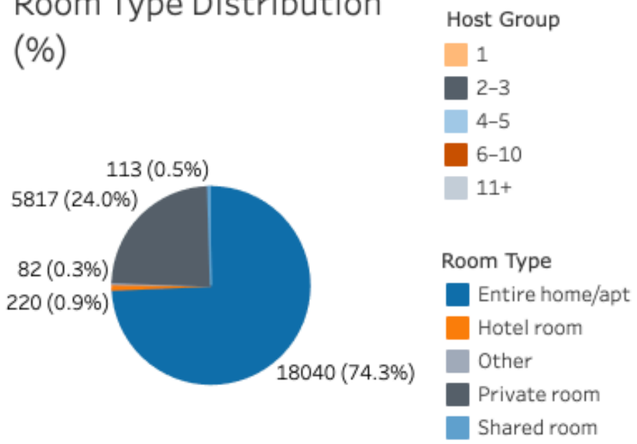
Airbnb Lisbon Dashboard: Prices, Availability, Listings

Room Type: (All)

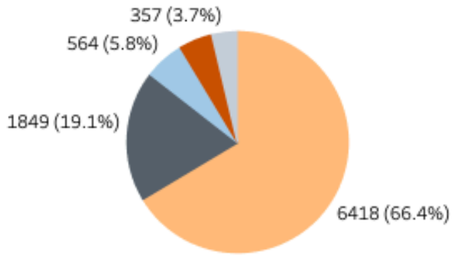
Host Group: (All)

Price: 1 — 1000

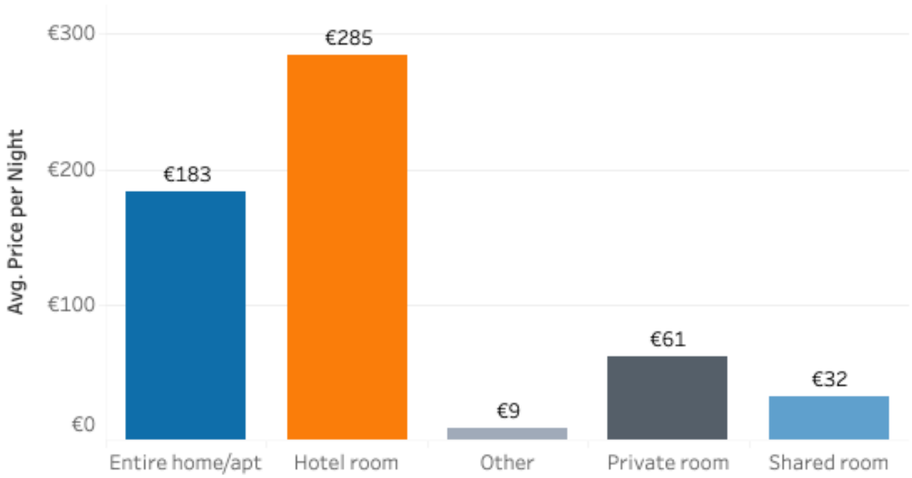
Room Type Distribution (%)



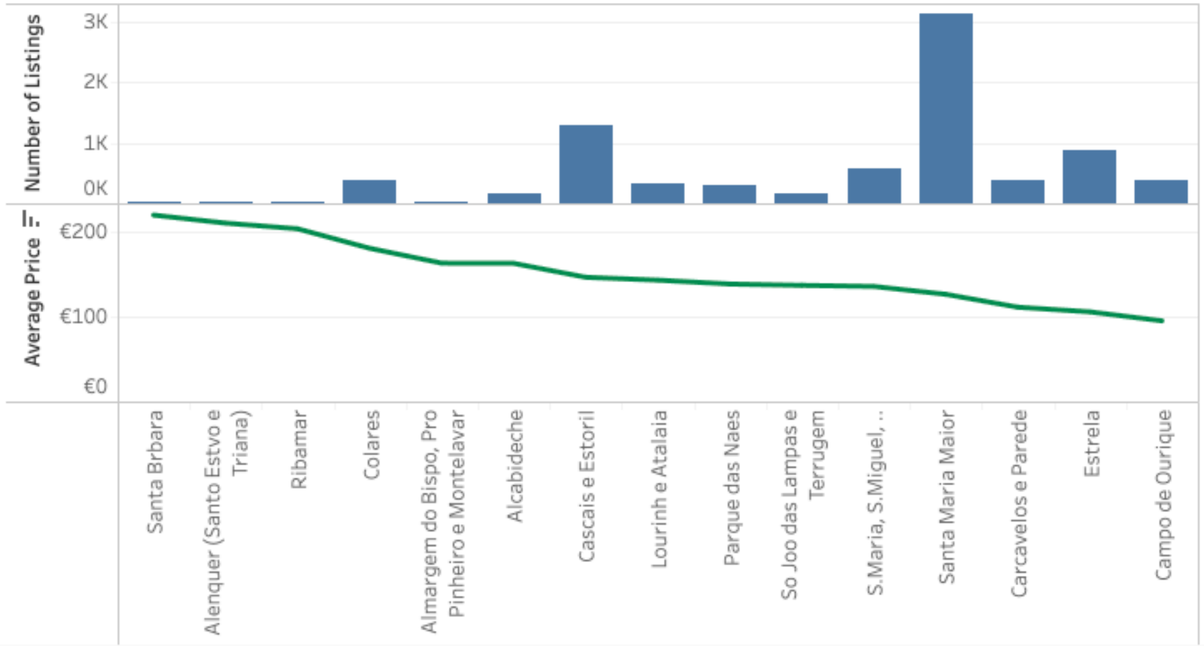
Host Group Distribution (%)



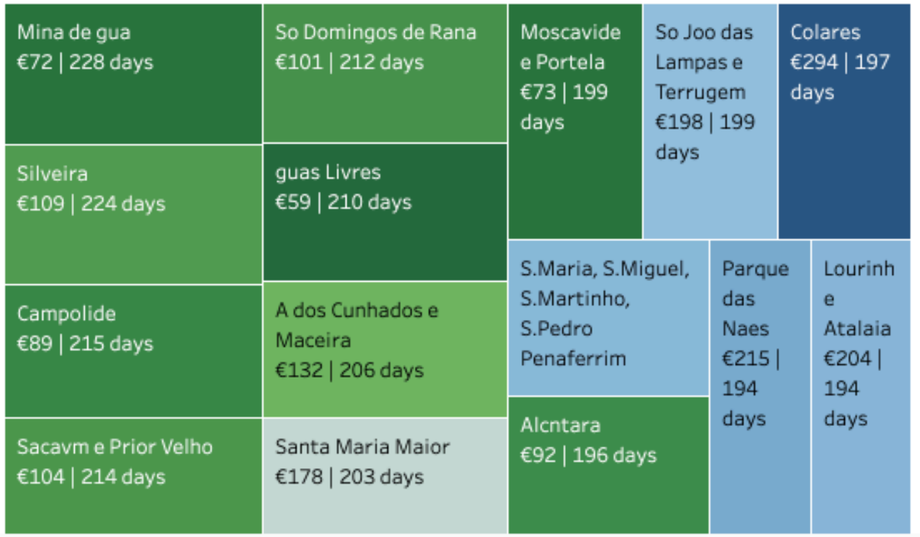
Average Price per Room Type



Average Price and Listings Count by Neighbourhood



Top 15 Lisbon Neighbourhoods: High Availability, Low Price



Airbnb Lisbon Dashboard: Prices, Availability, Listings

1. Project Description

1.1. Goal

The goal of this analysis is to explore the housing supply on Airbnb in the city of Lisbon and identify opportunities to improve the platform's efficiency.

The analysis aims to identify the factors influencing rental price, housing availability throughout the year, and host activity.

Special attention is given to:

- identifying the most popular housing formats,
- determining the neighborhoods with the highest concentration of supply and demand,
- analyzing the relationship between price, reviews, and number of bookings.

The insights obtained can be used for:

- recommendations to hosts regarding optimal pricing and increasing conversion,
- identifying target housing segments for marketing campaigns,
- improving the product experience of the platform for renters.

1.2. Data Source:

I downloaded the data from the website https://insideairbnb.com/get-the-data/, specifically selecting the file *listings.csv* for analysis.

The data is dated March 8, 2025.

All further analysis was conducted using BigQuery.

2. Data Preparation and Analysis

Overview of the data:

```sql
SELECT
  column_name,
  data_type,
  is_nullable
FROM `my-test-project-445920`.INFORMATION_SCHEMA.COLUMNS
WHERE table_name = 'listings';
```

column_name	data_type	is_nullable	mean
id	STRING	YES	Primary Key
name	STRING	YES	Listing Title
host_id	STRING	YES	Host ID
host_name	STRING	YES	Host Name
neighbourhood_group	STRING	YES	Neighbourhood Group
neighbourhood	STRING	YES	Neighbourhood
latitude	FLOAT64	YES	Geo Coordinates (Latitude)
longitude	STRING	YES	Geo Coordinates (Longitude)
room_type	STRING	YES	Room Type (apartment, room)
price	INT64	YES	Price per Night
minimum_nights	INT64	YES	Minimum Nights
number_of_reviews	STRING	YES	Number of Reviews
last_review	STRING	YES	Last Review Date
reviews_per_month	FLOAT64	YES	Average Reviews per Month
calculated_host_listings_count	INT64	YES	Host's Total Listings Count
availability_365	INT64	YES	Availability (Days per Year)
number_of_reviews_ltm	STRING	YES	Number of Reviews in the Last 12 Months
license	STRING	YES	License

What we observe:

Our data contains 18 columns with different data types.

I additionally described what each column means.

With the following query, I wanted to review the general information about the data:

```sql
SELECT
  COUNT(*) AS total_rows,
  COUNT(DISTINCT id) AS unique_listings,
  COUNT(DISTINCT host_id) AS unique_hosts,
  COUNT(DISTINCT neighbourhood) AS unique_neighbourhoods,
  COUNT(DISTINCT room_type) AS unique_room_types
FROM `my-test-project-445920.airbnb_lisbon.listings`;
```

Query results

Save resul

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	total_rows	unique_listings	unique_hosts	unique_neighbourho	unique_room_types
1	24278	24270	9663	199	54

Here we can see that:

- unique_listings ≈ total_rows — almost all rows are unique
- 9,663 unique_hosts for 24,270 listings
- 199 unique_neighbourhoods
- 54 unique_room_types — I believe something was mislabeled here; this needs to be reviewed further, as the number of room types seems too high

Understanding the abnormal number of room types, I wanted to immediately take a closer look at what's wrong:

```sql
SELECT
  room_type,
  COUNT(*) AS total
FROM `my-test-project-445920.airbnb_lisbon.listings`
GROUP BY room_type
ORDER BY total DESC;
```

In the data, **I noticed many unclear or inconsistent labels.**

There are only 4 main room types.

The task is to keep those and remove the rest.

Query results

Save results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	room_type	total
1	Entire home/apt	18040
2	Private room	5817
3	Hotel room	220
4	Shared room	113
5	*null*	25
6	59	3
7	60	2
8	42	2
9	50	2
10	68	2
11	94	2
12	66	2

Results per page: 50 | 1 – 50 of 55

I kept the main room types:

```sql
CREATE OR REPLACE VIEW `my-test-project-445920.airbnb_lisbon.listings_cleaned` AS
SELECT
  *,
  CASE
    WHEN room_type IN ('Entire home/apt', 'Private room', 'Shared room', 'Hotel room') THEN room_type
    ELSE 'Other'
  END AS room_type_cleaned
FROM `my-test-project-445920.airbnb_lisbon.listings`;
```

Now I check what remains:

Query results

Job information | Results | Chart | JSON

Row	room_type_cleaned	total
1	Entire home/apt	18040
2	Private room	5817
3	Hotel room	220
4	Shared room	113
5	Other	88

Next, I want to check how many missing values we have in the table.

I selected only the columns I need for further analysis, as not all of them will be used in queries.

```sql
SELECT
  COUNT(*) AS total_rows,
  COUNTIF(room_type IS NULL) AS null_room_type,
  COUNTIF(price IS NULL) AS null_price,
  COUNTIF(minimum_nights IS NULL) AS null_minimum_nights,
  COUNTIF(neighbourhood IS NULL) AS null_neighbourhood,
  COUNTIF(availability_365 IS NULL) AS null_availability,
  COUNTIF(reviews_per_month IS NULL) AS null_reviews_per_month
FROM `my-test-project-445920.airbnb_lisbon.listings`;
```

Query results

Save results | Open in

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	total_rows	null_room_type	null_price	null_minimum_nights	null_neighbourhood	null_availability	null_reviews_per_mo
1	24278	25	3192	16	16	16	3559

We can see that:

There are 24,278 rows in total.

Some columns have a small number of missing values (minimum_nights, neighbourhood, availability_365 – 16 each).

There are many missing values in the reviews column (reviews_per_month), which is quite natural.

I immediately remove NULL values:

```sql
CREATE OR REPLACE VIEW `my-test-project-445920.airbnb_lisbon.listings_cleaned_final` AS
SELECT
  *,
  -- Replacing NULL in reviews_per_month with 0
  IFNULL(reviews_per_month, 0) AS reviews_per_month_cleaned
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
WHERE room_type IS NOT NULL
  AND price IS NOT NULL
  AND minimum_nights IS NOT NULL
  AND neighbourhood IS NOT NULL
  AND availability_365 IS NOT NULL;
```

And check whether the missing values have been removed:

```sql
SELECT
  COUNTIF(room_type_cleaned IS NULL) AS null_room_type,
  COUNTIF(price IS NULL) AS null_price,
  COUNTIF(minimum_nights IS NULL) AS null_minimum_nights,
  COUNTIF(neighbourhood IS NULL) AS null_neighbourhood,
  COUNTIF(availability_365 IS NULL) AS null_availability,
  COUNTIF(reviews_per_month_cleaned IS NULL) AS null_reviews_per_month_cleaned
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned_final`;
```

Query results

Save results

Job information | Results | Chart | JSON | Execution details | Execution graph

Row	null_room_type	null_price	null_minimum_nights	null_neighbourhood	null_availability	null_reviews_per_mo
1	0	0	0	0	0	0

Brief summary of the previous steps:

The dataset contains 24,278 records about housing listings in Lisbon.

These data include:

- 9,663 unique hosts
- 199 neighborhoods
- 54 different room type categories, of which 4 are main ones (Entire home/apt, Private room, Shared room, Hotel room)

Actions taken to simplify further analysis:

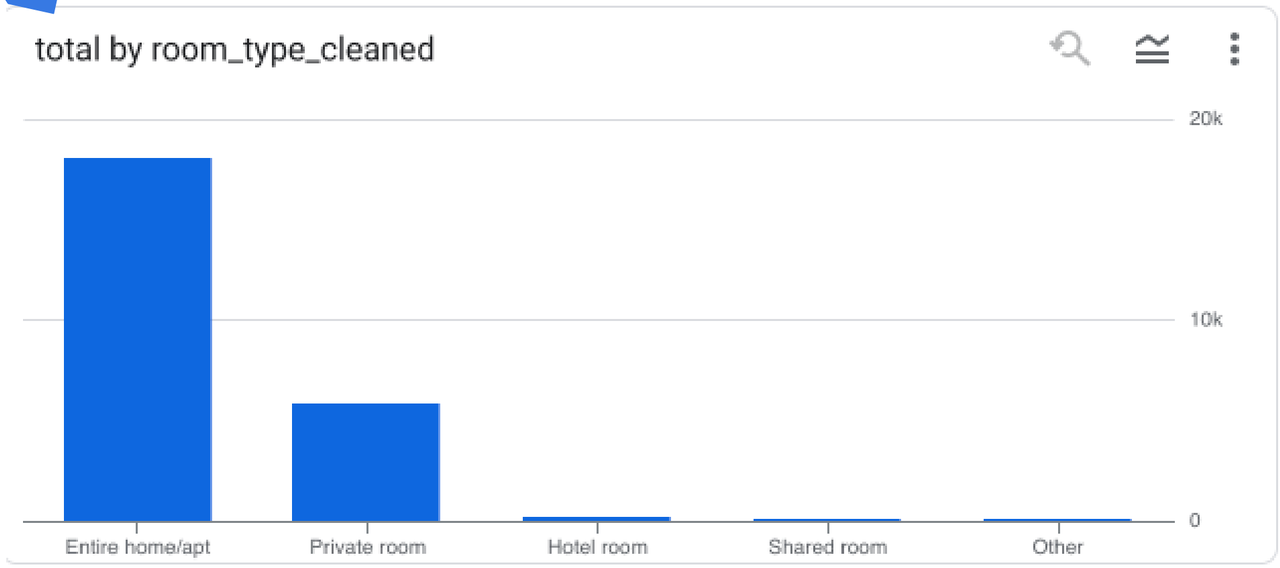
- The room_type field was cleaned — only 4 logical categories were kept
- Rows with NULL values in critically important columns (price, room_type, availability, etc.) were removed
- reviews_per_month was cleaned by replacing NULLs with 0 to allow inclusion of all listings, even inactive ones

3. SQL Analysis

Question #1

What types of housing are the most popular among hosts in Lisbon?

```sql
SELECT
  room_type_cleaned,
  COUNT(*) AS total
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
GROUP BY room_type_cleaned
ORDER BY total DESC;
```



We can see that **the most popular type of housing is "Entire home/apt"**.

Next is **"Private room", which also has a significantly higher number** than others.

Meanwhile, "Shared room" and "Hotel room" show much lower values.

Conclusion:

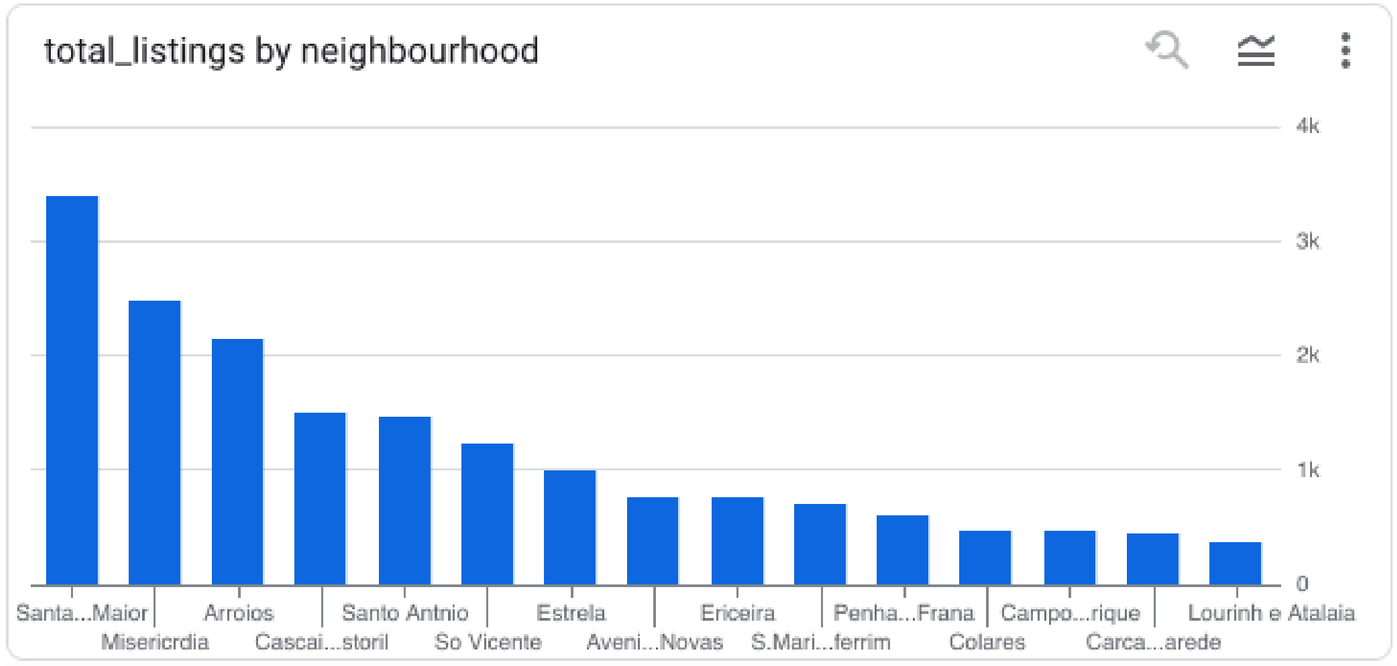
Airbnb can focus marketing efforts on "Entire home" as the core category.

Additionally, it may be beneficial to collaborate with developers and encourage them to open more "Hotel room" formats in tourist-heavy neighborhoods.

Question #2

Which neighborhoods in Lisbon have the highest number of listings?

```sql
SELECT
  neighbourhood,
  COUNT(*) AS total_listings
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
GROUP BY neighbourhood
ORDER BY total_listings DESC
LIMIT 15;
```



Here we can see that **the highest number of listings is** concentrated in the following five neighborhoods:

- Santa Maria Maior – 3,397 listings
- Misericórdia – 2,474
- Arroios – 2,137
- Cascais e Estoril – 1,506
- Santo António – 1,465

It is worth noting that these neighborhoods are indeed the most popular among tourists due to their proximity to historical landmarks, transport connections, and the most beautiful natural areas in Lisbon.

Neighborhoods with a lower number of listings are generally less convenient in terms of infrastructure and are not as attractive to short-term visitors.

Conclusion:

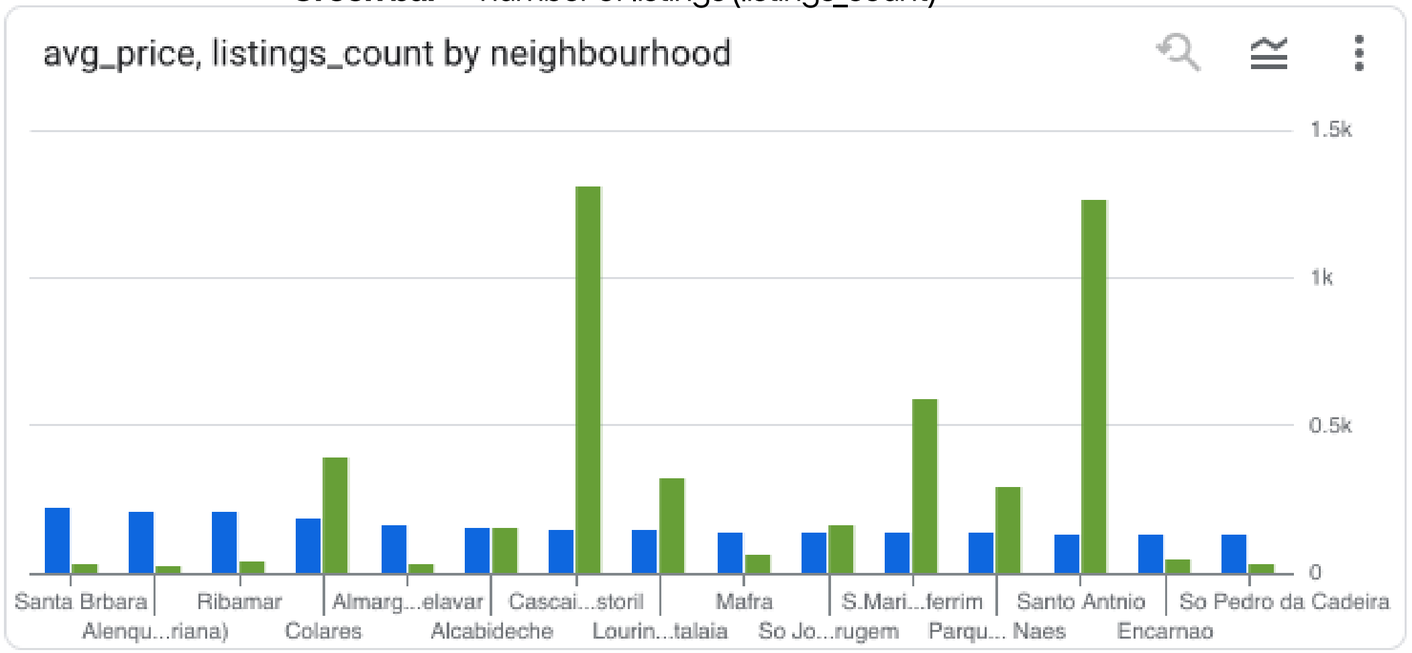
The development strategy could be adapted to support the most popular neighborhoods or include improvements such as tourist routes from less popular areas to points of interest in order to increase their attractiveness.

Question #3

What is the average rental price in each neighborhood of Lisbon?

```sql
SELECT
  neighbourhood,
  ROUND(AVG(price), 2) AS avg_price,
  COUNT(*) AS listings_count
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
WHERE price < 1000 -- Removing abnormally expensive listings
GROUP BY neighbourhood
HAVING listings_count >= 20 -- Selecting only neighborhoods with a sufficient number of listings
ORDER BY avg_price DESC
LIMIT 15;
```

Note:
Blue bar — average price (avg_price)
Green bar — number of listings (listings_count)



We observe that in most neighborhoods, price does not directly correlate with the number of listings:

- For example, in neighborhoods like **Santa Bárbara or Alenquer,** the average price is high **(€218.96 and €209.56)**, but **the number of listings is low (25–27)**.

→ This could be due to a generally low level of development in those areas, which pushes prices higher.

- In contrast, **Cascais e Estoril and Santo António** lead in terms of **number of listings (1,306 and 1,266),** yet have average prices around **€144.64 and €133.26.**

→ This might be explained by the larger geographical area covered by these neighborhoods.

- A clearer dynamic is seen in **Colares, Alcabideche, Loures, and Atalaia** — they show moderate **prices (€140–170)** and a sufficient **number of listings (150–400)**.

→ These areas could be of particular interest for business optimization, as they seem to have greater potential.

Question #4

What is the average housing price depending on room type?

```sql
SELECT
  room_type_cleaned,
  ROUND(AVG(price), 2) AS avg_price,
  COUNT(*) AS listings_count
FROM
  `my-test-project-445920.airbnb_lisbon.listings_cleaned_final`
GROUP BY
  room_type_cleaned
ORDER BY
  avg_price DESC;
```

Query results

Job information | Results | Chart | JSON | Execution deta

Row	room_type_cleaned	avg_price	listings_count
1	Hotel room	284.92	144
2	Entire home/apt	183.26	16224
3	Private room	60.98	4556
4	Shared room	32.03	90
5	Other	9.92	63

We can see that:

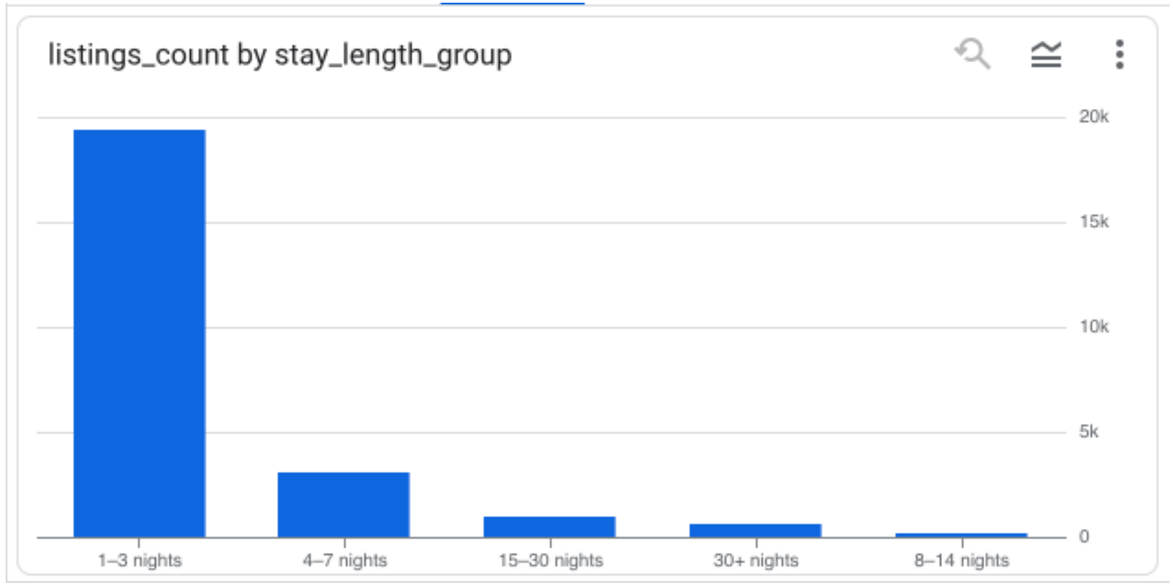
- **Hotel room** has the highest average price — approximately €285, but it is represented by only 144 listings.
- The most common type is **Entire home/apt** with around 16,200 listings and an average price of €183, making it a key market segment.
- **Private room** is a noticeably cheaper option (€61), but holds second place in popularity (~4,500 listings).

Based on this, it is already possible to develop marketing strategies or platform development initiatives in the Lisbon area.

Question #5

How long do guests typically stay in rented housing?

```sql
SELECT
  CASE
    WHEN minimum_nights BETWEEN 1 AND 3 THEN '1–3 nights'
    WHEN minimum_nights BETWEEN 4 AND 7 THEN '4–7 nights'
    WHEN minimum_nights BETWEEN 8 AND 14 THEN '8–14 nights'
    WHEN minimum_nights BETWEEN 15 AND 30 THEN '15–30 nights'
    ELSE '30+ nights'
  END AS stay_length_group,
  COUNT(*) AS listings_count
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
GROUP BY stay_length_group
ORDER BY listings_count DESC;
```



We can see that:

- Short stays **(1–3 days) are the most popular**.
- Medium stays (4–7 days) also take up a significant share.
- All other categories account for a much smaller portion.

It seems the product could improve the number of long-term stays by potentially creating a dedicated category and developing it further.

This information could also be useful for the marketing team to segment offerings and possibly introduce a new, user-friendly filter based on stay duration.

Question #6

How many listings does each host have?

```sql
SELECT
  host_id,
  COUNT(*) AS listings_per_host
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
GROUP BY host_id
ORDER BY listings_per_host DESC;
```

I initially ran a query and saw that there were 50 different results, which made the analysis difficult.
So I decided to slightly modify the query and group the hosts based on the number of listings.

Query results

Job information | Results | Chart | JSON

Row	host_group	host_count
1	1	6418
2	2–3	1849
3	4–5	564
4	6–10	475
5	11+	358

```sql
SELECT
  host_group,
  COUNT(*) AS host_count
FROM (
  SELECT
    host_id,
    MAX(calculated_host_listings_count) AS listings_count,
    CASE
      WHEN MAX(calculated_host_listings_count) = 1 THEN '1'
      WHEN MAX(calculated_host_listings_count) BETWEEN 2 AND 3 THEN '2–3'
      WHEN MAX(calculated_host_listings_count) BETWEEN 4 AND 5 THEN '4–5'
      WHEN MAX(calculated_host_listings_count) BETWEEN 6 AND 10 THEN '6–10'
      ELSE '11+'
    END AS host_group
  FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
  GROUP BY host_id
)
GROUP BY host_group
ORDER BY host_count DESC;
```

We observe that **the majority of hosts have only one property listed.**
→ This indicates that most users of the platform are private individuals. Hosts with 6–11+ listings represent a very small portion. These are likely businesses or professional hosting companies.

For the product, this insight could be used in the following ways:

- **Professional hosts** could be offered special loyalty programs or support.
- **Individual hosts** could benefit from easier onboarding (guides, templates, support).

Question #7

What is the average price depending on the number of listings a host has?

```sql
SELECT
  host_group,
  ROUND(AVG(price), 2) AS avg_price,
  COUNT(DISTINCT host_id) AS host_count
FROM (
  SELECT
    host_id,
    price,
    CASE
      WHEN calculated_host_listings_count = 1 THEN '1'
      WHEN calculated_host_listings_count BETWEEN 2 AND 3 THEN '2–3'
      WHEN calculated_host_listings_count BETWEEN 4 AND 5 THEN '4–5'
      WHEN calculated_host_listings_count BETWEEN 6 AND 10 THEN '6–10'
      ELSE '11+'
    END AS host_group
  FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
)
GROUP BY host_group
ORDER BY avg_price DESC;
```

Query results

Job information | Results | Chart | JSON | Execution detai

Row	host_group	avg_price	host_count
1	11+	181.59	357
2	6–10	175.48	475
3	1	142.18	6418
4	2–3	138.0	1849
5	4–5	118.2	564

We can see that **hosts with a greater number of apartments tend to set higher prices.**
Owners with only one apartment mostly have an average pricing level.

Here the business can decide which steps and host groups are worth supporting.
It's possible **to assist and educate hosts with a small number of listings** so they can improve their services and raise the quality and average price of their offers.
Or alternatively, **encourage and support hosts with a large number of listings.**

Question #8

Which types of housing receive more reviews?

```sql
SELECT
  room_type_cleaned,
  ROUND(AVG(reviews_per_month), 2) AS avg_reviews_per_month
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
WHERE reviews_per_month IS NOT NULL
GROUP BY room_type_cleaned
ORDER BY avg_reviews_per_month DESC;
```

Query results

Job information | Results | Chart | JSON

Row	room_type_cleaned	avg_reviews_per_mo
1	Other	3.19
2	Entire home/apt	1.56
3	Private room	1.24
4	Shared room	0.94
5	Hotel room	0.43

We observe that **Entire home/apt receives the most reviews** — but this is also the most popular type based on our first question.

Meanwhile, **Hotel room has the lowest number of reviews**. I believe this is related to the fact that hotels are more often booked through booking.com.

The business should consider whether it wants to attract more audience to hotel-type listings and possibly offer bonuses for leaving reviews, or whether this is not a target area of focus.

Question #9

Does the availability of housing (availability_365) differ by listing type?

```sql
SELECT
  room_type_cleaned,
  ROUND(AVG(availability_365), 2) AS avg_availability
FROM `my-test-project-445920.airbnb_lisbon.listings_
GROUP BY room_type_cleaned
ORDER BY avg_availability DESC;
```

Query results

Job information | Results | Chart | JSON

Row	room_type_cleaned	avg_availability
1	Shared room	197.87
2	Entire home/apt	192.67
3	Hotel room	183.72
4	Private room	169.93
5	Other	12.25

Based on this analysis, we see that **Shared room and Entire home/apt have the highest availability** — ~198 and ~193 days respectively.

Hotel room and Private room show slightly lower values (~184 and ~170 days).

The "Other" category is barely used — only ~12 days per year, which confirms its irrelevance.

We can conclude that **the most popular types of housing are actively available.**

Therefore, the platform can plan its marketing efforts over longer periods of time.

Question #10

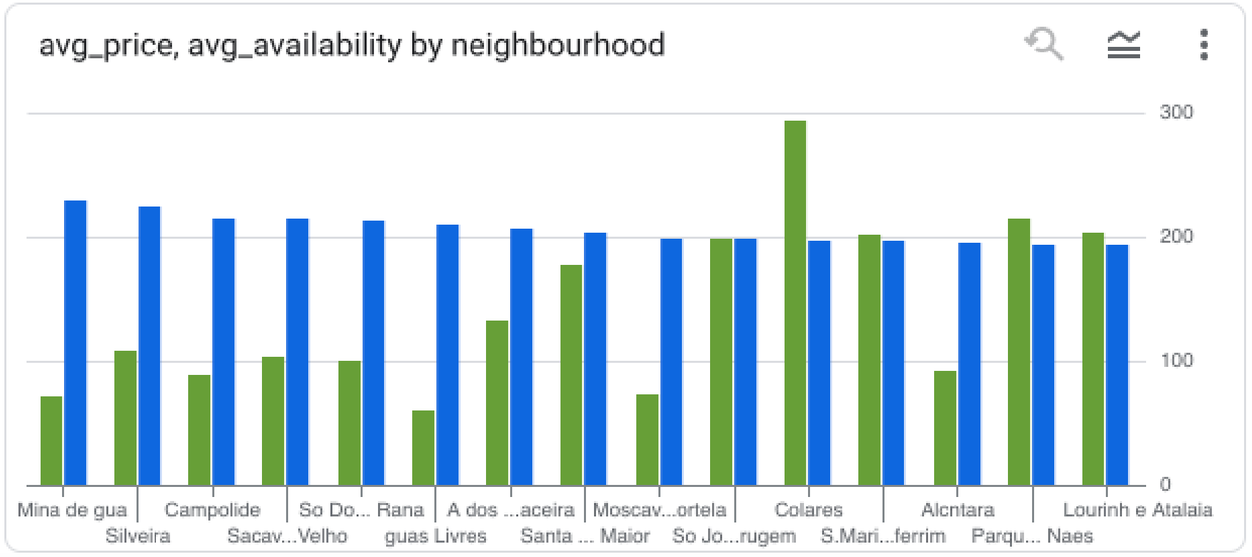
Which neighborhoods have the highest housing availability but relatively low average prices?

```sql
SELECT
  neighbourhood,
  ROUND(AVG(price), 2) AS avg_price,
  ROUND(AVG(availability_365), 2) AS avg_availability
FROM `my-test-project-445920.airbnb_lisbon.listings_cleaned`
GROUP BY neighbourhood
HAVING COUNT(*) > 50  -- Excluding low-popularity neighborhoods
ORDER BY avg_availability DESC, avg_price ASC
LIMIT 15;
```

Note:

Green color — avg_price

Blue color — avg_availability



Neighborhoods like Colares, Parque das Nações, and Alcântara show **high average rental prices with low availability.** This might indicate a limited number of bookable days or seasonality in those areas.

For the business, **this is a signal of potential to attract more hosts in these areas,** as demand may be exceeding supply.

On the other hand:

- **Mina d'Água** – High availability but one of the lowest average prices
- **Águas Livres** – Again, high availability at a moderate price
- **São Domingos de Rana** – A balance between high availability and low price

Airbnb could promote such areas as "affordable alternatives" — or even consider investing in property development in these neighborhoods directly.

4. Dashboard: Airbnb Lisbon Dashboard — Prices, Availability, Listings

You can explore the full interactive dashboard at the following link: https://public.tableau.com/

5. Conclusions

1. **Entire home/apt is the most popular type** of listing on the platform and dominates the market share.
2. **Prices vary significantly depending on the neighborhood:** premium areas (Colares, Alcântara) have the highest prices, while other areas appear undervalued.
3. **Hosts with a large number of listings** set higher prices and are likely operating at the level of small businesses.
4. **Housing availability is generally high (170–200 days per year)**, suggesting that hosts aim for steady income or that the tourist season in Portugal is long.
5. **Review activity is higher for private rooms and entire apartments**, possibly indicating stronger demand in these categories.

6. Recommendations

1. **Segment hosts based on the number of listings:**

- Individual hosts — offer educational campaigns and onboarding support
- Professional hosts — develop loyalty programs and pricing recommendations

2. **Reorganize marketing strategy by region:**

- Highlight undervalued neighborhoods with high availability and low prices
- If users browse listings but don't book, consider redirecting demand from oversaturated or expensive areas to more affordable ones

3. **Introduce smarter user recommendations:**

- Emphasize availability and listing activity more clearly

4. **Clean up the housing type categories (room_type)** — some listings don't contribute to meaningful analysis and were grouped into an "Other" category. These should either be removed or reformatted