

TABLEAU PROJECT

Airbnb Analysis

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

	A	B	C	D	E	F	G	H	I	K	M	N	O	P	Q
1	id,name,hst_id,host_name,														
2	6606,"Fab private seattle url"														
3	9419,Glorious sun room w/ memory foambed														
4	9596,"the down home , spacious														
5	11012,"the orange house, quiet 'n central"														
6	25002,Beautiful Private Spot														
7	26795,Lake Union Cottage -														
8	119103,Tiny Tiny Guesthouse														
9	202251,Seattle Phinney Ridge														
10	21036,"The 5-Star House"														
11	215882,Relaxing Garden Cott														
12	219532,"Fun home in Funky F														
13	226536,Seine Room in Sun														
14	226677,Sunny Parisian room														
15	229990,All Beach 1/2 block to														

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.

A		B		C		D		E		F		G		H	
		General		Text				Text				Text			
1	id,name,hst_id,host_name,			Text				Text							
2	6606,"Fab private seattle url"			Date	DMY			Text							
3	9419,Glorious sun room w/ memory foambed			Text				Text							
4	9596,"the down home , spacious			Text				Text							
5	11012,"the orange house, quiet 'n central"			Text				Text							
6	25002,Beautiful Private Spot			Text				Text							
7	26795,Lake Union Cottage -			Text				Text							
8	119103,Tiny Tiny Guesthouse			Text				Text							
9	202251,Seattle Phinney Ridge			Text				Text							
10	21036,"The 5-Star House"			Text				Text							
11	215882,Relaxing Garden Cott			Text				Text							
12	219532,"Fun home in Funky F			Text				Text							
13	226536,Seine Room in Sun			Text				Text							
14	226677,Sunny Parisian room			Text				Text							
15	229990,All Beach 1/2 block to			Text				Text							

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

595	10245/1169682152698	Line Ratio Room 2umin 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.

B3820	=TEXTJOIN(TRUE;"https://www.airbnb.com/rooms/";A3820)
3813	8578490 https://www.airbnb.com/rooms/
3814	3383329 https://www.airbnb.com/rooms/
3815	8101950 https://www.airbnb.com/rooms/
3816	8902327 https://www.airbnb.com/rooms/
3817	10267360 https://www.airbnb.com/rooms/
3818	9604740 https://www.airbnb.com/rooms/
3819	10208623 https://www.airbnb.com/rooms/
3820	6606 https://www.airbnb.com/rooms/6606
3821	9419

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.

722270	въѣHistoric Loft Suite @ Downtown Albany	22270 Historic Loft Suite @ Downtown Albany
868857	вѣCozy Garden Suite Full Amenities @Downtown Albany	58857 Cozy Garden Suite Full Amenities @Downtown Albany
993402	вѣHistoric Full Amenities 2BR Apt @Downtown Albany	39402 Historic Full Amenities 2BR Apt @Downtown Albany
330970	Muhammed Ali Lounge & Pool Hall вѣўѣ	30970 Muhammed Ali Lounge & Pool Hall
528939	UNIQUE Gem in HEART of Albany в... Spiral Staircase	28939 UNIQUE Gem in HEART of Albany Spiral Staircase
529316	WOW в... Bright Apt в... PRIME downtown Walk Everywhere	29316 WOW Bright Apt PRIME downtown Walk Everywhere
529447	в... 2 BEDROOM / 1 BATH TRENDY Apt в... Downtown Albany	29447 2 BEDROOM / 1 BATH TRENDY Apt Downtown Albany
207226	в... RARE 1BR Apt with Character GREAT Location! W/Dв...	27226 RARE 1BR Apt with Character GREAT Location! W/D
890336	в...CHIC, Renovated Brownstones... in PRIME DT Albany	90336 CHIC, Renovated Brownstonein PRIME DT Albany
579900	в...MODERN 1BR GREAT Location Downtown! W/D Parking!в...	79900 MODERN 1BR GREAT Location Downtown! W/D Parking!

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

	G	H
ac/pict	14703116	https://www.airbnb.com/users/show/ Gil
14942		https://www.airbnb.com/users/show/ Joyce
30559		Angielena

1.6. Also needed to replace **longitude & latitude** values “.” with “,” to match.

H	I	J
-73.7537	room_type	-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	Entire home/apt	-122,3661741
-73.80553	Private room	-122,3685191
-73.82438	Private room	-122,375856
-73.78297	Entire home/apt	-122,3572161
-73.76867	Private room	-122,3668866
-73.83743	Entire home/apt	-122,3584778

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

- (Select All)
- Ballard, Seattle, WA
- é¥¿é...å¾, 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 Im 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.

Get & Transform Data					
E2	B	C	D	E	F
1	House Nur Street Name	Street Typ	Zip Codde		
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.

```
ćwiczenia > 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', '98108'
7
8 rows = 6996
9 data = [[ 'House Number', 'Street Name', 'Street Type', 'Zip Codde']]
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	A
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.

	A1	listing_id	date	reviewer_id	reviewer_name	comments
1	listing_id	date	reviewer_id	reviewer_name	comments	
2	6606589612029-07-17	15086633349452706	15125057149980095	573113	Clayt	The Urban Cottage is comfortable, beautiful, fun and really convenient. Joyce is an amazing host and super friendly. The Wallingford neighborhood is such an awesome place to stay (quiet, clean, comfortable, private). I highly recommend! Joyce was a wonderful host and warm hospitality from Joyce. Even though we never got a chance to see each other I felt welcome. The neighborhood is great. The cottage is cozy, bright and has all the comforts you could want. It's close Fremont, Ballard and Wallingford. The cottage has everything you need for a great stay.
3	6606589612029-07-17	15086633349452706	15125057149980095	573114	Orla	Very nice place, very clean, very quiet. Great location, great price, great host! Thanks for the wine, coffee and internet.
4	6606589612029-07-17	15086633349452706	15125057149980095	573115	Joyce	Wallingford
5	6606589612029-07-17	15086633349452706	15125057149980095	573116	Jane	Beautiful cottage and warm hospitality from Joyce. Even though we never got a chance to see each other I felt welcome. The neighborhood is great. The cottage is cozy, bright and has all the comforts you could want. It's close Fremont, Ballard and Wallingford. The cottage has everything you need for a great stay.
6	6606589612029-07-17	15086633349452706	15125057149980095	573117	Mari	Wallingford
7	6606589612029-07-17	15086633349452706	15125057149980095	573118	Mich	Wallingford
8	6606589612029-07-17	15086633349452706	15125057149980095	573119	Lana	Wallingford
9	6606589612029-07-17	15086633349452706	15125057149980095	573120	Sam	Wallingford
10	6606589612029-07-17	15086633349452706	15125057149980095	573121	Nick	Wallingford
11	6606589612029-07-17	15086633349452706	15125057149980095	573122	Taryn	Wallingford
12	6606589612029-07-17	15086633349452706	15125057149980095	573123	Jack	Wallingford
13	6606589612029-07-17	15086633349452706	15125057149980095	573124	June	Wallingford
14	6606589612029-07-17	15086633349452706	15125057149980095	573125	Linds	Wallingford
15	6606589612029-07-17	15086633349452706	15125057149980095	573126	Santti	Wallingford
16	6606589612029-07-17	15086633349452706	15125057149980095	573127	Kelvi	Wallingford
17	6606589612029-07-17	15086633349452706	15125057149980095	573128	Sams	Wallingford
18	6606589612029-07-17	15086633349452706	15125057149980095	573129	Jame	Wallingford
19	6606589612029-07-17	15086633349452706	15125057149980095	573130	Derek	Wallingford
20	6606589612029-07-17	15086633349452706	15125057149980095	573131	Abai	Wallingford
21	6606589612029-07-17	15086633349452706	15125057149980095	573132	Timurovich	Wallingford
22	6606589612029-07-17	15086633349452706	15125057149980095	573133	The place	Wallingford
23	6606589612029-07-17	15086633349452706	15125057149980095	573134	Pros:	Wallingford
24	6606589612029-07-17	15086633349452706	15125057149980095	573135	Uf,	Wallingford
25	6606589612029-07-17	15086633349452706	15125057149980095	573136	cleaned	Wallingford
26	6606589612029-07-17	15086633349452706	15125057149980095	573137	and transferred over to the base dataset.	Wallingford
27	6606589612029-07-17	15086633349452706	15125057149980095	573138	660k reviews in total.	Wallingford

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

	B13	listing_id	id	date	reviewer_id	reviewer_name	comments
1							
13							Yes
17							Yes
20							Yes
23							Yes
25							Yes
27							Yes
29							Yes
~							

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
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	1515445574754474	21.09.2025	668748876	Luis	Awesome
660662						

2.3. Well, for the **calendar** data (appointments) there are some that I can leave aside this time, I guess...

A screenshot of a database viewer showing a table named "calendar". The table has columns: id, listing_id, date, start_time, end_time, status, and is_booked. There are three rows:

- Row 1: id=1045636, listing_id=93568848, date=28.04.2026, start_time=f, end_time=, status=5, is_booked=365
- Row 2: id=1045637, listing_id=93568848, date=29.04.2026, start_time=f, end_time=, status=5, is_booked=365
- Row 3: id=1045638, listing_id=93568848, date=30.04.2026, start_time=f, end_time=, status=5, is_booked=365

2.4. Now as it's all clean I can go and start with **Tableau**. Trying to remember to keep the data type same for JOINS.

The screenshot shows the Tableau Data Source editor for the "Listings" data source. It displays a join configuration between the "Listings" and "Calendar" tables. The join is set to "Inner" and is based on the "id" field from both tables. The "Fields" pane on the left lists the "listing_id" field from the "Listings" table.

2.4.1. Aaand issue fixed.

The screenshot shows the Tableau Data Source editor for the "Listings" data source. It displays the joined table "Listings" which contains 56 fields and 1067219 rows. The "Fields" pane on the left lists the "listing_id" field from the joined "Listings" table.

2.4.2. And I also connect **listings & reviews** by listings id.

The screenshot shows the Tableau Data Source editor for the "Listings" data source. It displays a join configuration between the "Listings" and "Reviews" tables. The join is set to "Inner" and is based on the "id" field from both tables. The "Fields" pane on the left lists the "listing_id" field from the "Reviews" table.

2.4.3. Wellll, that I was not expecting... Well, at least not having so many rows. Will need to reduce them significantly.

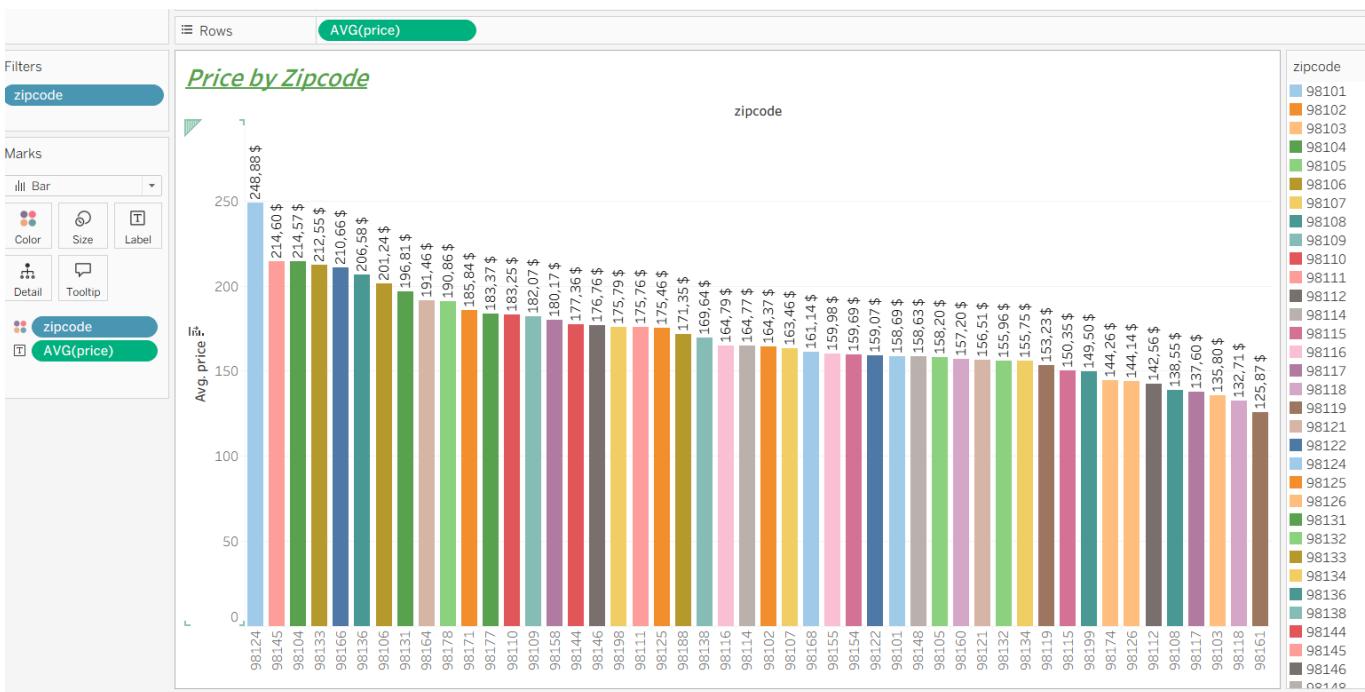
The screenshot shows the Tableau "Filter Data" dialog box. It displays a warning message: "Tableau Public only works with less than 15 000 000 rows of data. Your data exceeds this limit. Add filters below to reduce the number of rows to import." Below this, there is a "Filters" section with tabs for "Filter", "Scope", and "Details". Under "Scope", there is a "Number of Rows" input field set to "104 159 519 rows in the original data source". At the bottom, there is a note: "⚠ Define additional filters" and buttons for "OK" and "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

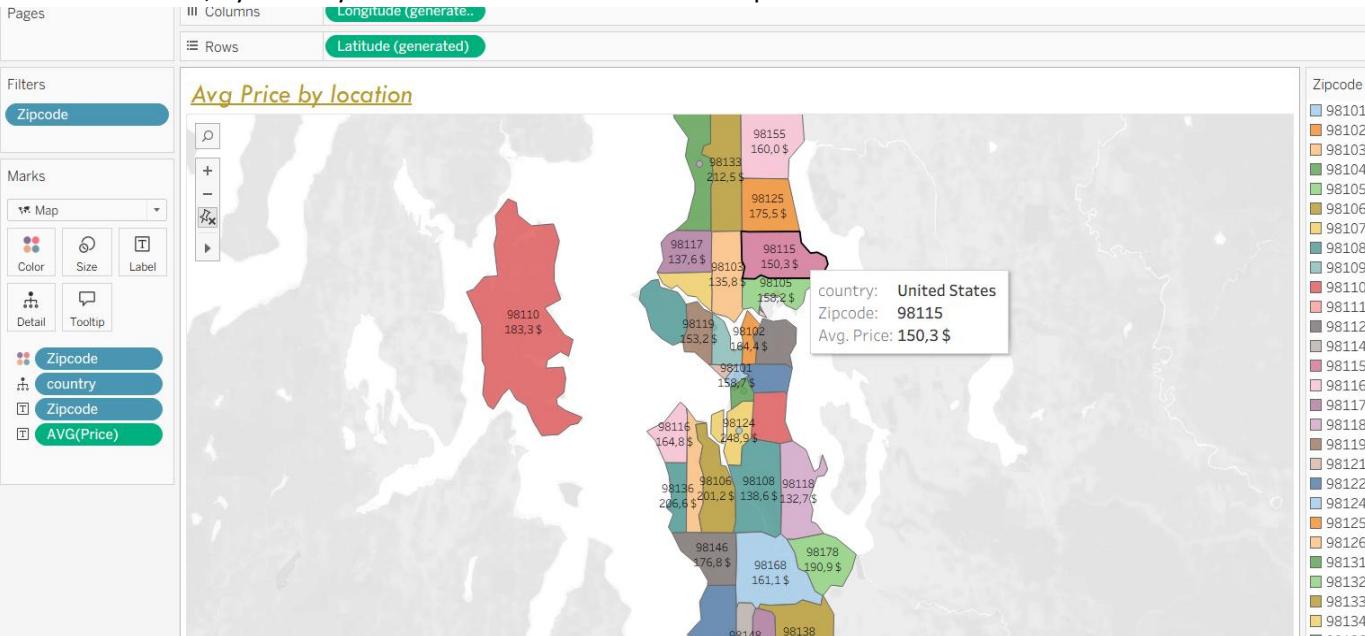
The screenshot shows the Tableau interface with the 'Marks' shelf open. Under the 'Color' section, there is a small circular icon with three dots of different colors (red, green, blue). This indicates that the 'zipcode' field is currently mapped to color in the visualization.

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

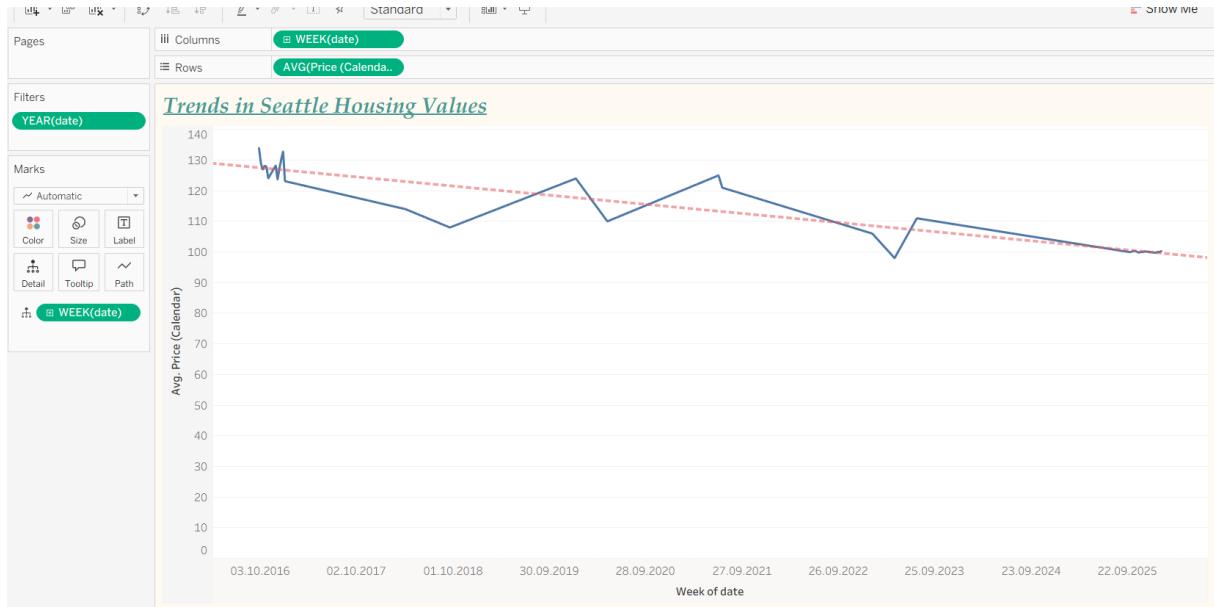
- We will start from the Zip code and the average price for the accommodation. Let's also color it and sort. And with that it's the first visualization, it's done.



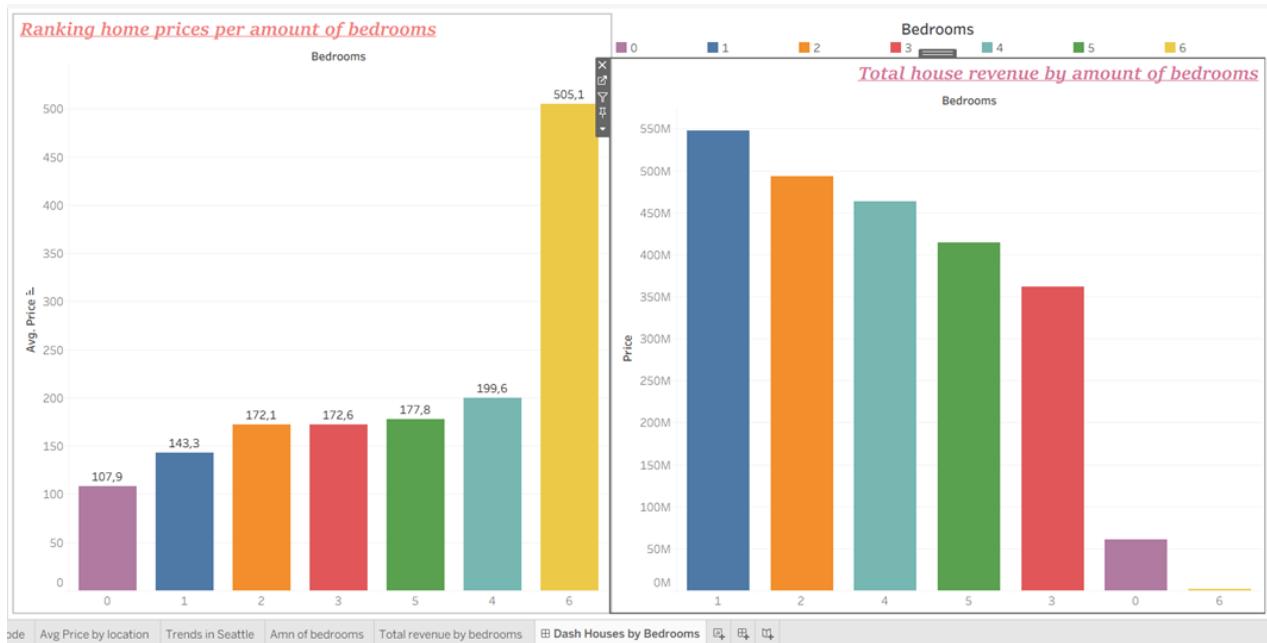
- The next sheet – a **Map** style. The colors here match the colors by zip codes from the previous sheet, by the way. The data used here matches the previous visualization.



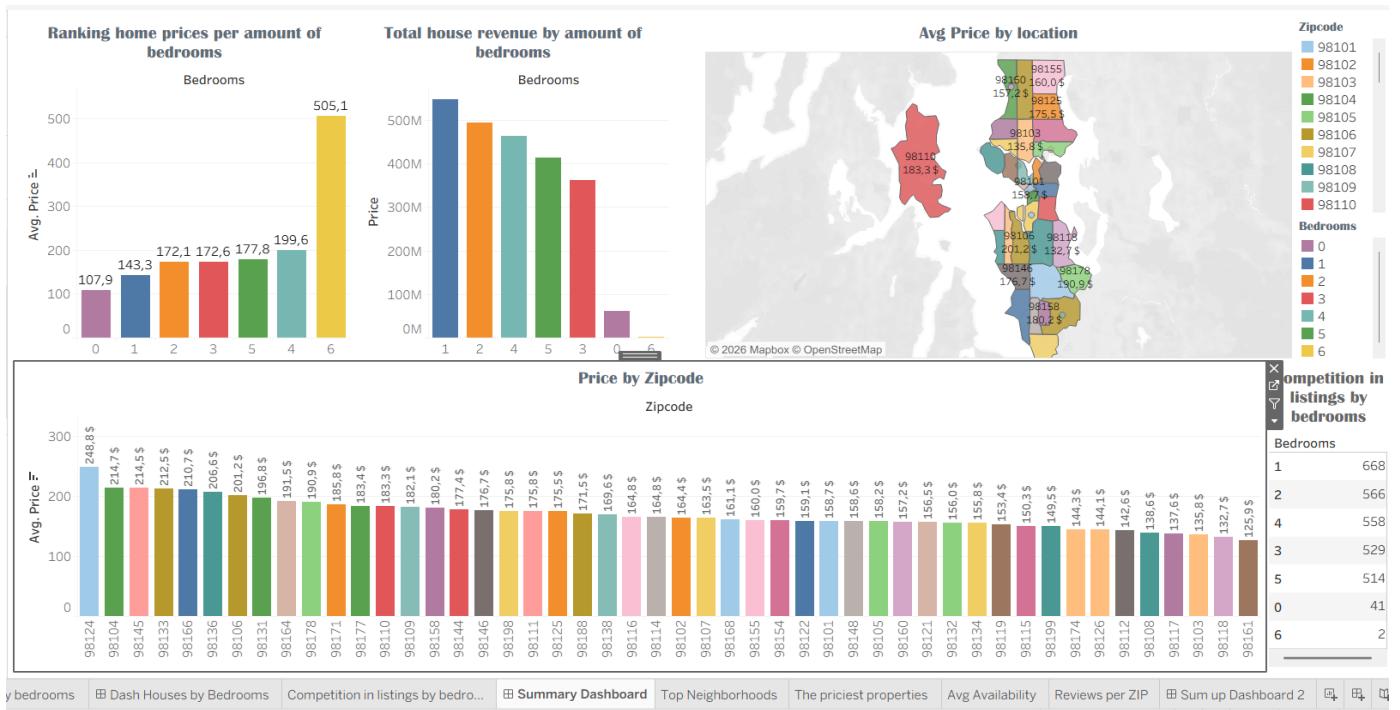
5. The next one will be showing the visualization by years and prices. Prices fluctuated early on, declined through 2019, peaked around 2021 amid post-pandemic demand, and have since shown a gradual cooling trend—ending near 100 in 2025—with a subtle downward trajectory highlighted by the trend line.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



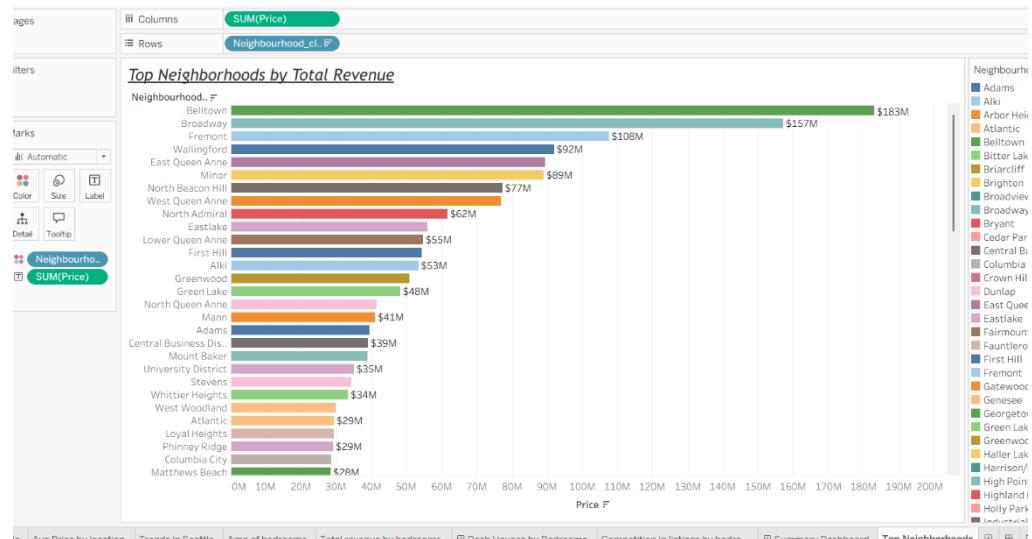
6. Next step – adding two worksheets where **Avg prices & Total revenue** for houses shown by amount of **Bedrooms**. Basically the same data used, so not to waste much time – one visual with created first Dashboard. Totally, 1 bedroom-houses are bringing the most efficiency by Avg price and total revenue.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



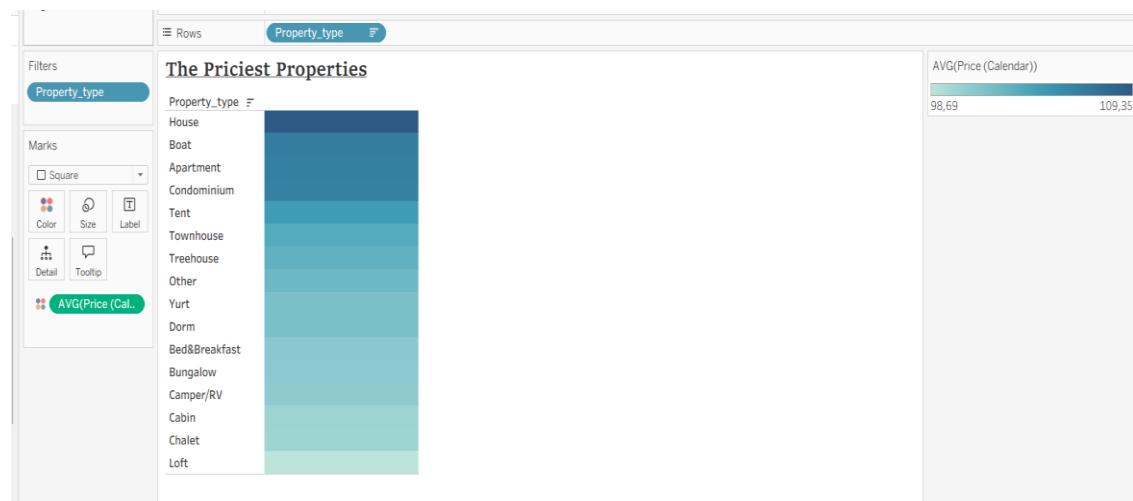
7. And with that is how the final Dashboard looks like!



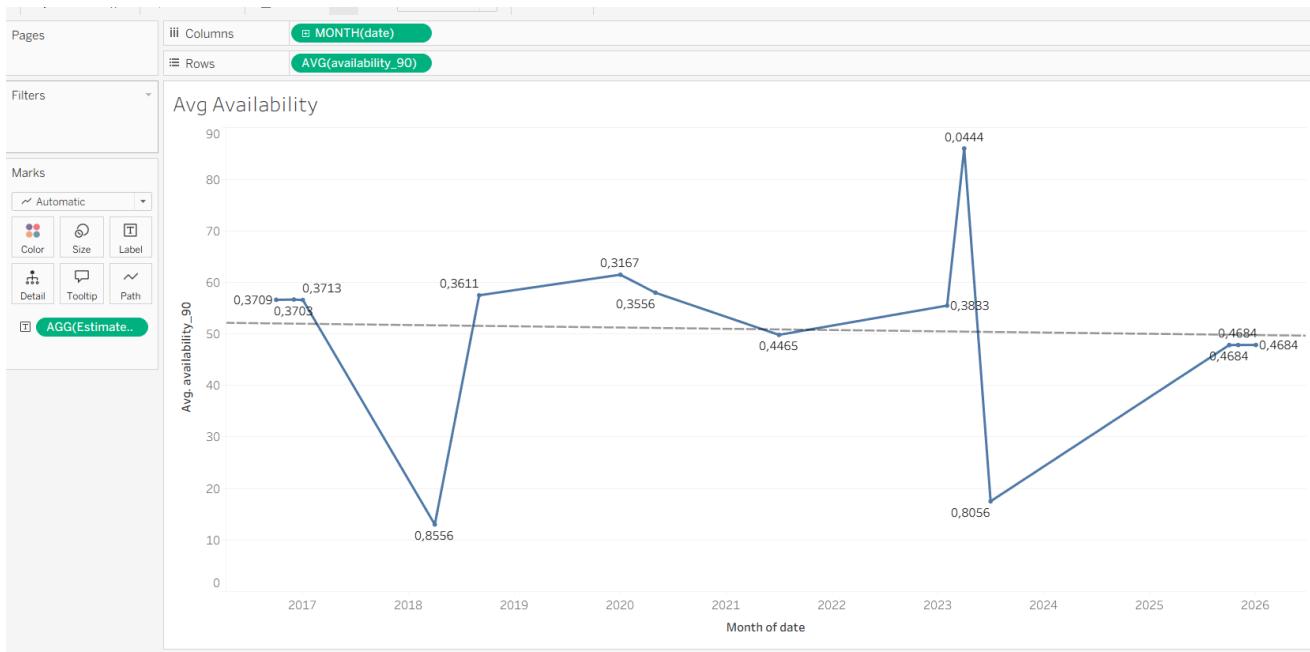
8. Additional sheet with Top Neighborhoods.



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!

