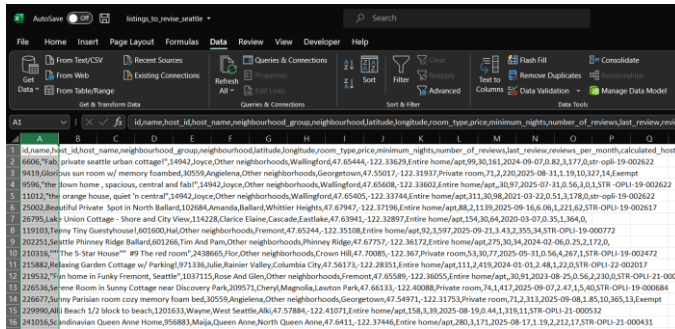


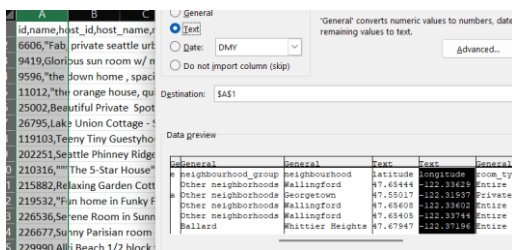
TABLEAU PROJECT

Airbnb Analysis

1. First step is cleaning data in Excel to prepare it for the visualization in Tableau.



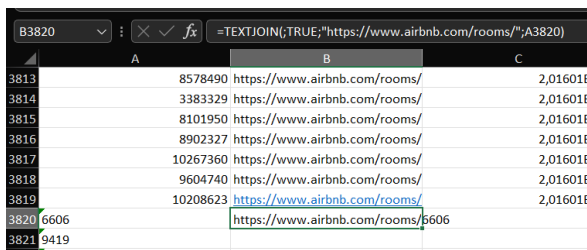
- 1.1. It's also important to check the **data types** before proceeding with the Data transferring to **columns**, because some numerical ones can then look not as expected and we'll have much more additional work to do when updating it.



- 1.2. And to pay attention to the data after transferring.

593	10245 / 1109682152098	1 line patio room lumin to
594	1024716578885966830	2BR 4 Beds
595	Convention Ctr	
596	Downtown Ctr	
597	Cruise Port"	
598	1024719315299934834	NEW! Chic+Fun Family Re
599	1024740303676802773	DT Seattle Red Door: 4brs

- 1.3. Additionally, our main file (to which I want to add the updated data from few last years) contains url address for the pages. I can add it using TEXTJOIN here.



- 1.4. For some names we have weird signs in there. Needs to be cleared as well. Used FIND & REPLACE to do so. Looks way better now.

	name	
722270	Historic Loft Suite @ Downtown Albany	22270 Historic Loft Suite @ Downtown Albany
668857	Cozy Garden Suite Full Amenities @Downtown Albany	58857 Cozy Garden Suite Full Amenities @Downtown Albany
993402	Historic Full Amenities 2BR Apt @Downtown Albany	33402 Historic Full Amenities 2BR Apt @Downtown Albany
330970	Muhammed Ali Lounge & Pool Hall b-yb-y	30970 Muhammed Ali Lounge & Pool Hall
528939	UNIQUE Gem in HEART of Albany B... Spiral Staircase	28939 UNIQUE Gem in HEART of Albany Spiral Staircase
529316	WOW B... Bright Apt B... PRIME downtown Walk Everywhere	29316 WOW Bright Apt PRIME downtown Walk Everywhere
529447	B... 2 BEDROOM / 1 BATH TRENDY Apt B... Downtown Albany	29447 2 BEDROOM / 1 BATH TRENDY Apt Downtown Albany
207226	RARE 1BR Apt with Character GREAT Location! W/Da...	72226 RARE 1BR Apt with Character GREAT Location! W/D
890336	CHIC, Renovated Brownstones... in PRIME DT Albany	30336 CHIC, Renovated Brownstone in PRIME DT Albany
579900	MODERN 1BR GREAT Location Downtown! W/D Parking!	79900 MODERN 1BR GREAT Location Downtown! W/D Parking!
		37196 MODERN 1BR Apt w/ PRIVATE deck! W/D

1.5. Using the same functionality here for the host id url.

=TEXTJOIN(TRUE;"https://www.airbnb.com/users/show/";G3820)		
G	H	
ac/pict	14703116	https://www.airbnb.com/users/show/14703116 Gil
14942		https://www.airbnb.com/users/show/14942 Joyce
30559		Angielena

1.6. Also needed to replace longitude & latitude values "." with "," to match.

H	I	J
longitude	room_type	
-73.7537	Entire home/apt	-122.3710252
-73.76724	Entire home/apt	-122.3656665
-73.75966	Entire home/apt	-122.3694832
-73.76724	Entire home/apt	-122.3692791
-73.76506	Entire home/apt	-122.3724706
-73.75851	Private room	-122.3661741
-73.80553	Private room	-122.3685191
-73.82438	Entire home/apt	-122.375856
-73.78297	Private room	-122.3572161
-73.76867	Private room	-122.3668886
-73.82763	Entire home/apt	-122.3584778

1.7. Location description that needs to be cleaned as well.

- ☒ (Select All)
- ☒ Ballard, Seattle, WA
- ☒             , WA
- ☒ Phinney Ridge Seattle, WA
- ☒ Seattle , WA
- ☒ Seattle, WA
- ☒ West Seattle, WA
- ☒ (Blanks)

1.8. After I combined the new, fresh UP to date file with listings from AIRBNB, I went checking on the base file and saw, that data still has some columns that were not present in downloaded listing. I decided to add missing addresses. As I'm just working on Seattle and mainly need to visualize the data I possess (after I clean it), it doesn't have to be exactly accurate. So, why not to use Python here, I thought.

E2	A	B	C	D	E	F
1	House Number	Street Name	Street Type	Zip Code		
2	563	Cheasty Boulevard South	Pl	98105	563, Cheasty Boulevard	
3	524	Post Alley	Ter	98117	524, Post Alley, 98117	
4	864	Yesler Way	Blvd	98126	864, Yesler Way, 98126	
5	245	Lake Washington Boulevard	St	98188	245, Lake Washington	
6	573	Denny Way	Dr	98188	573, Denny Way, 98188	

1.8.1. Since we already have all necessary libraries in, and just common known data from available sources, here how the final code looks like.

```
cwiczenia > python > random_streets.py ...
1 import random
2 import csv
3
4 street_names = ['1st Avenue (Seattle)', 'Alaskan Way', 'The Ave', 'Broadway (Seattle)', 'Cant
5 street_types = ['St', 'Ave', 'Blvd', 'Way', 'Dr', 'Pl', 'Ln', 'Rd', 'Ct', 'Ter']
6 zip_codes = ['98101', '98102', '98103', '98104', '98105', '98106', '98107', '981
7
8 rows = 6996
9 data = [['House Number', 'Street Name', 'Street Type', 'Zip Code']]
10
11 for _ in range(rows):
12     house_num = random.randint(1, 999)
13     street_name = random.choice(street_names)
14     street_type = random.choice(street_types)
15     zip_code = random.choice(zip_codes)
16     data.append([house_num, street_name, street_type, zip_code])
17
18 with open('seattle_addresses.csv', 'w', newline='') as f:
19     writer = csv.writer(f)
20     writer.writerows(data)
21
22 print('CSV file "seattle_addresses" generated with 6996 rows!')
```

2. Some position don't have accomodation type while others do? Not a problem!
CHOOSE+RANDBETWEEN+UNIQUE(earlier) – and we have it handled!

	AL	AM	AN
3817	f	House	Entire home/apt
3818	f	Condominium	Entire home/apt
3819	t	Apartment	Entire home/apt
3820	t	Camper/RV	Entire home/apt
3821	t		Private room
3822	t		Entire home/apt
3823	t		Entire home/apt
3824	t		Entire home/apt

- 2.1. Or like that, for **bed_type**.

	AS	AT
1	Real Bed	{ "Cable TV", "Wireless Internet", Kitchen, "Free Parki
1	Real Bed	{TV, "Wireless Internet", Kitchen, "Free Parking on Pr
1	Real Bed	{TV, "Cable TV", Internet, "Wireless Internet", Kitchen
3	Pull-out Sofa	
1	Real Bed	

- 2.2. The **UP TO DATE** reviews file has pretty many positions though... Let's clean them and see what we have.

listing_id	date	reviewer_id	reviewer_name	comments
6606	5664	2009-07-17	18085 Vivian	"The Urban Cottage is comfortable, beautiful, fun and really convenient! Joyce is an amazing host and super friendly. The Wallingfo
6606	338761	2011-06-27	434031 Elliott	"Joyce was a wonderful host and the urban cottage is a such an awesome place to stay (quiet, clean, comfortable, private). I high
6606	467904	2011-08-22	976182 Allie	"Beautiful cottage and warm hospitality from Joyce. Even though we never got a chance to see each other I felt welcome. The neig
6606	480017	2011-08-27	997921 Brittney	"Joyce is a wonderful host! She is warm, helpful and fun to visit with. The cottage is cozy, bright and has all the comforts you co
6606	487278	2011-08-30	206903 Pascal	"Joyce's cottage is the perfect Seattle location! It's close Fremont, Ballard and Wallingford. The cottage has everything you need
6606	505388	2011-09-06	552477 Ivy	"Joyce's cottage and home were absolutely lovely! So private and pretty, and so centrally located in Seattle. Joyce is a wonderful hos
6606	570582	2011-09-26	110380 Wendy	"Sweet little place, quiet, cozy, great location. Loved taking outdoor showers. Composting toilet is just fine, does make the room
6606	1322495	2012-05-21	2354790 Salve	"Joyce accommodated the needs of my daughter and me on very short notice. We were thrilled at the quality of the Urban Cott
6606	1395422	2012-06-01	1845181 Susan And David	"The Urban Cottage was a wonderful place to stay for our visit to Seattle and was very nicely furnished with all that
6606	1575812	2012-06-27	1821528 Ivan	"Joyce was a great host and she made us feel very much at home. The cottage is in a prime neighborhood in Wallingford with ea

- 2.2.1. Apparently won't need those...

A	B
listing_id,date,reviewer_id,reviewer_name,comments	x
 	Yes
 	Yes
 	Yes
 	Yes
 	Yes
 	Yes
 	Yes

- 2.2.2. Some data (numeric) was not read well by Excel after using **Data to columns**. Good that I had a backup file.

listing_id	date	reviewer_id	reviewer_name	comments
98328113	12516651	1.414114E+18	11.07.2025	2.3E+08 Clayt
98328113	12566458	1.414114E+18	12.07.2025	5.4E+08 Orla
98328113	12617497	1.414114E+18	19.07.2025	1.6E+08 Joyce
98328113	12668562	1.414114E+18	20.07.2025	2E+08 Jane
98328113	12770043	1.414114E+18	22.07.2025	3E+07 Mari
98328113	12828762	1.414114E+18	25.07.2025	1.3E+07 Mich
98328113	12828762	1.414114E+18	27.07.2025	2.1E+08 Lana
98328113	12922368	1.414114E+18	04.08.2025	1800023 Sam
98328113	13023336	1.414114E+18	08.08.2025	4.8E+08 Nick
98328113	13111025	1.414114E+18	17.08.2025	3E+08 Terry
98328113	13430170	1.414114E+18	30.08.2025	5.2E+08 Jack
98328113	14190246	1.414114E+18	01.09.2025	4E+07 June
98328113	14248945	1.414114E+18	22.09.2025	7E+07 Linds
98328113	14306279	1.414114E+18	18.05.2025	6.9E+08 Santl
98328113	14408311	1.414114E+18	17.08.2025	5.9E+08 Kelvi
98328113	14545797	1.414114E+18	12.05.2025	6.7E+08 Sami
98328113	14545797	1.414114E+18	15.05.2025	4.2E+07 Jame
98328113	14545797	1.414114E+18	07.06.2025	5E+07 Derel

- 2.2.3. Uf, cleaned and transferred over to the base dataset. 660k reviews in total.


660657	15086633349452706	15125057149980095	17.09.2025	526418307	Abai Timurovich	The place v
660658	15086633349452706	15132283696065981	18.09.2025	5797957	Darren	Loved our :
660659	15086633349452706	15154525337124636	21.09.2025	43268500	Sarah	Pros:
660660	15109762030252905	15147477048674354	20.09.2025	113219	Barbara	Three frien
660661	15109762030252905	15154445574754474	21.09.2025	668748876	Luis	Awesome
660662						

[illegible]

The screenshot shows the Tableau interface with a join operation between 'Data Source' and 'Calendar' tables. The join type is set to 'Full Outer'. The resulting table has columns 'id' and 'listing_id'.

⚠ Listings

Name Listings

Description 

No description available.

Fields

Type	Field Name	Physical Ta...	Rem...
------	------------	----------------	--------

Data preview unavailable

The screenshot shows the Tableau interface with a join operation. The 'Listings' table is joined with the 'Calendar' table, and the result is then joined with the 'Reviews' table. The join type is set to 'Inner'.

Filter Data

Tableau Public only works with less than 15 000 000 rows of data. Your data exceeds this limit. Add filters below to reduce the number of rows to import.

Filters

Filter	Scope	Details
--------	-------	---------

Add... Edit... Remove...

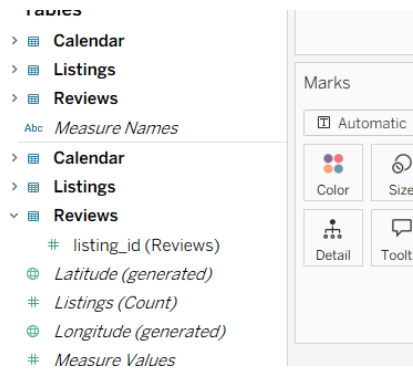
Number of Rows

104 159 519 rows in the original data source

Define additional filters

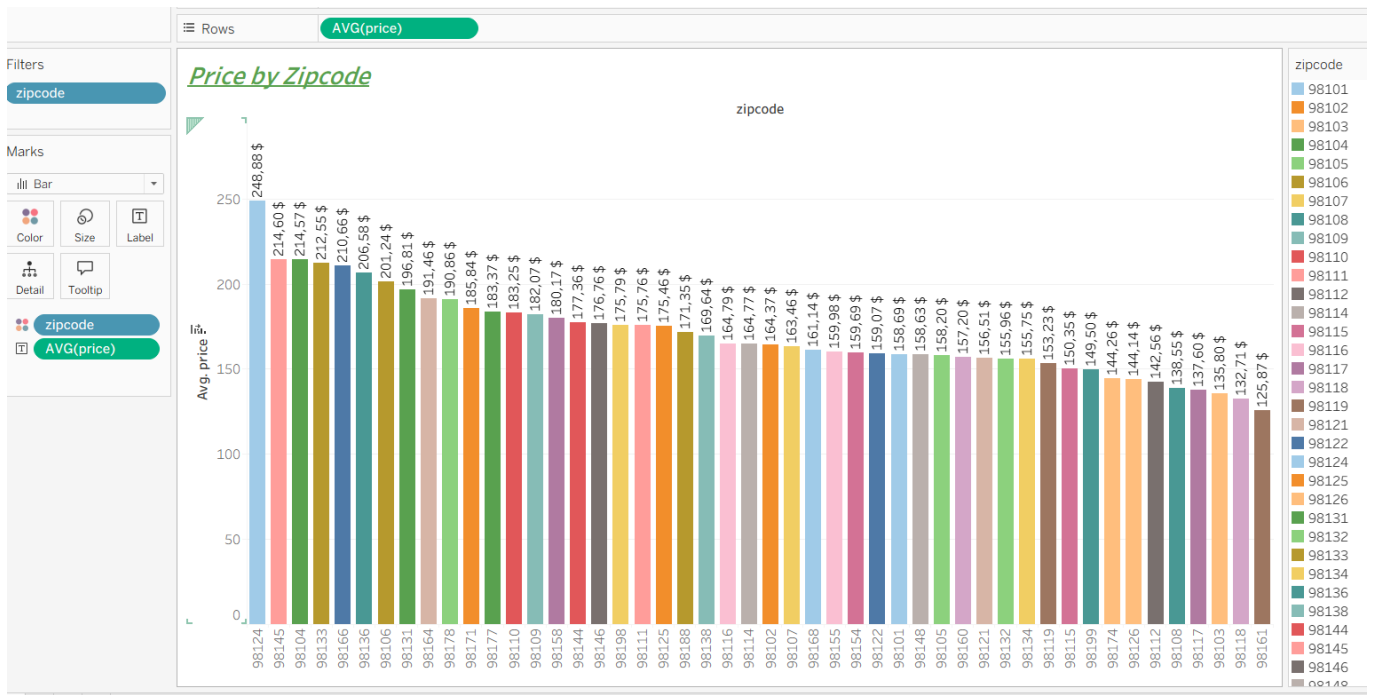
OK Cancel

2.4.4. It took really a while, but I manage to reduce it down to Tableau limits – from 104 mill rows to around 15 mill. It works now and that's great. And this all will be about verifying based on the

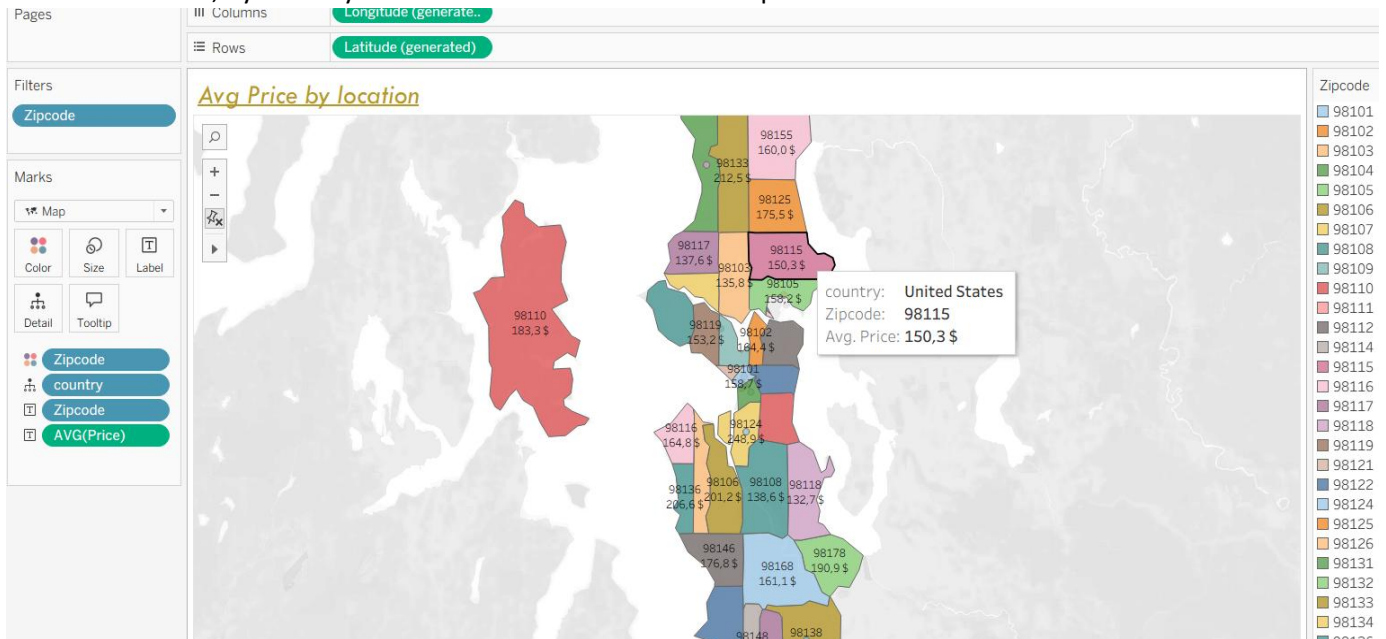


available data – where is the best to buy home and start Airbnb business.

3. We will start from the Zip code and the average price for the accommodation. Let's also color it and sort. And with that it's the first visualization, it's done.



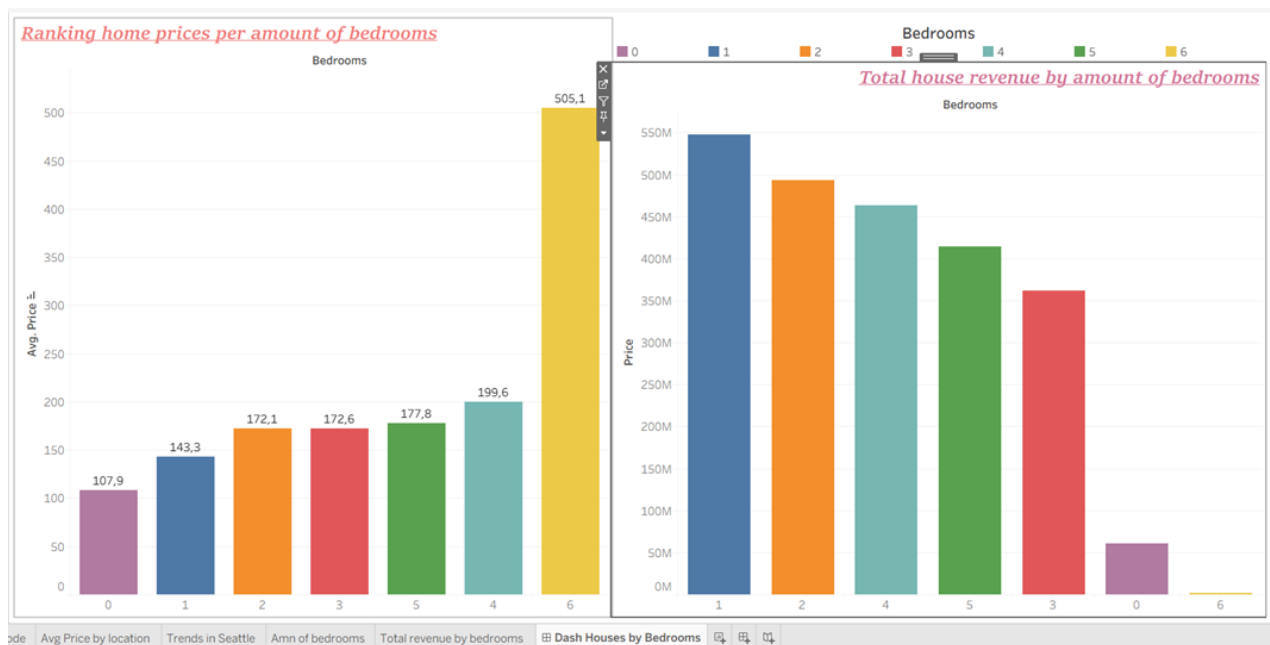
4. The next sheet – a **Map** style. The colors here match the colors by zip codes from the previous sheet, by the way. The data used here matches the previous visualization.



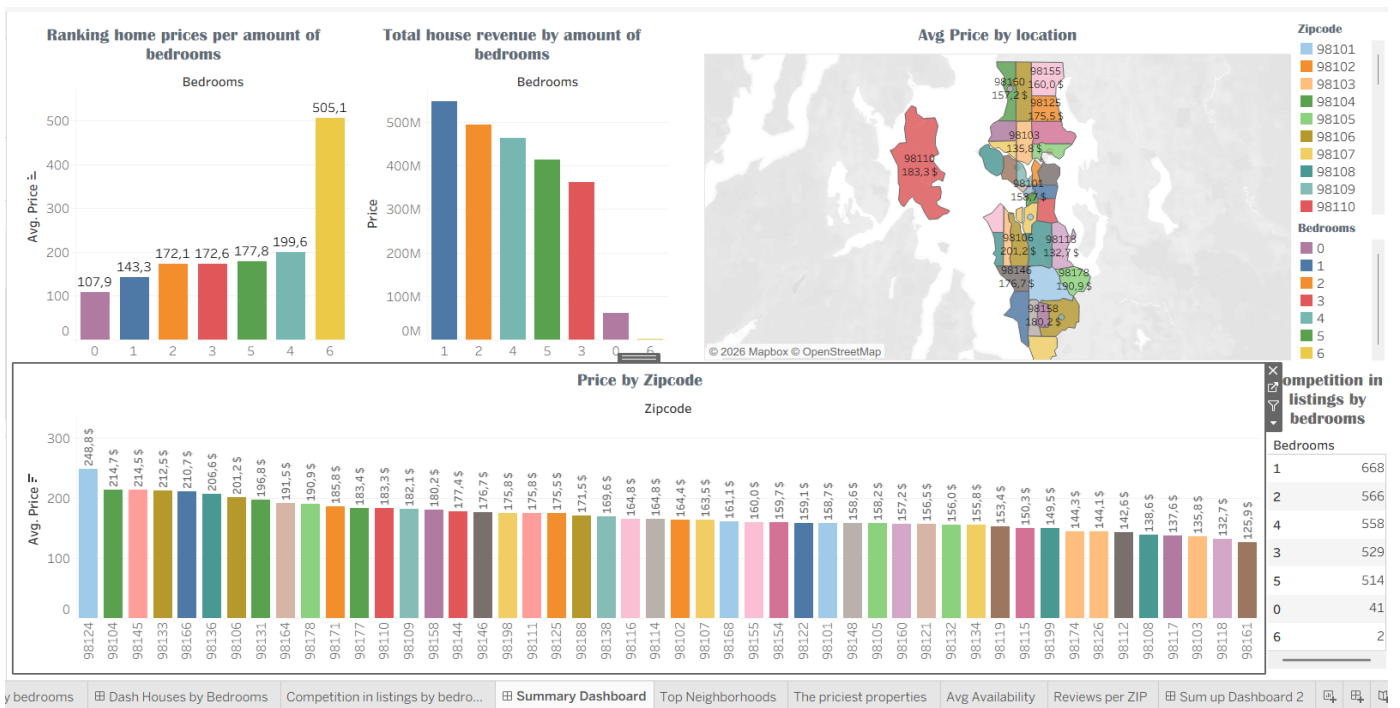
- The next one will be showing the visualization by years and prices. Prices fluctuated early on, declined through 2019, peaked around 2021 amid post-pandemic demand, and have since shown a gradual cooling trend—ending near 100 in 2025—with a subtle downward trajectory highlighted by the trend line. This reflects real-world patterns of tech-driven growth followed by recent market stabilization due to higher interest rates and increased inventory.



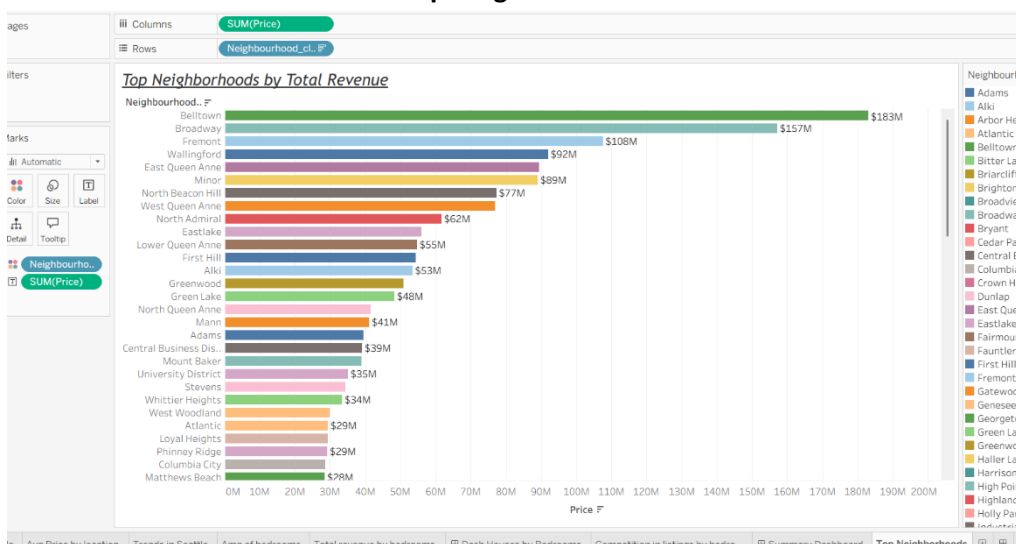
- Next step – adding two worksheets where **Avg prices & Total revenue** for houses shown by amount of **Bedrooms**. Basically the same data used, so not to waste much time – one visual with created first Dashboard. Totally, 1 bedroom-houses are bringing the most efficiency by Avg price and total revenue. So, the user doesn't have to by 2-3, or even 6 bedrooms house to rent it out.



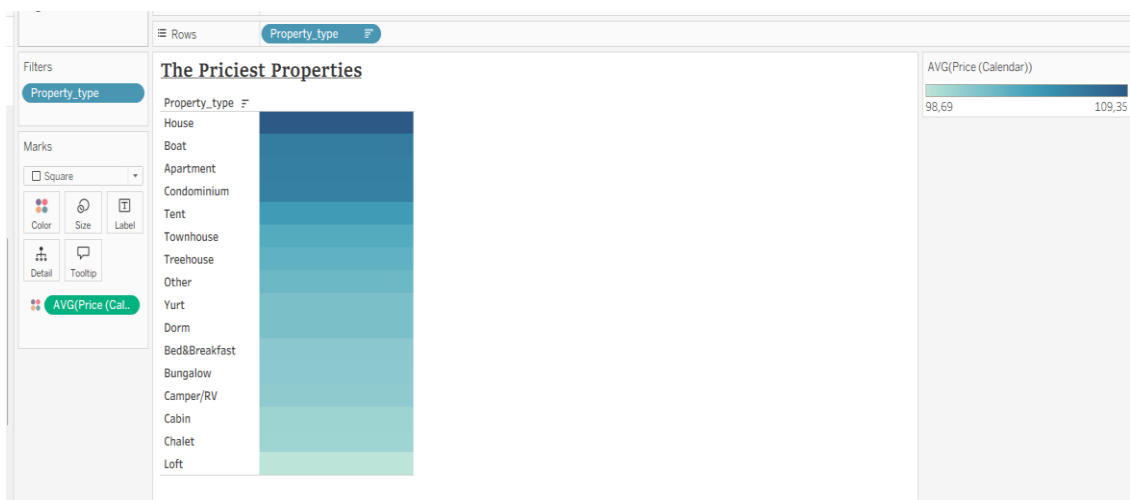
7. And with that is how the final **Dashboard** looks like!



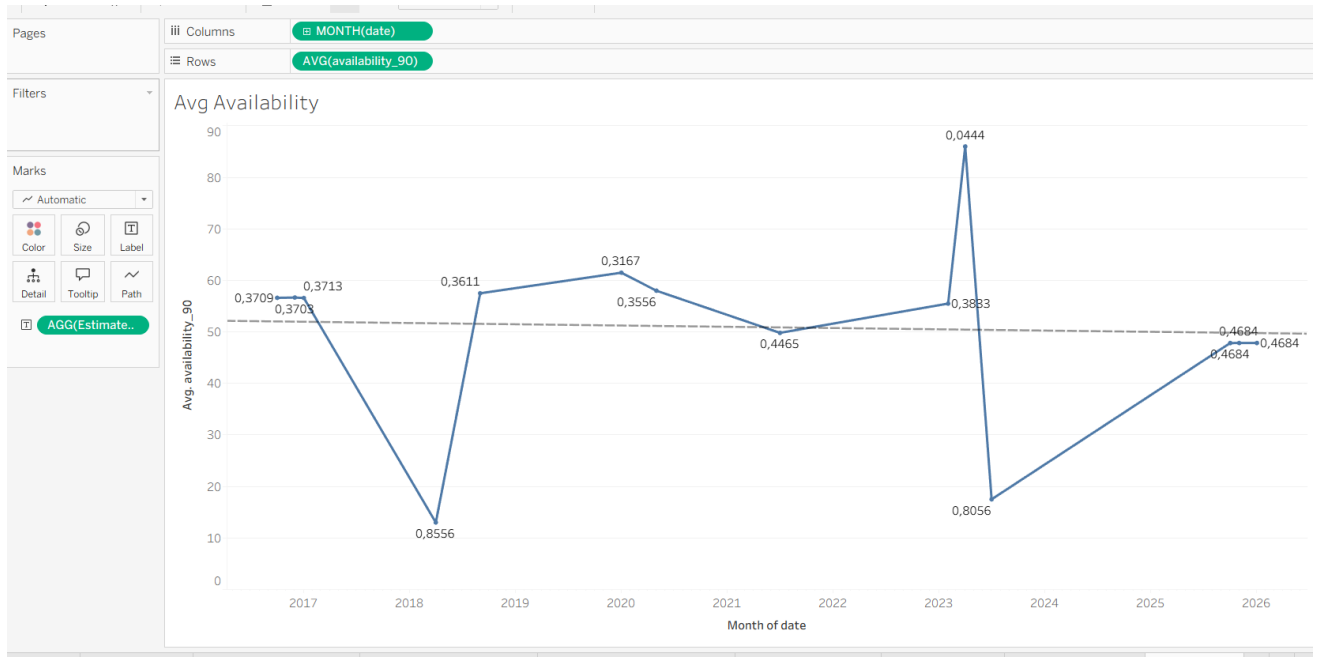
8. Additional sheet with **Top Neighborhoods**.



9. By that one now, we can see that the **Priciest** property (avg Prices) are Houses.



10. With this worksheet we can see **Avg Availability Over time**. Here the lowest value means that there were the highest Occupancy at that period of time.



11. And just like that – I finished the second **Dashboard** for this project!

