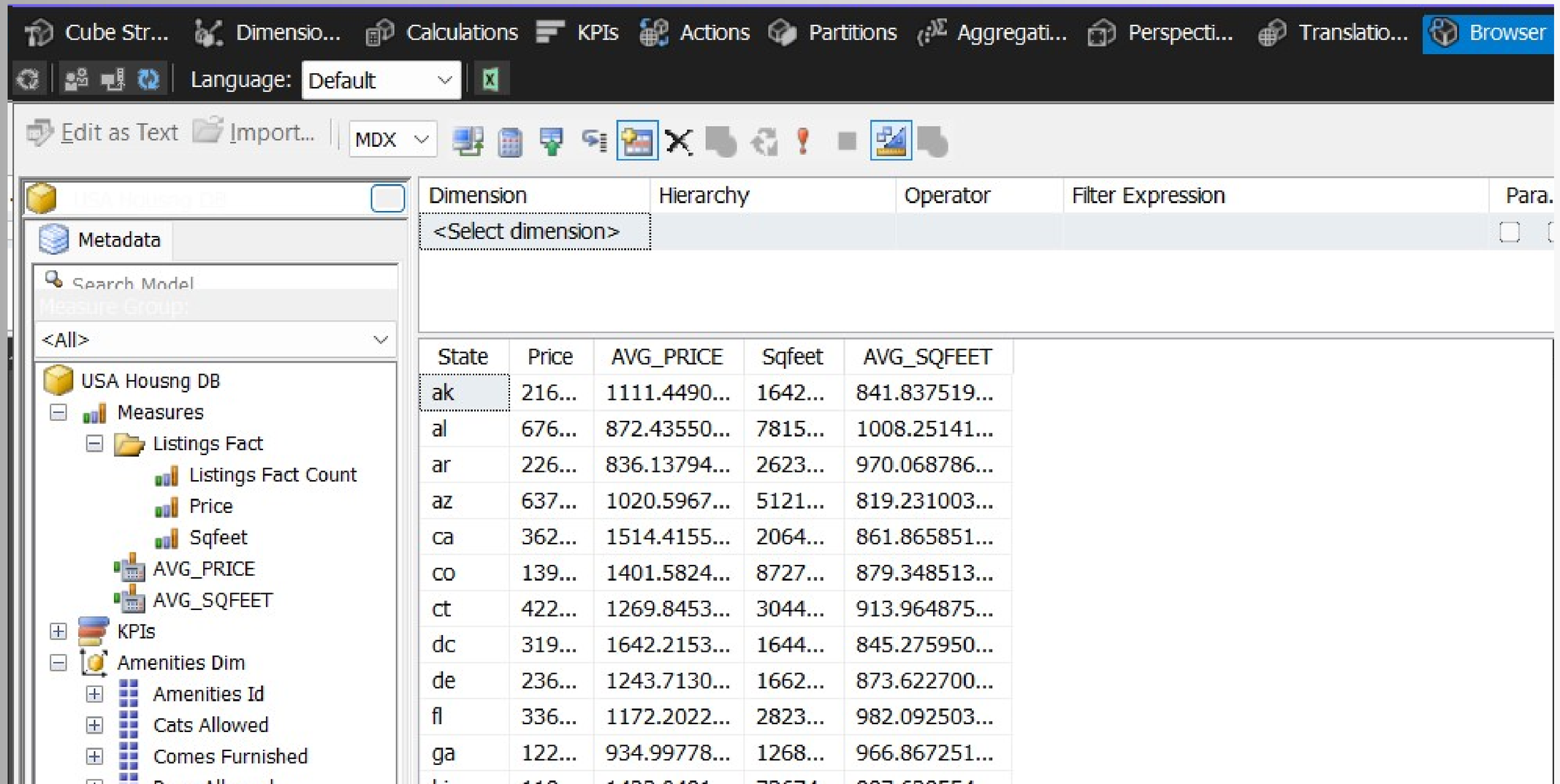
Cube(2)

We processed the cube and ran it succesfully .



Cube(3)

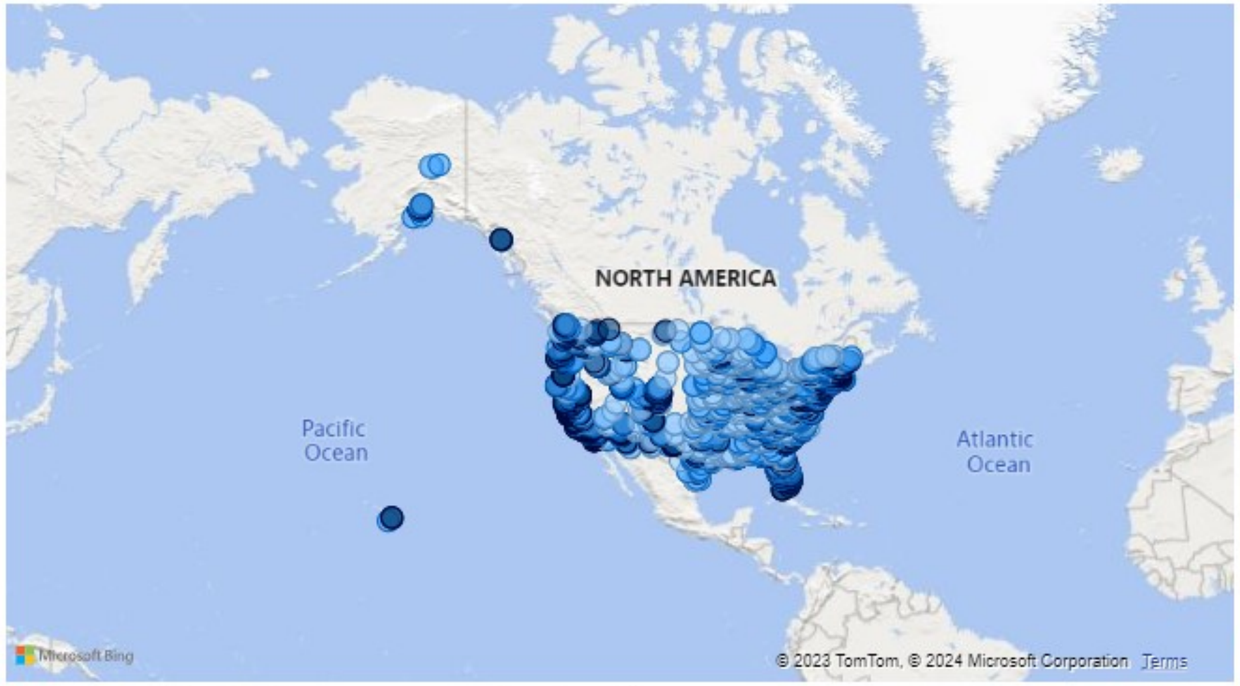
We can browse it to see it's working correctly.



Data Visualization

We connected our Analysis Services Project-our cube specifically- to PowerBI Desktop to create visualizations

Map of Listings



Total Listings

340.83K

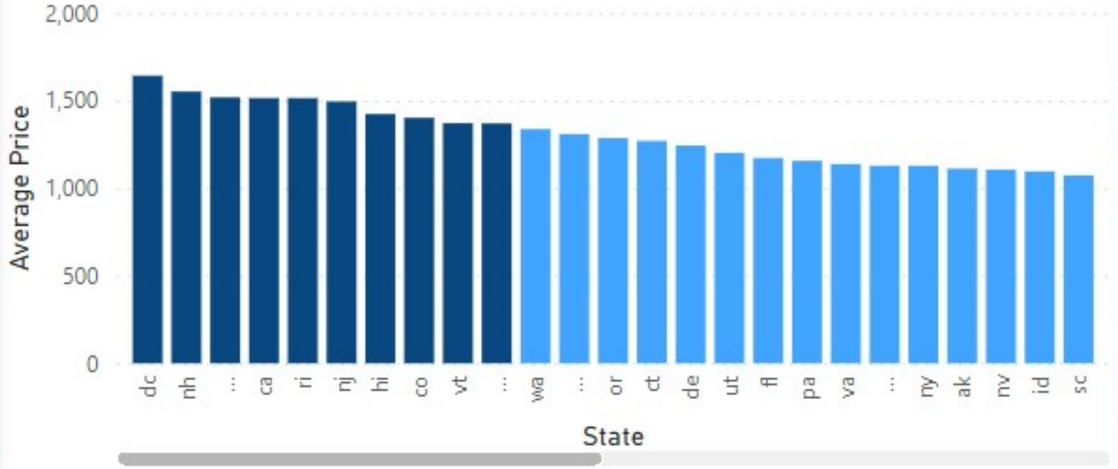
Average Price ($)

1,083

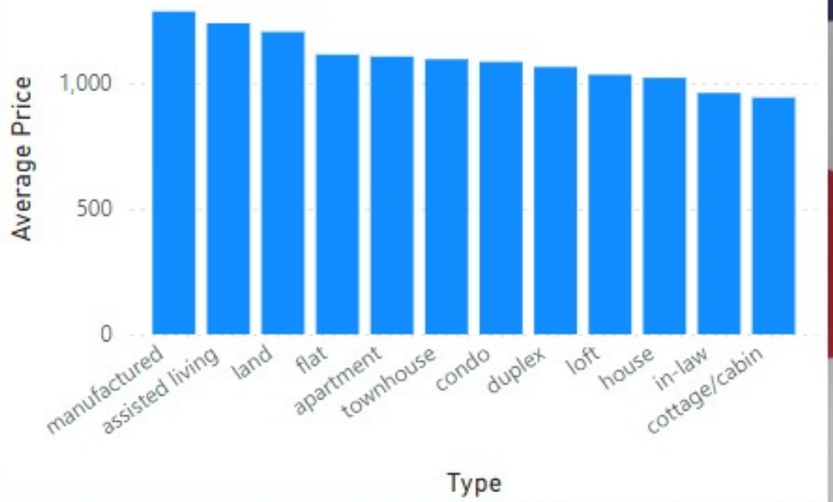
Average Sqfeet

926

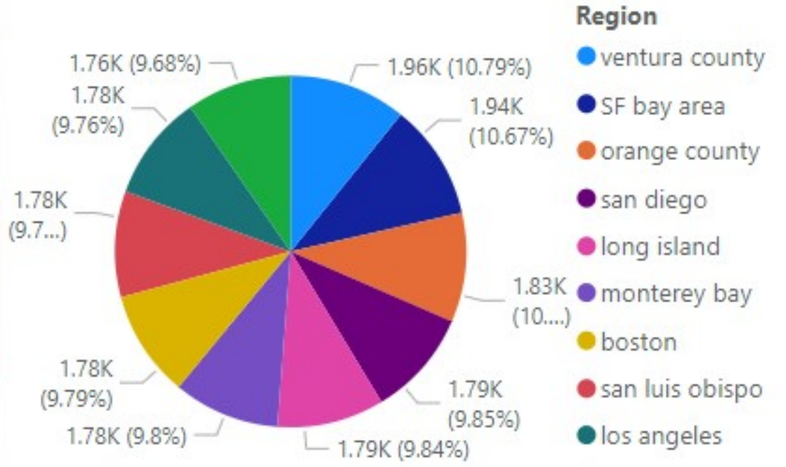
Average Price per State



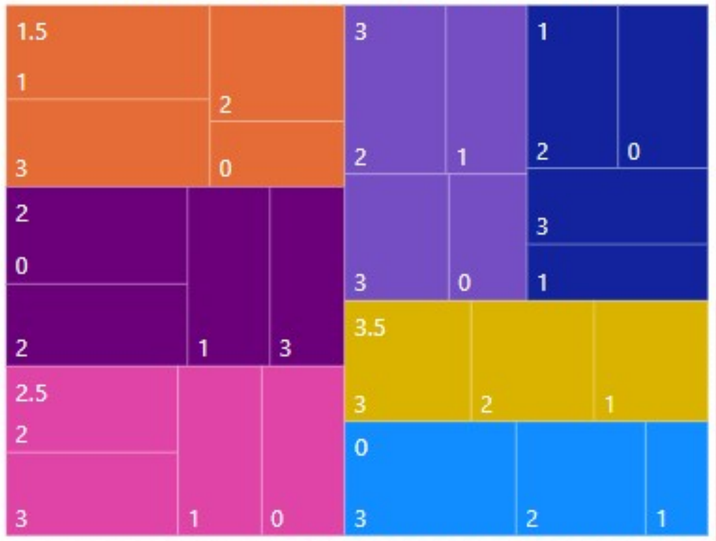
Average Price per Type



10 Most Expensive Regions

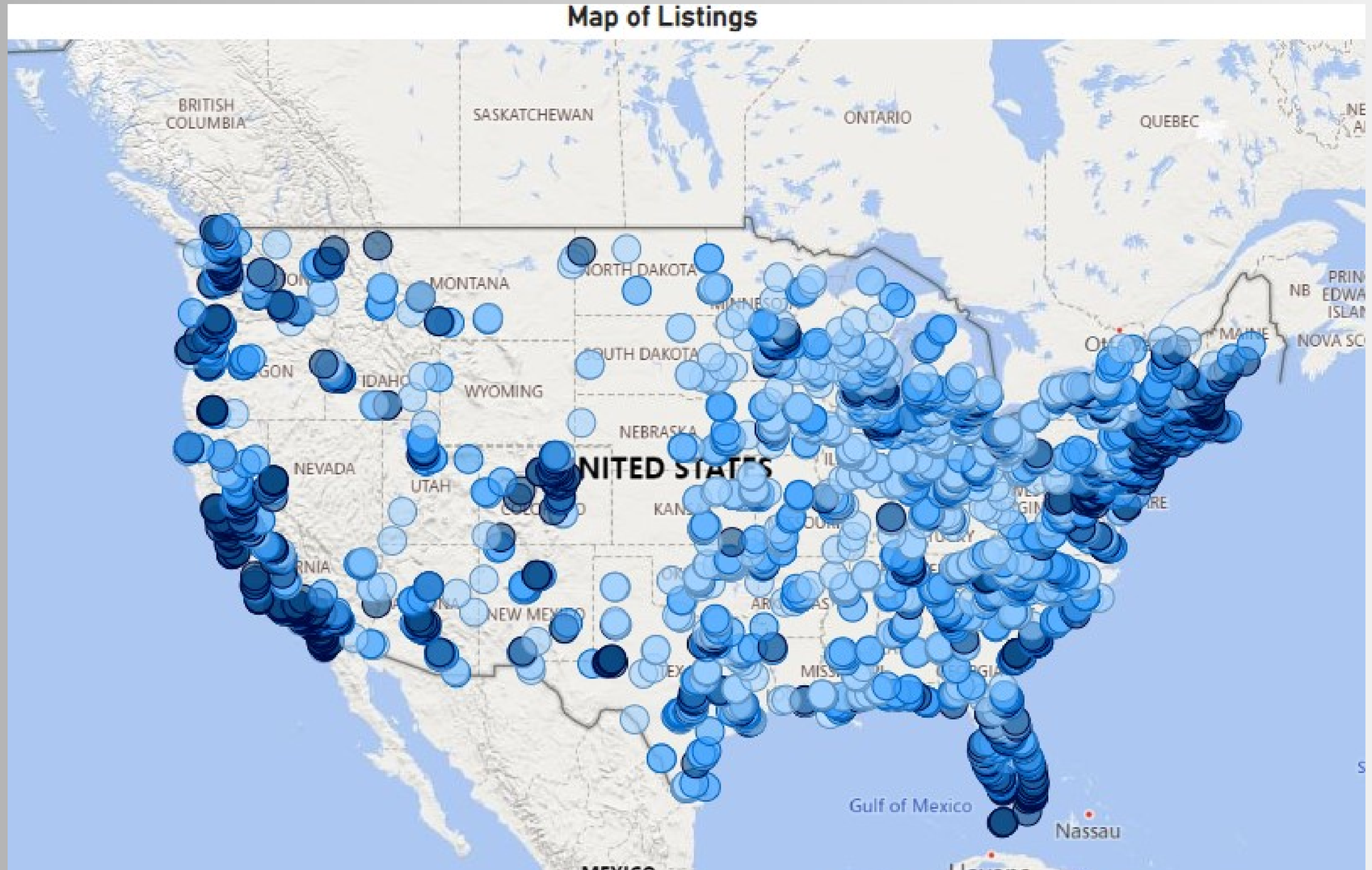


Beds & Baths Distribution by Sqfeet



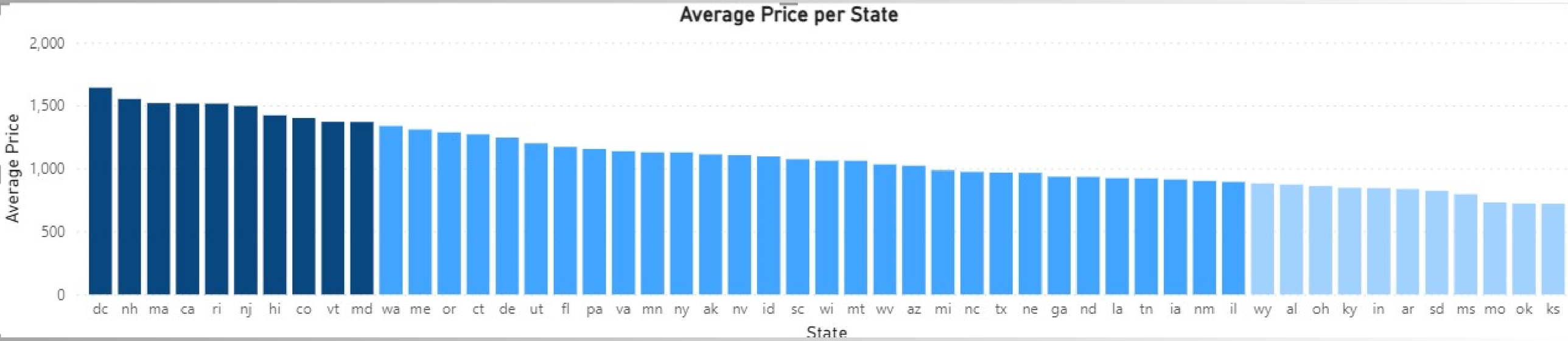
Visualization(1)

Firstly, we created a Map that shows all the listings in America



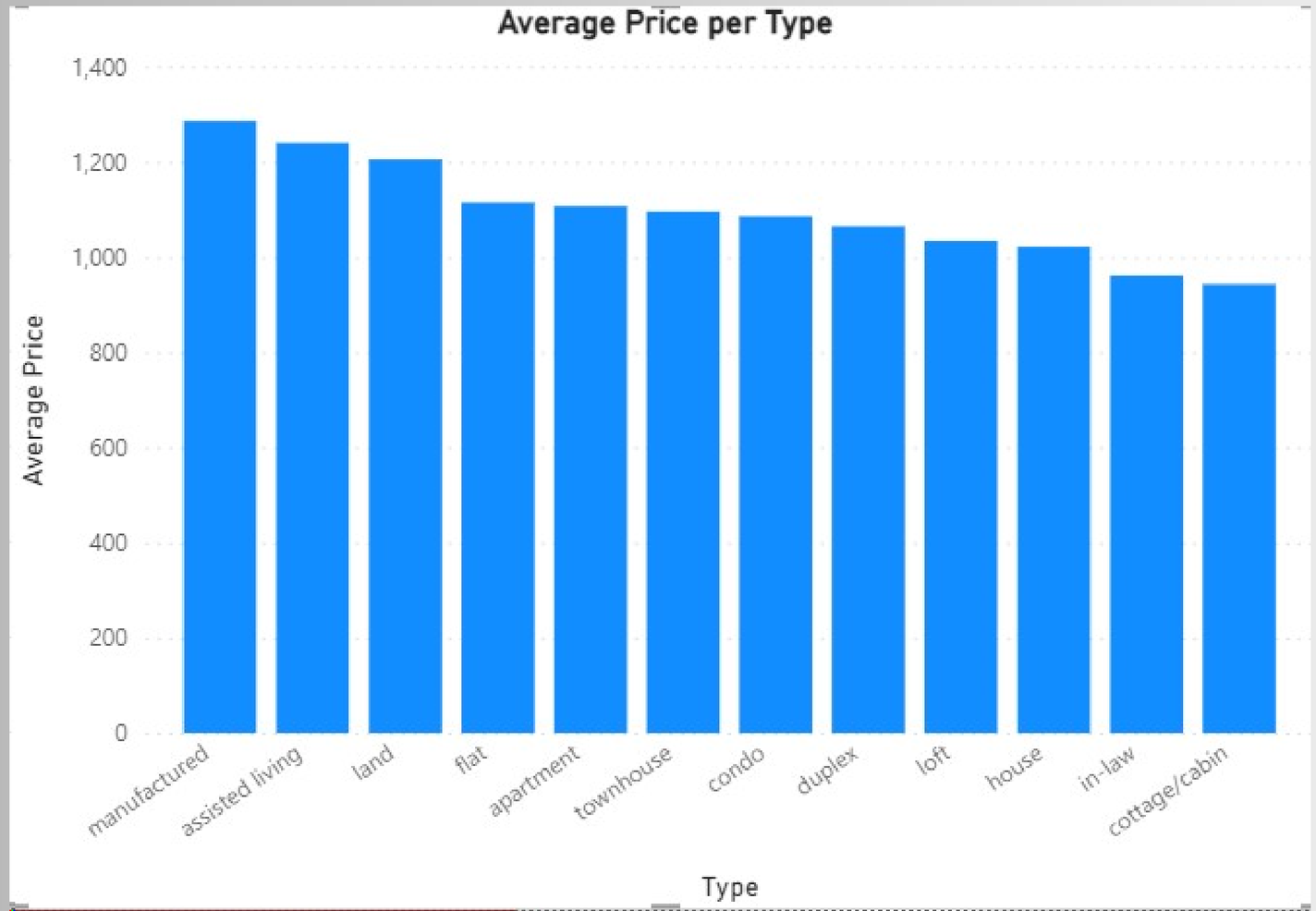
Visualization(2)

Also, we created a column chart that visualizes the average price per state



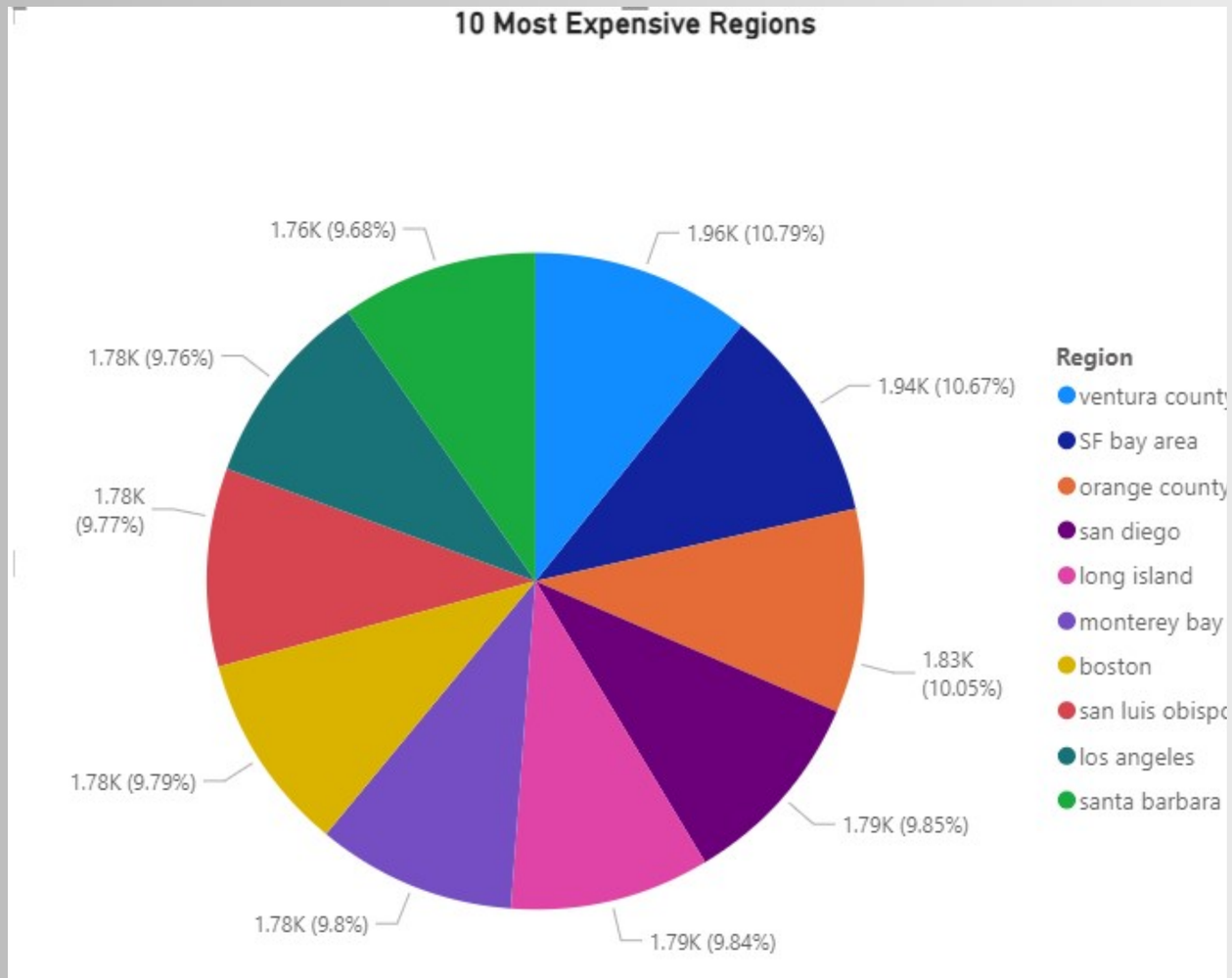
Visualization(3)

Afterwards, we created another column chart showcasing the average price per type of the listing.



Visualization(4)

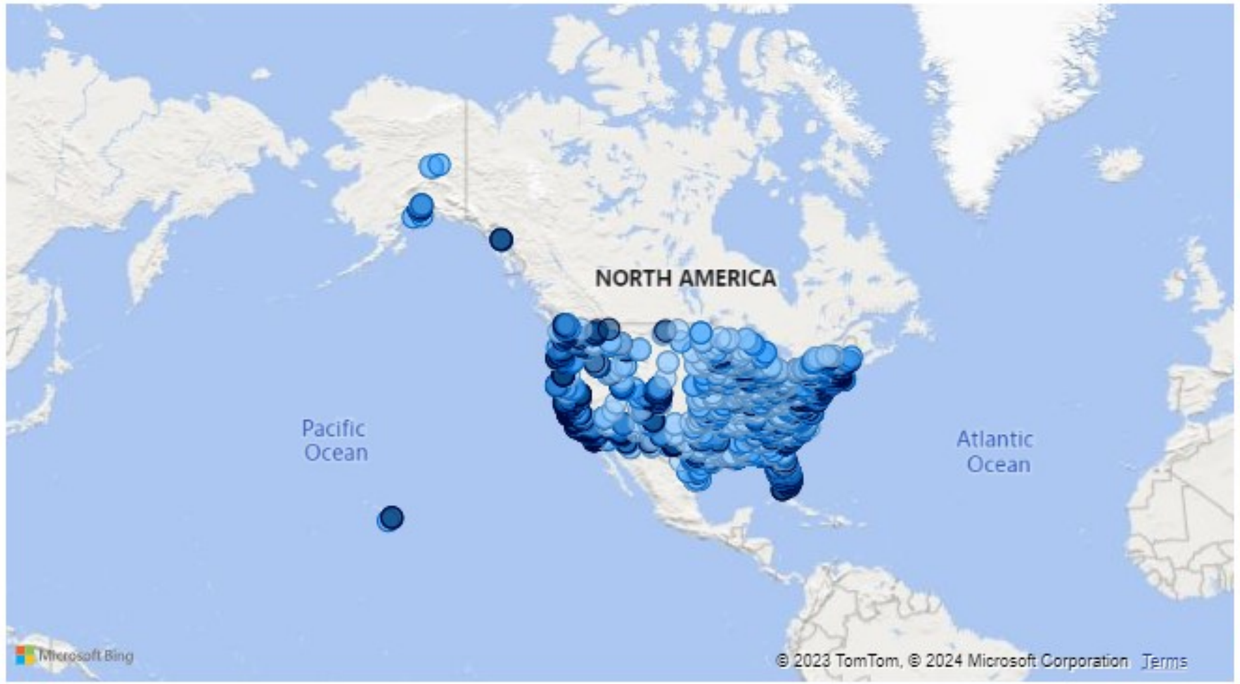
Then, we added a pie chart showing the 10 most expensive regions, based on the average price.



Visualization(5)

Firstly, we created a Treemap that shows the bed and bath distribution by Sqfeet.

Map of Listings



Total Listings

340.83K

Average Price ($)

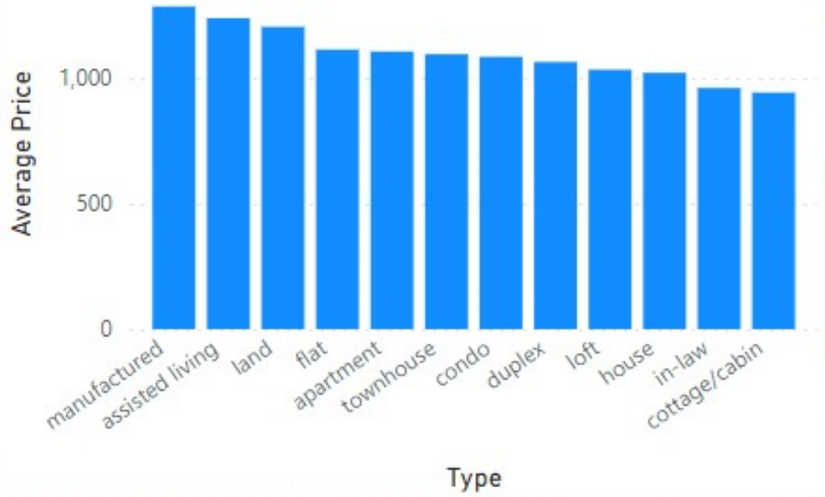
1,083

Average Sqfeet

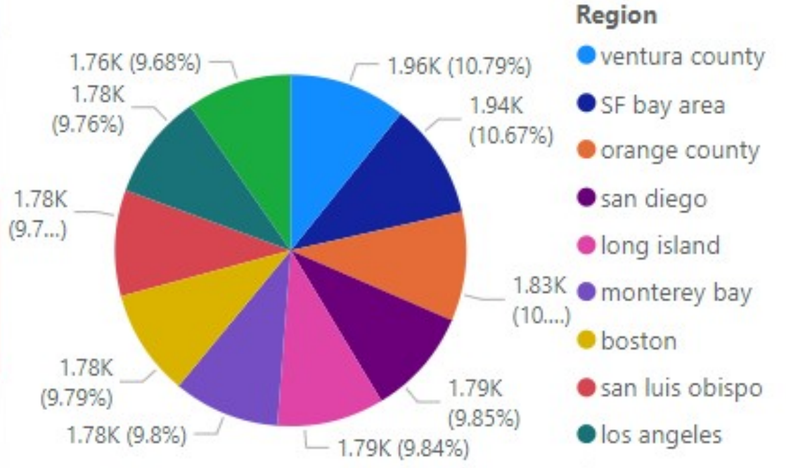
926

Average Price per State

Average Price per Type

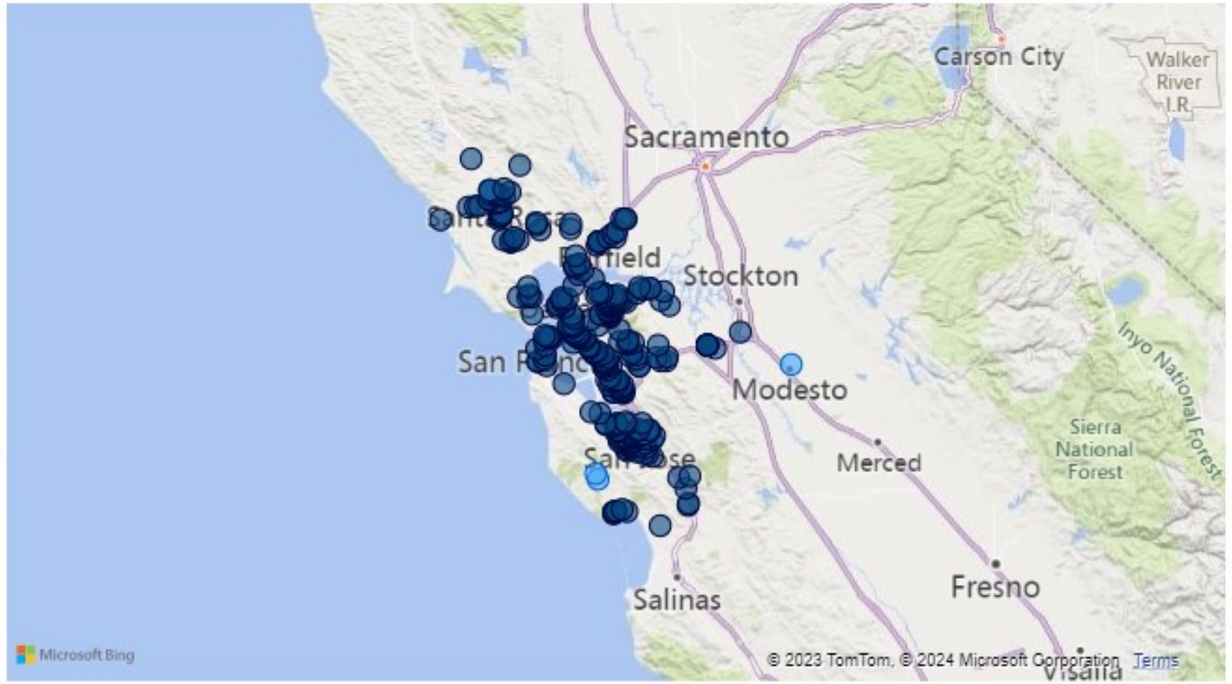


10 Most Expensive Regions



Beds & Baths Distribution by Sqfeet

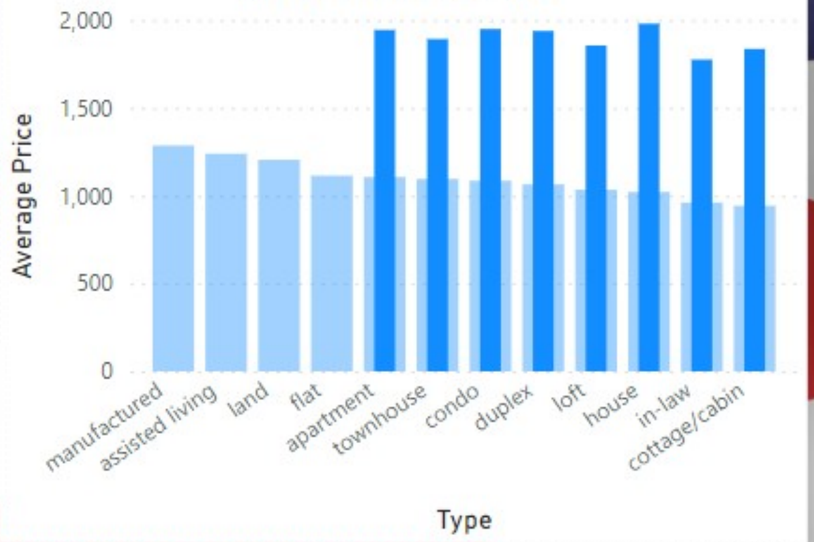
Map of Listings



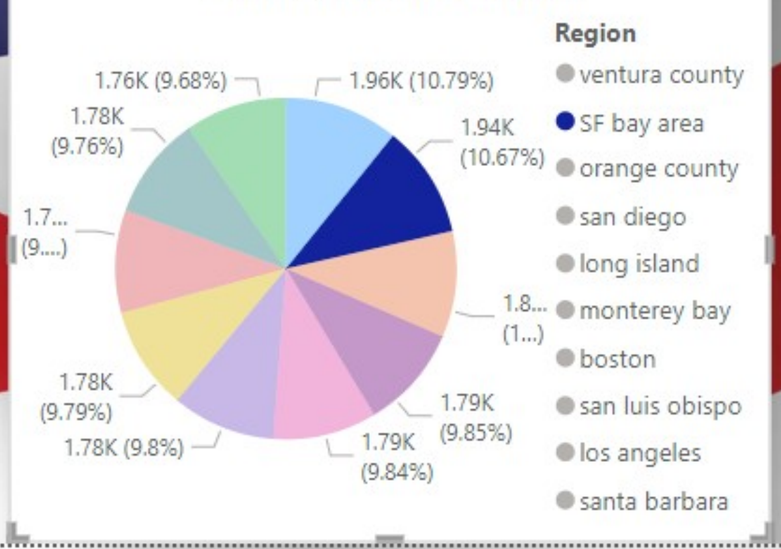
Total Listings	Average Price ($)	Average Sqfeet
580	1,941	697

Average Price per State

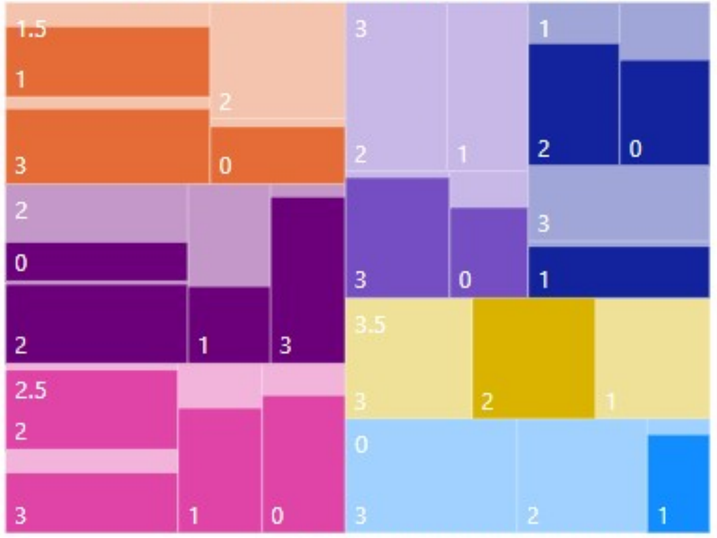
Average Price per Type



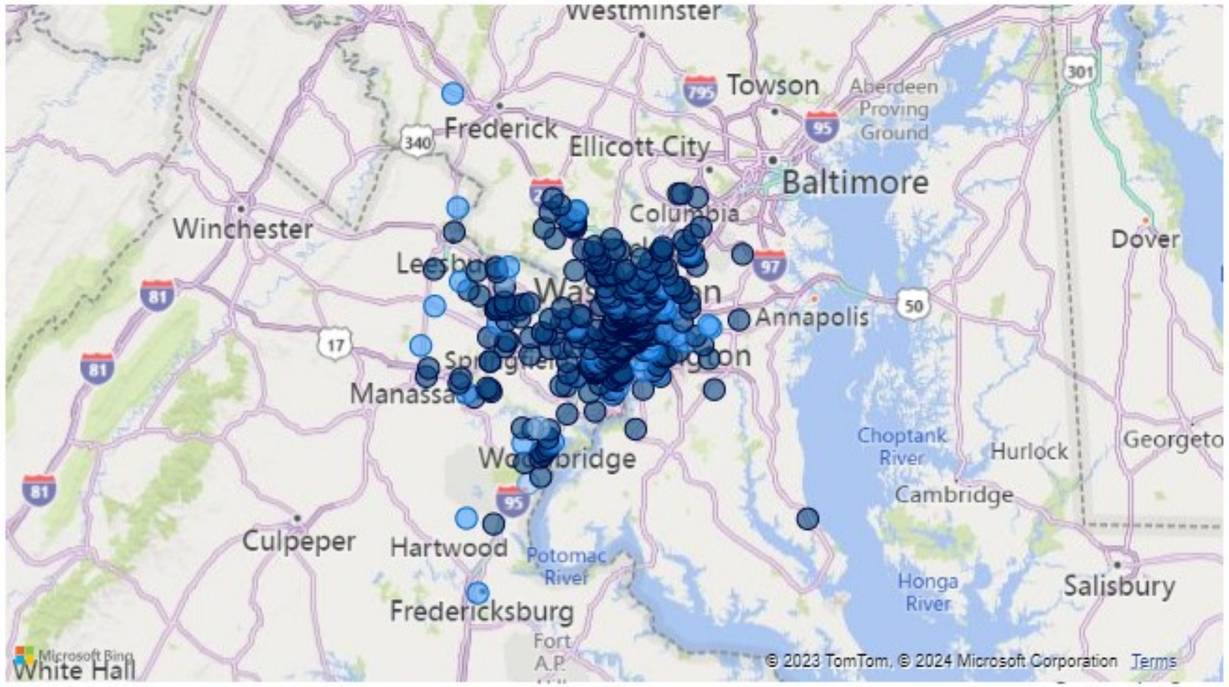
10 Most Expensive Regions



Beds & Baths Distribution by Sqfeet



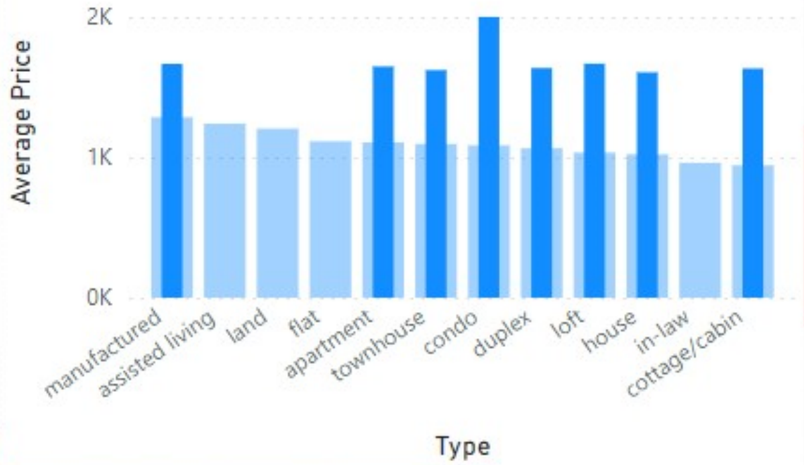
Map of Listings



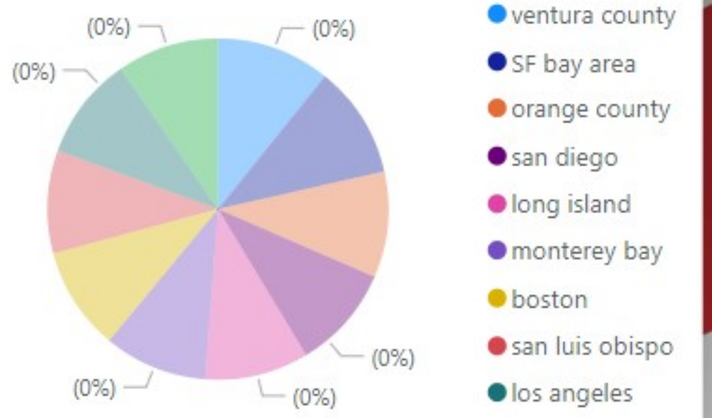
Total Listings	Average Price ($)	Average Sqfeet
1946	1,642	845

Average Price per State

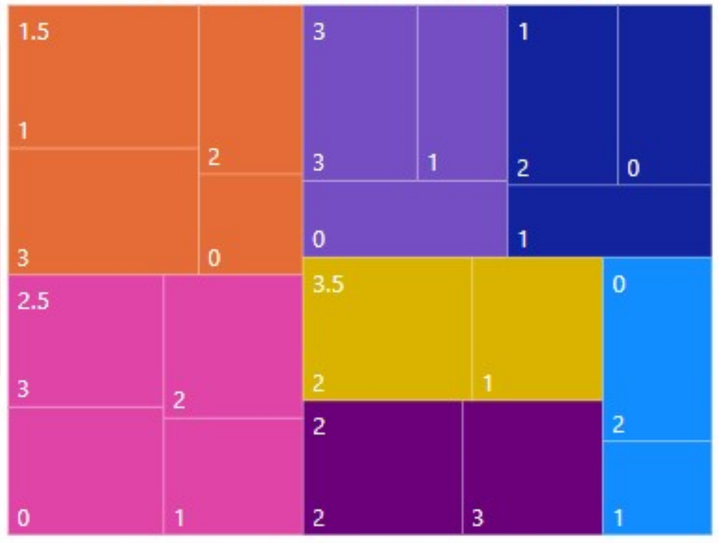
Average Price per Type



10 Most Expensive Regions



Beds & Baths Distribution by Sqfeet



Data Mining

The open-source tools we used for the process of data mining are the Jupyter Notebook, with the application of many Python Libraries, and RapidMiner.

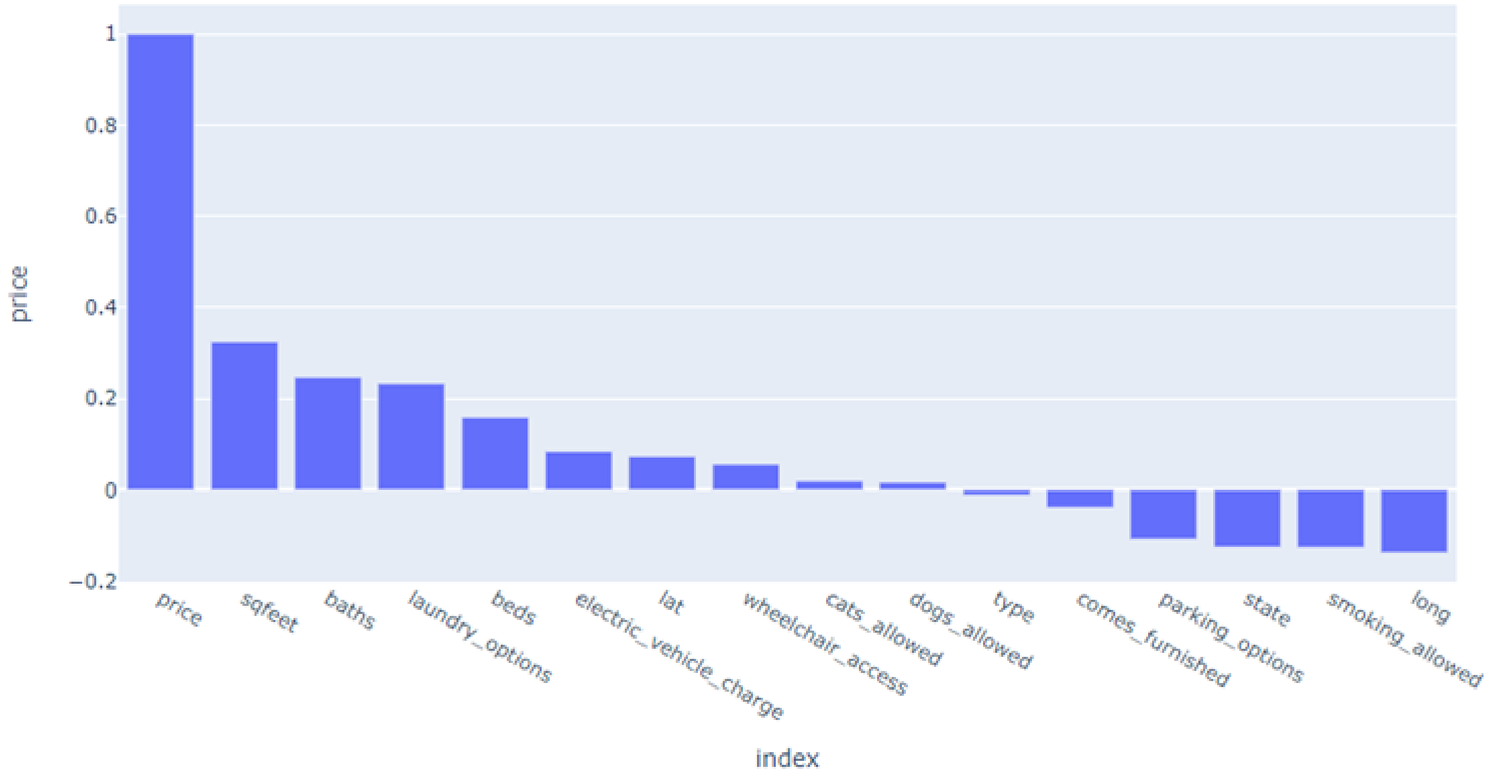
Data Mining(1)

For the first data mining task, we decided to focus on various models of regression, creating predictive models, and exploring the relationships between the price of a listing and its amenities.

Data Mining(2)

This bar chart depicting the correlation between features and the variable price.

```
In [5]:
import plotly.express as px
correlation=df.corr()["price"].reset_index().sort_values("price",ascending=False)
fig=px.bar(correlation, x="index", y="price")
fig.show()
```



Data Mining(3)

After examining the previous chart, we desided to classify the features based on their correlation to the price. So, we have the following groups:

Strong correlation: This category includes the square footage (sqfeet) of the building, the number of baths, and the availability of washer options, where the correlation value is greater than 0.2 (correlation > 0.2).
Moderate correlation: This category includes the number of bedrooms, the geographic latitude, the provision of electric vehicle charger, accessibility for people with disabilities, as well as the pet policy. The correlation value is greater than 0 and less than 0.2 (0 < correlation < 0.2).
Negative correlation: This category encompasses the type of listing, the furnishing, parking availability, the state, smoking policy, and the geographic longitude. The correlation value seems to have values less than 0 and greater than -0.2 (-0.2 < correlation < 0).

Data Mining(4)

We desided to try 4 different regression models:

- Linear Regrssion Model
- XGBoost (Extreme Gradient Boosting)
- Gradient Boosting Regressor
- RandomForestRegressor

In [21]:
```
models={"LR":[lr_r2, lr_MSE , lr_RMSE],
        "XGB":[xgb_r2, xgb_MSE, xgb_RMSE],
        "GBR":[gbr_r2, gbr_MSE , gbr_RMSE],
        "RAN":[ran_r2, ran_MSE , ran_RMSE],}
models=pd.DataFrame(models)
models=models.rename(index={0:"R^2", 1:"MSE", 2:"RMSE"})
models
```

Out[21]:

	LR	XGB	GBR	RAN
R^2	0.236733	0.813321	0.619766	0.902279
MSE	110576.509409	27044.711885	55085.563050	14157.106064
RMSE	332.530464	164.452765	234.703138	118.983638

In [24]:

```
mean_predicted_values = pred_graph_ran.groupby("State").mean()
mean_predicted_values["Percentage Difference"] = ((mean_predicted_values["Predicted Values"] - mean_predicted_values["True Va
mean_predicted_values = mean_predicted_values.sort_values(by='Percentage Difference', ascending=False)
mean_predicted_values
```

Out[24]:

State	True Values	Predicted Values	Percentage Difference
mo	714.893443	752.474338	5.256853
wv	1030.940594	1066.885376	3.486601
wy	882.631579	904.886051	2.521377
nh	1529.367257	1548.574080	1.255867
oh	857.030864	865.527325	0.991383
ma	1506.730056	1520.239671	0.896618
nd	920.321897	928.144969	0.850036
ia	909.406689	916.434176	0.772755
ar	832.871046	839.225442	0.762951
pa	1153.967693	1161.781681	0.677141

Data Mining(5)

Focusing on the first three and the last three examples, we observe the following:

1. Missouri (MO): The predicted price is 5.26% higher than the actual price.
2. West Virginia (WV): The predicted price is 3.49% higher than the actual price.
3. Wyoming (WY): The predicted price is 2.52% higher than the actual price.

All three examples show a higher prediction than the actual price, which could be significant for an investor or homeowner as it may influence their decisions regarding renting out their property or purchasing it.

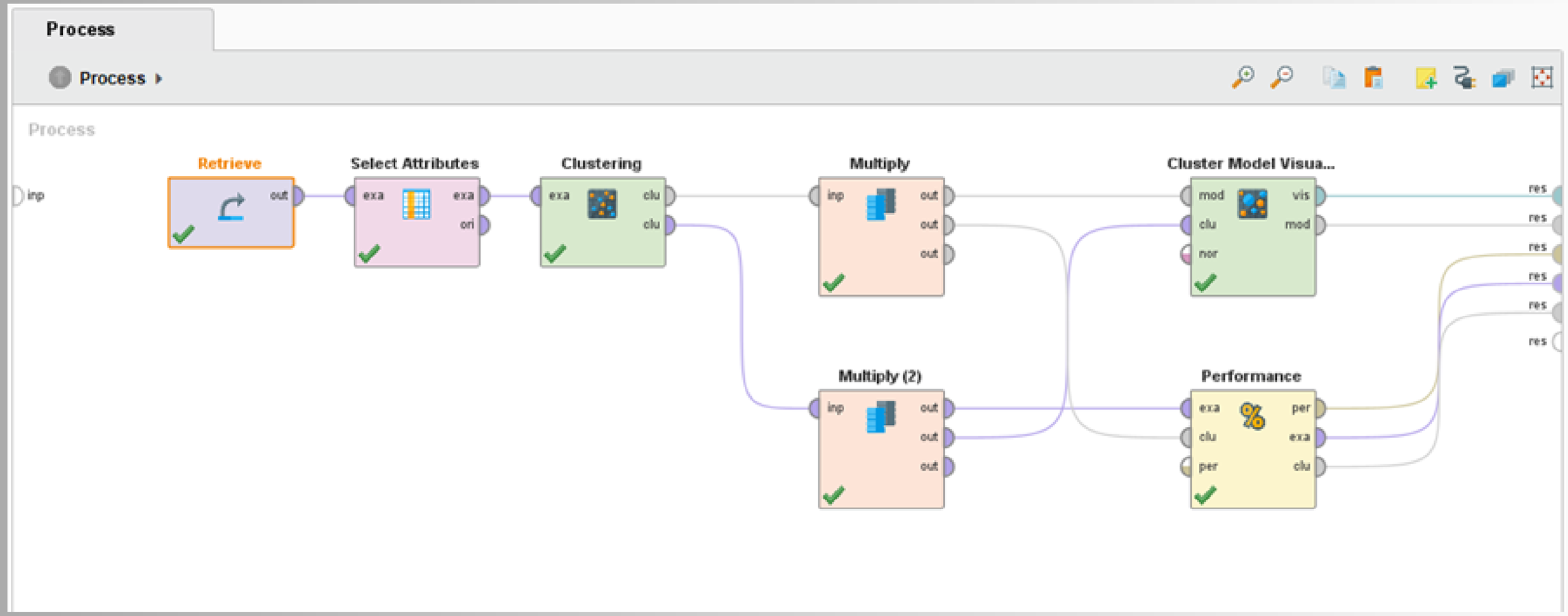
Data Mining(6)

1. Nebraska (NE): The predicted price is 0.36% lower than the actual price.
2. Wisconsin (WI): The predicted price is 0.41% lower than the actual price.
3. Alaska (AK): The predicted price is 0.71% lower than the actual price.

The last three examples show a slight underestimation by the model. Therefore, an investor or homeowner could take this information into account when determining the price they are willing to pay for acquiring a property or accepting an offer for a property they own in these specific states.

Data Mining(7)

For the clustering method, we chose the open-source tool RapidMiner and created the following process



Data Mining(8)

Results of the process above.

Cluster	cats_allowed	dogs_allowed	wheelchair_access	electric_vehicle_charge	comes_furnished	laundry_options	parking_options
Cluster 0	0.016	0.008	0.001	0.001	0.004	0	0.063
Cluster 1	0.986	1	0	0.008	0.019	1	1
Cluster 2	0.179	0.032	0.039	0.004	0.042	1	0
Cluster 3	0.999	1	0.062	0.004	0.015	1	0
Cluster 4	0.999	1.000	0.043	0.006	0.029	0	0
Cluster 5	0.005	0.096	0	0.007	1	0.891	0.996
Cluster 6	0.988	1.000	1	0.082	0.198	0.993	0.933
Cluster 7	1	0	0.049	0.005	0.018	0.961	1
Cluster 8	0.013	0.039	1	0.017	0.321	0.960	0.981
Cluster 9	0	0	0	0.004	0	1	1
Cluster 10	0.925	1	0.025	0.002	0.061	0	1

Data Mining(9)

Cluster Model

```
Cluster 0: 43122 items
Cluster 1: 138990 items
Cluster 2: 7343 items
Cluster 3: 56759 items
Cluster 4: 24856 items
Cluster 5: 3790 items
Cluster 6: 19375 items
Cluster 7: 9193 items
Cluster 8: 1557 items
Cluster 9: 31399 items
Cluster 10: 4444 items
Total number of items: 340828
```

Data Mining(10)

1. Barren Listings (Cluster 0): This cluster includes listings that do not offer any amenities. The percentage of these listings is 12.5%.
2. Pet-Friendly Haven with Laundry & Parking Amenities (Cluster 1): This cluster comprises all spaces that provide pet-friendly accommodations along with laundry facilities and parking. The occurrence rate is 40.5%.
3. Pet-Friendly Nest with Laundry Option & NO Parking (Cluster 3): This category is identical to the previous one, except it does not include parking spaces. The occurrence rate is 16.5%.
4. Pet Heaven (Cluster 4): This cluster encompasses listings that allow both cats and dogs, with a 100% occurrence rate. It constitutes 7% of the total.
5. Convenient Laundry & Parking Hub (Cluster 9): This cluster includes amenities such as laundry facilities and parking spaces. The occurrence rate is 9.1%.

Ευχαριστούμε πολύ!