# Low Level Design (LLD)


## AirBNB Travel Data Analysis

| | |
|---|---|
| Author | Anurag Gupta, Hemant Sharma |
| Document Version | LLD-V1.0 |
| Last Revised Date | 20/08/2023 |

# Document Version Control

| Date Issued | Version | Description | Author |
|---|---|---|---|
| **20/08/2023** | LLD-V1.0 | First Version of Complete LLD | Anurag Gupta,<br>Hemant Sharma |

# Contents

# Abstract

AirBNB began in 2008 when two designers who had space to share hosted three travelers looking for a place to stay. Now, millions of hosts and travelers choose to create a free AirBNB account so they can list their space and book unique accommodation anywhere in the world. In addition, AirBNB experience hosts share their passions and interests with both travelers and locals.

AirBNB helps make sharing easy, enjoyable, and safe. We verify personal profiles and listings, maintain a smart messaging system so hosts and guests can communicate with certainty, and manage a trusted platform to collect and transfer payments.

AirBNB has provided many travelers a great, easy and convenient place to stay during their travels. Similarly, it has also given an opportunity for many to earn extra revenue by listing their properties for residents to stay. However, with so many listings available with varying prices, how can an aspiring host know what type of property to invest in if his main aim is to list it in AirBNB and earn rental revenue? Additionally, if a traveler wants to find the cheapest listing available but with certain features, he prefers like 'free parking' etc., how does he know what aspects to look intoto find a suitable listing? There are many factors, which influence the price of a listing. Which is why we aim to find the most important factors that affect the price and more importantly the features that is common among the most expensive listings. This will allow an aspiring AirBNB hostto ensure that his listing is equipped with those important features such that he will be able to charge a higher price without losing customers. Moreover, a traveler will also know the factors to look into to get the lowest price possible while having certain features he prefers.

In the arena of rising new generation and innovation, Travel enterprise is advancing with the function of Data Science and Analytics. Data analysis can assist them to understand their businessin a quiet distinct way and helps to improve the exceptional of the provider by using identifying the vulnerable areas of the business. This examine demonstrates the how distinct analysis assist outto make higher business choices and help examine customer tendencies and pride, that may lead to new and higher products and services. Different evaluation accomplished along with ExploratoryData Analysis and Descriptive Analysis on type of use instances to get the important thing insights from these records primarily based on which enterprise decisions might be taken.

# 1 Introduction

## 1.1 Why this Low-Level design document?

The purpose of this LLD or a Low-Level Design (LLD) document is to give the internal logical design of the actual program code for Airbnb Data Analysis project. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document. This document is intended for both the stakeholders and the developers of this project and will be proposed to the higher management for its approval.

The main objective of the project is to analyze the various aspects with different use cases which covers many aspects of AirBNB listings. It helps in not only understanding the meaningful relationships between attributes but it also allows us to do our own research and come-up with our findings.

## 1.2 Scope

Low-level design (LLD) is a component-level design process that follows a step-by step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

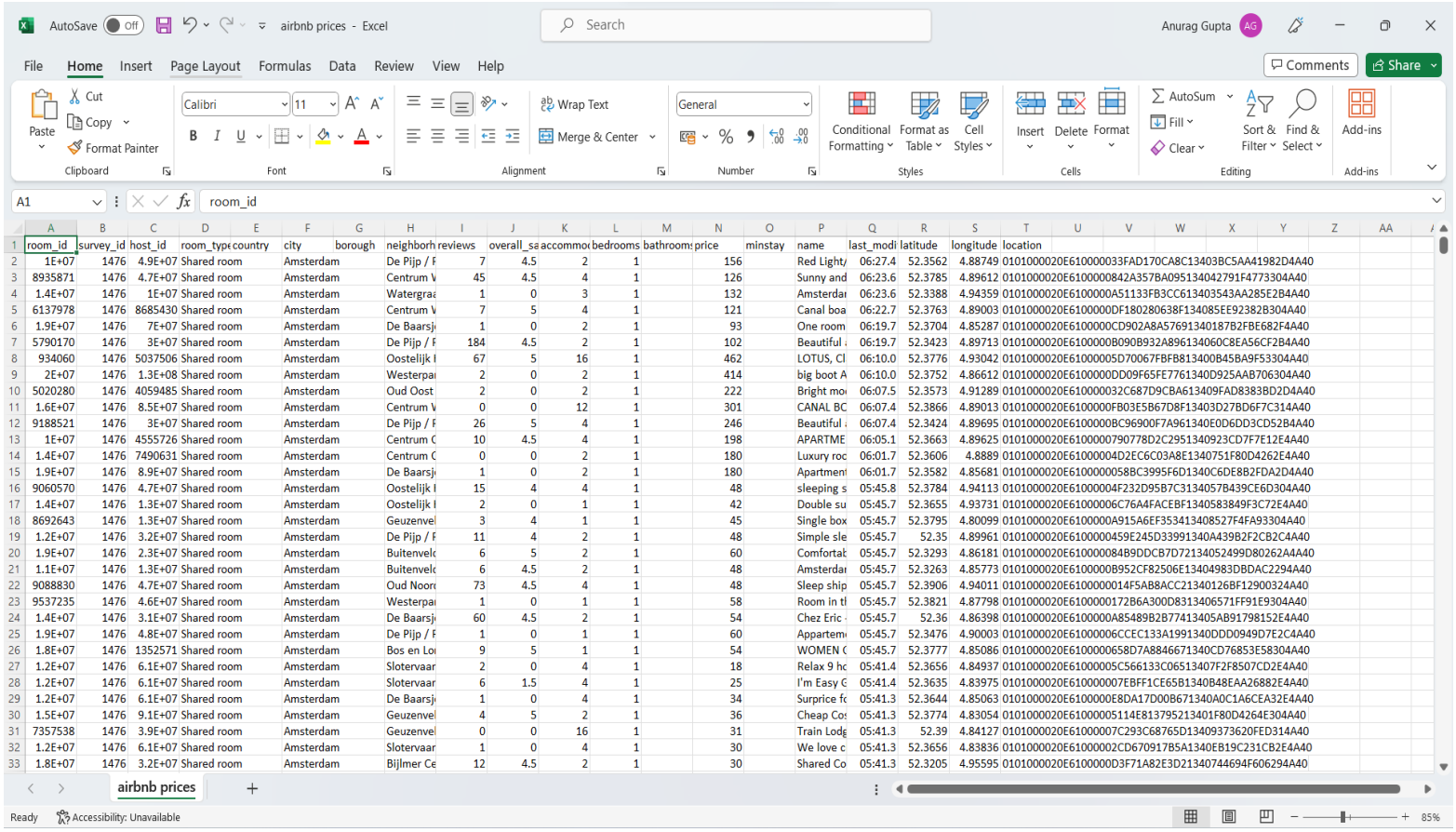
This study demonstrates the how different analysis help out to make better business decisions and help analysis customer trends and satisfaction, which can lead to new and better products and services.

## 1.3 Constraints

The analysis must be user-friendly, code must be neat & clean, EDA must be automated as much as possible because it will save huge amount of time. Moreover, users should not be required to have any of the coding knowledge as the insights they are looking for are mentioned in-detail with respective visuals.

# 2 Technical Specifications

## 2.1 Listings Dataset



| room_id | survey_id | host_id | room_type | country | city | borough | neighborh | reviews | overall_sa | accommo | bedrooms | bathrooms | price | minstay | name | last_modi | latitude | longitude | location |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1E+07 | 1476 | 4.9E+07 | Shared room | | Amsterdam | | De Pijp / F | 7 | 4.5 | 2 | 1 | | 156 | | Red Light/ | 06:27.4 | 52.3562 | 4.88749 | 0101000020E6100000033FAD170CA8C13403BC5AA41982D4A40 |
| 8935871 | 1476 | 4.7E+07 | Shared room | | Amsterdam | | Centrum W | 45 | 4.5 | 4 | 1 | | 126 | | Sunny and | 06:23.6 | 52.3785 | 4.89612 | 0101000020E6100000842A357BA095134042791F4773304A40 |
| 1.4E+07 | 1476 | 1E+07 | Shared room | | Amsterdam | | Watergraa | 1 | 0 | 3 | 1 | | 132 | | Amsterdar | 06:23.6 | 52.3388 | 4.94359 | 0101000020E6100000A51133FB3CC613403543AA285E2B4A40 |
| 6137978 | 1476 | 8685430 | Shared room | | Amsterdam | | Centrum W | 7 | 5 | 4 | 1 | | 121 | | Canal boa | 06:22.7 | 52.3763 | 4.89003 | 0101000020E6100000DF180280638F134085EE92382B304A40 |
| 1.9E+07 | 1476 | 7E+07 | Shared room | | Amsterdam | | De Baarsj | 1 | 0 | 2 | 1 | | 93 | | One room | 06:19.7 | 52.3704 | 4.85287 | 0101000020E6100000CD902A8A576913401087B2FBE682F4A40 |
| 5790170 | 1476 | 3E+07 | Shared room | | Amsterdam | | De Pijp / F | 184 | 4.5 | 2 | 1 | | 102 | | Beautiful | 06:19.7 | 52.3423 | 4.89713 | 0101000020E6100000B090B932A896134060C8EA56CF2B4A40 |
| 934060 | 1476 | 5037506 | Shared room | | Amsterdam | | Oostelijk I | 67 | 5 | 16 | 1 | | 462 | | LOTUS, Cl | 06:10.0 | 52.3776 | 4.93042 | 0101000020E6100000D70067FBFB813400B45BA9F53304A40 |
| 2E+07 | 1476 | 1.3E+08 | Shared room | | Amsterdam | | Westerpar | 2 | 0 | 2 | 1 | | 414 | | big boot A | 06:10.0 | 52.3752 | 4.86612 | 0101000020E6100000DD09F65FE7761340D925AAB706304A40 |
| 5020280 | 1476 | 4059485 | Shared room | | Amsterdam | | Oud Oost | 2 | 0 | 2 | 1 | | 222 | | Bright mo | 06:07.5 | 52.3573 | 4.91289 | 0101000020E6100000032C687D9CBA613409FAD8383BD2D4A40 |
| 1.6E+07 | 1476 | 8.5E+07 | Shared room | | Amsterdam | | Centrum W | 0 | 0 | 12 | 1 | | 301 | | CANAL BC | 06:07.4 | 52.3866 | 4.89013 | 0101000020E6100000FB03E5B67D8F13403D27BD6F7C314A40 |
| 9188521 | 1476 | 3E+07 | Shared room | | Amsterdam | | De Pijp / F | 26 | 5 | 4 | 1 | | 246 | | Beautiful | 06:07.4 | 52.3424 | 4.89695 | 0101000020E6100000BC96900F7A961340E0D6DD3CD52B4A40 |
| 1E+07 | 1476 | 4555726 | Shared room | | Amsterdam | | Centrum C | 10 | 4.5 | 4 | 1 | | 198 | | APARTME | 06:05.1 | 52.3663 | 4.89625 | 0101000020E6100000790778D2C2951340923CD7F7E12E4A40 |
| 1.4E+07 | 1476 | 7490631 | Shared room | | Amsterdam | | Centrum C | 0 | 0 | 2 | 1 | | 180 | | Luxury roc | 06:01.7 | 52.3606 | 4.8889 | 0101000020E61000004D2EC6C03A8E1340751F80D4262E4A40 |
| 1.9E+07 | 1476 | 8.9E+07 | Shared room | | Amsterdam | | De Baarsj | 1 | 0 | 2 | 1 | | 180 | | Apartment | 06:01.7 | 52.3582 | 4.85681 | 0101000020E6100000058BC3995F6D1340C6DE8B2FDA2D4A40 |
| 9060570 | 1476 | 4.7E+07 | Shared room | | Amsterdam | | Oostelijk I | 15 | 4 | 4 | 1 | | 48 | | sleeping s | 05:45.8 | 52.3784 | 4.94113 | 0101000020E61000004F232D95B7C3134057B439CE6D304A40 |
| 1.4E+07 | 1476 | 1.3E+07 | Shared room | | Amsterdam | | Oostelijk I | 2 | 0 | 1 | 1 | | 42 | | Double su | 05:45.7 | 52.3655 | 4.93731 | 0101000020E61000006C76A4FACEBF1340583849F3C72E4A40 |
| 8692643 | 1476 | 1.3E+07 | Shared room | | Amsterdam | | Geuzenvel | 3 | 4 | 1 | 1 | | 45 | | Single box | 05:45.7 | 52.3795 | 4.80099 | 0101000020E6100000A915A6EF353413408527F4FA93304A40 |
| 1.2E+07 | 1476 | 3.2E+07 | Shared room | | Amsterdam | | De Pijp / F | 11 | 4 | 2 | 1 | | 48 | | Simple sle | 05:45.7 | 52.35 | 4.89961 | 0101000020E6100000459E245D33991340A439B2F2CB2C4A40 |
| 1.9E+07 | 1476 | 2.3E+07 | Shared room | | Amsterdam | | Buitenvelc | 6 | 5 | 2 | 1 | | 60 | | Comfortak | 05:45.7 | 52.3293 | 4.86181 | 0101000020E61000008 4B9DDCB7D72134052499D80262A4A40 |
| 1.1E+07 | 1476 | 1.3E+07 | Shared room | | Amsterdam | | Buitenvelc | 6 | 4.5 | 2 | 1 | | 48 | | Amsterdar | 05:45.7 | 52.3263 | 4.85773 | 0101000020E6100000B952CF82506E13404983DBDAC2294A40 |
| 9088830 | 1476 | 4.7E+07 | Shared room | | Amsterdam | | Oud Noord | 73 | 4.5 | 4 | 1 | | 48 | | Sleep ship | 05:45.7 | 52.3906 | 4.94011 | 0101000020E6100000014F5AB8ACC21340126BF12900324A40 |
| 9537235 | 1476 | 4.6E+07 | Shared room | | Amsterdam | | Westerpar | 1 | 0 | 1 | 1 | | 58 | | Room in tl | 05:45.7 | 52.3821 | 4.87798 | 0101000020E6100000172B6A300D8313406571FF91E9304A40 |
| 1.4E+07 | 1476 | 3.1E+07 | Shared room | | Amsterdam | | De Baarsj | 60 | 4.5 | 2 | 1 | | 54 | | Chez Eric | 05:45.7 | 52.36 | 4.86398 | 0101000020E6100000A85489B2B77413405AB91798152E4A40 |
| 1.9E+07 | 1476 | 4.8E+07 | Shared room | | Amsterdam | | De Pijp / F | 1 | 0 | 1 | 1 | | 60 | | Appartem | 05:45.7 | 52.3476 | 4.90003 | 0101000020E61000006CCEC133A1991340DDD0949D7E2C4A40 |
| 1.8E+07 | 1476 | 1352571 | Shared room | | Amsterdam | | Bos en Lo | 9 | 5 | 1 | 1 | | 54 | | WOMEN ( | 05:45.7 | 52.3777 | 4.85086 | 0101000020E6100000658D7A8846671340CD76853E58304A40 |
| 1.2E+07 | 1476 | 6.1E+07 | Shared room | | Amsterdam | | Slotervaar | 2 | 0 | 4 | 1 | | 18 | | Relax 9 hc | 05:41.4 | 52.3656 | 4.84937 | 0101000020E6100000C566133C06513407F2F8507CD2E4A40 |
| 1.2E+07 | 1476 | 6.1E+07 | Shared room | | Amsterdam | | Slotervaar | 6 | 1.5 | 4 | 1 | | 25 | | I'm Easy G | 05:41.4 | 52.3635 | 4.83975 | 0101000020E61000000 7EBFF1CE65B1340B48EAA26882E4A40 |
| 1.2E+07 | 1476 | 6.1E+07 | Shared room | | Amsterdam | | De Baarsj | 1 | 0 | 4 | 1 | | 34 | | Surprice fc | 05:41.3 | 52.3644 | 4.85063 | 0101000020E6100000E8DA17D00B671340A0C1A6CEA32E4A40 |
| 1.5E+07 | 1476 | 9.1E+07 | Shared room | | Amsterdam | | Geuzenvel | 4 | 5 | 2 | 1 | | 36 | | Cheap Cos | 05:41.3 | 52.3774 | 4.83054 | 0101000020E6100000 5114E81379521340 1F80D4264E304A40 |
| 7357538 | 1476 | 3.9E+07 | Shared room | | Amsterdam | | Geuzenvel | 0 | 0 | 16 | 1 | | 31 | | Train Lodg | 05:41.3 | 52.39 | 4.84127 | 0101000020E61000007C293C68765D1340937362 0FED314A40 |
| 1.2E+07 | 1476 | 6.1E+07 | Shared room | | Amsterdam | | Slotervaar | 1 | 0 | 4 | 1 | | 30 | | We love c | 05:41.3 | 52.3656 | 4.83836 | 0101000020E61000002CD670917B5A1340EB19C231CB2E4A40 |
| 1.8E+07 | 1476 | 3.2E+07 | Shared room | | Amsterdam | | Bijlmer Ce | 12 | 4.5 | 2 | 1 | | 30 | | Shared Co | 05:41.3 | 52.3205 | 4.95595 | 0101000020E6100000D3F71A82E3D21340744694F606294A40 |

## 2.2 Listings Dataset Overview –

The Listings dataset consists of a table with 18722 records and 20 features. Features are distributed as 11 Continuous features and 9 Categorical features. There are a total 20% of records having Missing values.

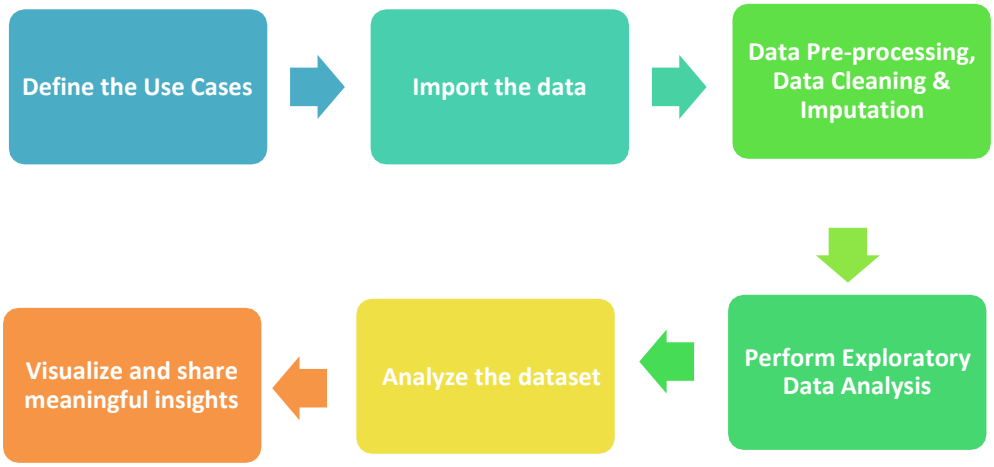
```
In [4]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18723 entries, 0 to 18722
Data columns (total 20 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   room_id               18723 non-null  int64
 1   survey_id             18723 non-null  int64
 2   host_id               18723 non-null  int64
 3   room_type             18723 non-null  object
 4   country               0 non-null      float64
 5   city                  18723 non-null  object
 6   borough               0 non-null      float64
 7   neighborhood          18723 non-null  object
 8   reviews               18723 non-null  int64
 9   overall_satisfaction  18723 non-null  float64
 10  accommodates          18723 non-null  int64
 11  bedrooms              18723 non-null  float64
 12  bathrooms             0 non-null      float64
 13  price                 18723 non-null  float64
 14  minstay               0 non-null      float64
 15  name                  18671 non-null  object
 16  last_modified         18723 non-null  object
 17  latitude              18723 non-null  float64
 18  longitude             18723 non-null  float64
 19  location              18723 non-null  object
dtypes: float64(9), int64(5), object(6)
memory usage: 2.9+ MB
```

```
In [6]: df.isnull().sum()

Out[6]: room_id                    0
        survey_id                  0
        host_id                    0
        room_type                  0
        country                18723
        city                       0
        borough                18723
        neighborhood               0
        reviews                    0
        overall_satisfaction       0
        accommodates               0
        bedrooms                   0
        bathrooms              18723
        price                      0
        minstay                18723
        name                      52
        last_modified              0
        latitude                   0
        longitude                  0
        location                   0
        dtype: int64
```

# 3 Architecture



# 3.1 Architecture Description –

## 3.1.1 Data Description –

As we have seen earlier, in our listings dataset, we have around 1.19 Lacs of records with 20 different features. Features are distributed as 10 Continuous features and 10 Categorical features and in our reviews dataset, we have around 3.44 Lacs of records with 6 different features among them there are 3 Continuous features and 3 Categorical features. These datasets are given in the form of Comma Separated Value (.csv) format.

## 3.1.2 Define the Use Cases –

At this stage, based on the given dataset and business problems we have defined the several Use Cases to perform the analysis on and this will definitely help out get the key insights from this data based on which business decisions will be taken. Furthermore, it helps in not only understanding the meaningful relationships between attributes but it also allows us to do our own research and come-up with our findings.

## 3.1.3 Import the Dataset –

As we have received the dataset in the form of Comma Separated Value (.csv) format, therefore we can import the same using Pandas read_csv( ) function.

In [2]:
```
#loading dataset
data = pd.read_csv("airbnb prices.csv")
df = data.copy()
df.head()
```

Out[2]:

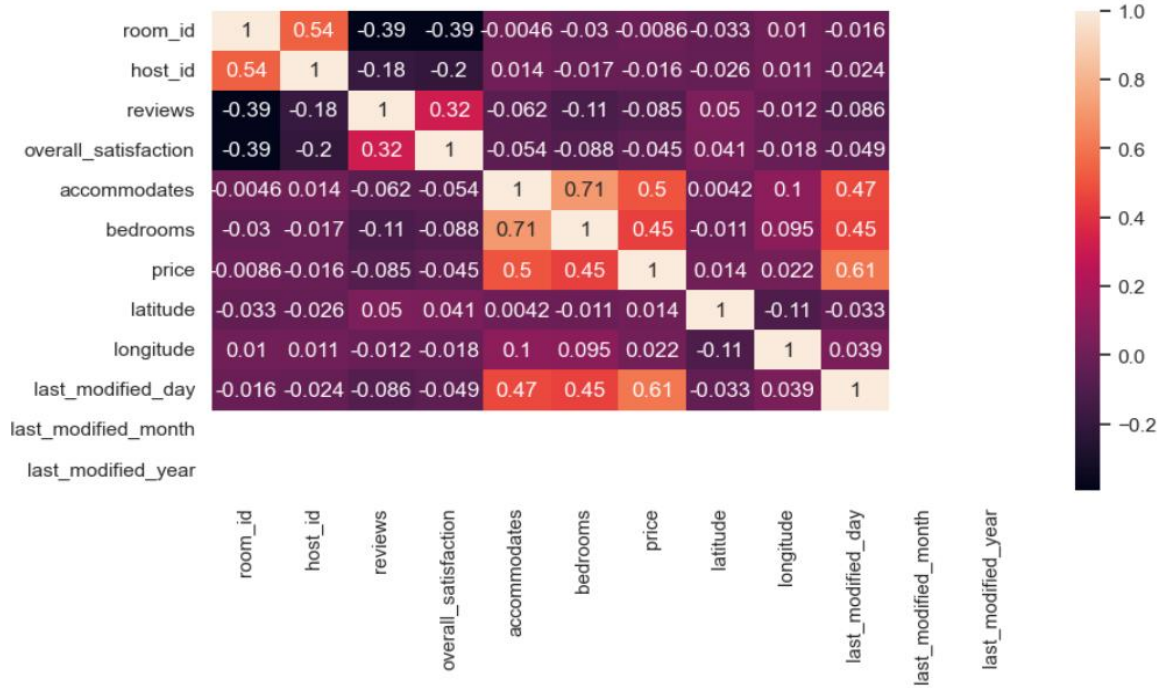
| | room_id | survey_id | host_id | room_type | country | city | borough | neighborhood | reviews | overall_satisfaction | accommodates | bedrooms | bathroor |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 10176931 | 1476 | 49180562 | Shared room | NaN | Amsterdam | NaN | De Pijp / Rivierenbuurt | 7 | 4.5 | 2 | 1.0 | Na |
| 1 | 8935871 | 1476 | 46718394 | Shared room | NaN | Amsterdam | NaN | Centrum West | 45 | 4.5 | 4 | 1.0 | Na |
| 2 | 14011697 | 1476 | 10346595 | Shared room | NaN | Amsterdam | NaN | Watergraafsmeer | 1 | 0.0 | 3 | 1.0 | Na |
| 3 | 6137978 | 1476 | 8685430 | Shared room | NaN | Amsterdam | NaN | Centrum West | 7 | 5.0 | 4 | 1.0 | Na |
| 4 | 18630616 | 1476 | 70191803 | Shared room | NaN | Amsterdam | NaN | De Baarsjes / Oud West | 1 | 0.0 | 2 | 1.0 | Na |

### 3.1.4 Exploratory Data Analysis (EDA)–

- "Exploratory Data Analysis" (EDA) is a "Data Exploration" step in the Data Analysis Process where a number of techniques are used to better understand the dataset being used.
- Understanding the Dataset can refer to a number of things including but not limited to…
  - Extracting Important "Variables".
  - Identifying "Outliers", "Missing Values", or "Human Error".
  - Understanding the Relationships between variables.
  - Ultimately, maximizing our insights of a dataset and minimizing potential "Error" that may occur later in the process.
- In other words, it will give you a better Understanding of the "Variables" and the "Relationships" between them.
- Here, we make use of data preprocessing module to automate our EDA process.
- It provides the following information:
  - Overview: detect the types of columns in a DataFrame.
  - Variables: variable type, unique values, distinct count, missing values
  - Quartile statistics like minimum value, Q1, median, Q3, maximum, range, interquartile range
  - Descriptive statistics like mean, mode, standard deviation, sum, median absolute deviation, coefficient of variation, kurtosis, skewness.
  - Correlations: highlighting of highly correlated variables, Spearman, Pearson and Kendall matrices
  - Missing Values: Bar Chart, Heatmap and spectrum of missing values.

- Below we can visualize correlation with Heatmap:

```
In [62]: #heatmap for correlation
         plt.figure(figsize=(10,5))
         sns.heatmap(df_cleaned.corr(),annot=True)
Out[62]: <Axes: >
```



### 3.1.5 Data Pre-processing, Data Cleaning & Imputation (Handling the Categorical & Numerical Variables) –

Data pre-processing is a process of preparing the raw data and making it suitable for our analysis purpose, where we have to do lot of Data Cleaning, handle the missing values by using appropriate imputation techniques and based on that variable nature i.e. either of Categorical & Numerical variable. Here, in this project, we have done the substitution/imputation of missing values using either mean, median or mode according to the nature of those variables. Moreover, we also removed the columns which are does not participate in our analysis.

### 3.1.6 Analyze the Data –

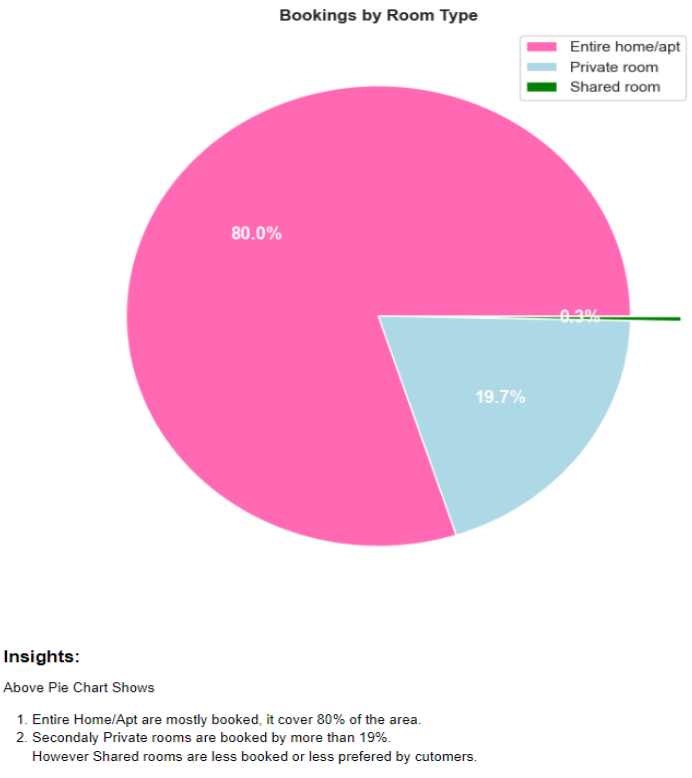
Once the pre-processing is done, we are good to go with our actual analysis where we write lines of codes and logics to prepare our data as per the defined use cases.

### 3.1.7 Visualize & Share Meaningful Insights –

Finally, it is time to turn our data into some sort of visual representation. In short, Data visualization is the process of translating large data sets and metrics into charts, graphs and other visuals such as Bar Plot, Pie Chart, Heat map, Box Plot, Scatter Plot, and many more. The resulting visual representation of data makes it easier to identify and share insights about the information represented in the data.

Here is the beautiful glimpse of one of our visuals are –

>> Which Room Type is Mostly liked?




All those different analysis help out to make better business decisions and help analyze customer trends and satisfaction, which can lead to new and better products and services.

# 4 Technology Stack

| | |
|---|---|
| **Data Manipulation & Mathematical Computation Library** | Pandas, NumPy |
| **Visualization Library** | Matplotlib, Seaborn, Plotly |
| **Dataset** | .CSV Format |
| **IDE** | Jupyter Notebook |