

## **Housing SQL Project Documentation**

### **Web Scrapping From <https://housing.com>:**

Tools used: 1) Google Colaboratory

Language: 1) Python Language

Python Libraries used: Pandas, Numpy, re, urllib, bs4, requests, ssl, tqdm

Scrapped the data from 5 locations

- 1) Bangalore
- 2) Hyderabad
- 3) Chennai
- 4) Pune
- 5) Mumbai

#### **For Pune:**

Execute the pune.ipynb file in the google colab. After execution completed, all scrapped data will be saved in pune.csv file. Download the csv file by using the download option, after downloaded, the file will be saved in downloads in system.

#### **For Chennai:**

Execute the chennai.ipynb file in the google colab. After execution completed, all scrapped data will be saved in chennai.csv file. Download the csv file by using the download option, after downloaded, the file will be saved in downloads in system.

#### **For Hyderabad:**

Execute the Hyderabad.ipynb file in the google colab. After execution completed, all scrapped data will be saved in Hyderabad.csv file. Download the csv file by using the download option, after downloaded, the file will be saved in downloads in system.

#### **For Bangalore:**

Execute the Bangalore.ipynb file in the google colab. After execution completed, all scrapped data will be saved in Bangalore.csv file. Download the csv file by

using the download option, after downloaded, the file will be saved in downloads in system.

### **For Mumbai:**

Upload the pune.csv, chennai.csv, Hyderabad.csv, Bangalore.csv files by using upload button in mumbai.ipynb in google colab.(Merging the data from multiple csv files into one csv file)

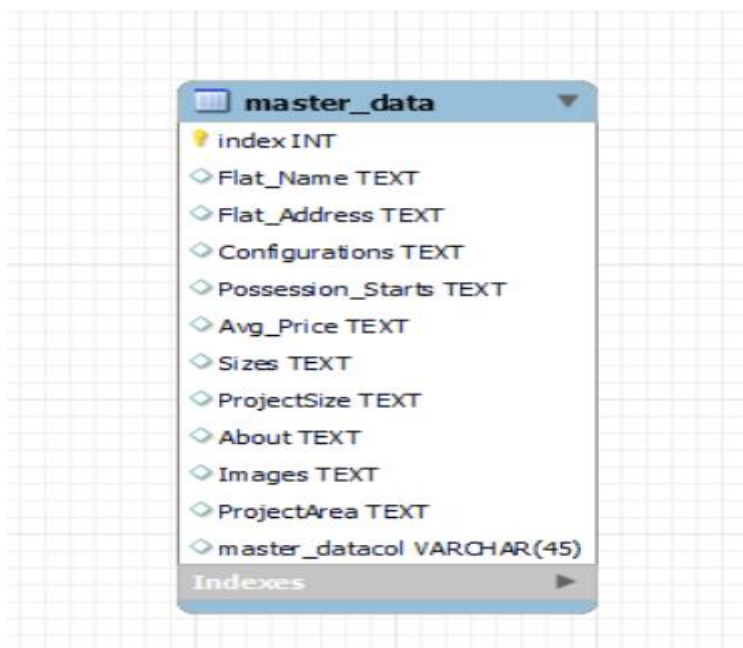
Execute the mumbai.ipynb file in the google colab. After execution completed, all scrapped data will be saved in mumbai.csv file.

Merging the data into one csv file named Final\_Data.csv.

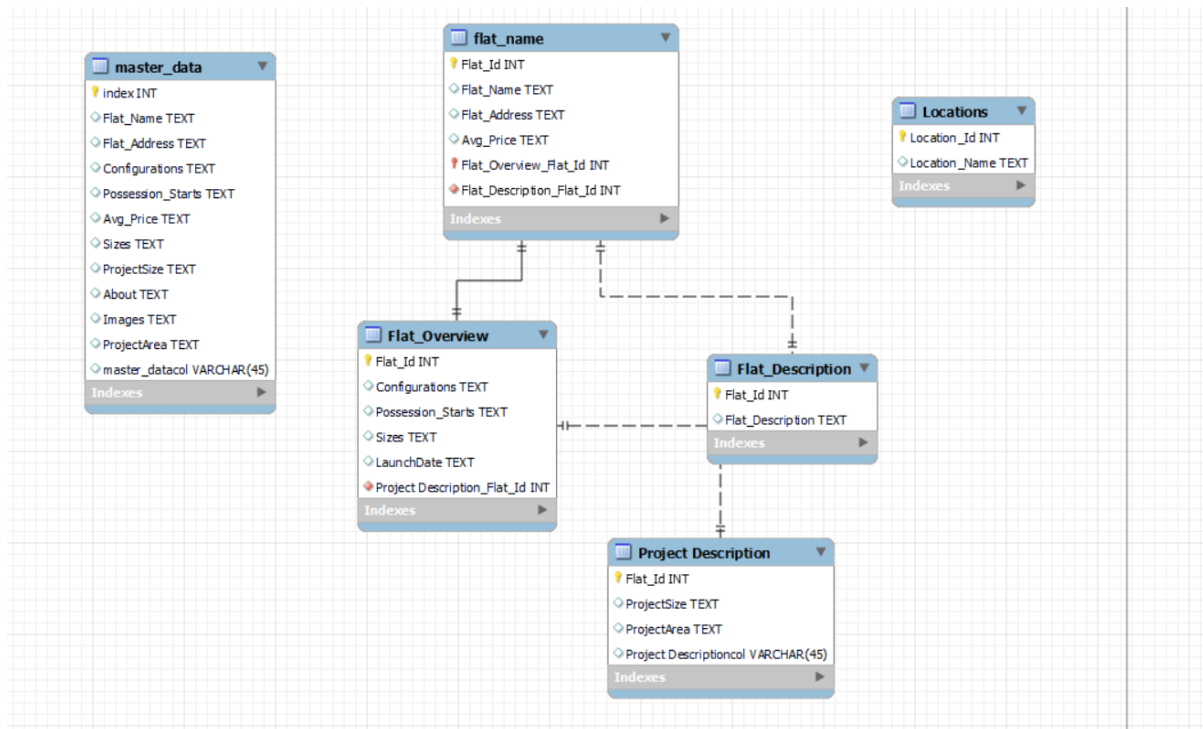
Download the Final\_Data.csv file by using the download option, after downloaded, the file will be saved in downloads in system.

### **DataBase Design:**

#### **Master Data Table Design:**



#### **Flat\_Names Table Design:**



## **DataBase Creation:**

Tools used: 1) MySQL Workbench

2) Command Prompt

3) Notepad++

Open the MySQL Workbench and launch the Local instance MySQL80, open the sql console.

Execute the below the sql query to create a database name housing.

**create database housing;**

Execute the below the sql query to use the database name housing.

**use housing;**

Open the Notepad++, create the **Master\_table\_Creation.py** file and save it documents folder in your system.

In the **Master\_table\_Creation.py** file, Connection of mysql server to python code is there and creation of master data table that has to be created in housing database.

Open the command prompt:

Run the command: **C:\Users\ashok> cd Documents**

It will connect to Documents folder to the system.

Run the command: **C:\Users\ashok\Documents> python Master\_table\_Creation.py**

After executing the command, master data table will be created in housing database.

Open another command prompt:

Run the command: **sql --local\_infile -u root -p**

**C:\Users\ashok> sql --local\_infile -u root -p**

Enter the password: **\*\*\*\*\*** (Give the password that when installing the mysql server 8.0 in your system)

After giving the password, it will connect to mysql server.

Execute the **show databases;** syntax

**mysql> show databases;**

It will show all the databases that we have in SQL server. housing database will also be visible.

**mysql> use housing;**

It will connect to housing database.

Copy the Final\_Data.csv file and paste it in the C:\Program Files\MySQL\MySQL Server 8.0\uploads\

**mysql> load data local infile 'C:\\Program Files\\MySQL\\MySQL Server 8.0\\uploads\\Final\_Data.csv'**

- into table master\_data**
- fields terminated by ','**
- enclosed by ''''**
- lines terminated by '\n'**
- ignore 1 rows;**

after executed the above command, the data in Final\_Data.csv is imported into master data table in housing database.

Open the MySQL Workbench, Create the stored procedures for creation of tables and insertion of tables.

**Creation\_Tables\_Procedure.sql** -> Create all the sql queries for creating tables and it has to be stored on one Procedure.

Five tables (sql queries created) : flat\_names, flat\_overview, flat\_description, locations and project description.

**Creation\_Flat\_Overview\_Procedure.sql** -> Create all the insert queries for inserting the data and it has to be stored on one Procedure.

Open the Notepad++, create the **Creation\_and\_Insertion.py** file and write the code to connect to database and for calling the procedures and save it Documents folder in the system.

Open the command prompt, Run the python file.

**C:\Users\ashok\Documents> python Creation\_and\_Insertion.py**

After executed the py file, all the tables and the data will be inserted into the housing database.

### **Search by Location, Configurations or Prices:**

Open the MySQL WorkBench,

**Search\_by\_Location\_procedure.sql** -> Create the procedure for “if the location is matched by user input, the flat names and addresses will be visible for that particular location”.

**Configurations\_Procedure.sql** -> Create the procedure for “if the configurations is matched by user input, the flat names and addresses will be visible for that particular configurations”.

**choosing\_price.sql** -> Create the procedure for “if the price is matched by user input, the flat names and addresses and its price will be visible for that particular price”.

Open the Notepad++, create the **Procedure2.py** file for getting the data search by location, configurations or Prices by calling the procedures and save it Document Folder.

Open the command Prompt, execute the python file.

**C:\Users\ashok\Documents> python Procedure2.py**

After executing the py file, It will ask questions:

### **Materialized Views:**

**Sql\_view1.sql** -> Create the view to view the full details of flat names and their Features.

**SQL\_view1.sql** -> Create the view to store in the procedure to execute in the py file.

### **Indexes:**

**Sql\_Index1.sql** -> created a index for the Flat\_Id for the flat\_names table.

**Sql\_index2.sql** -> created a index for the Flat\_Id for the flat\_overview table.

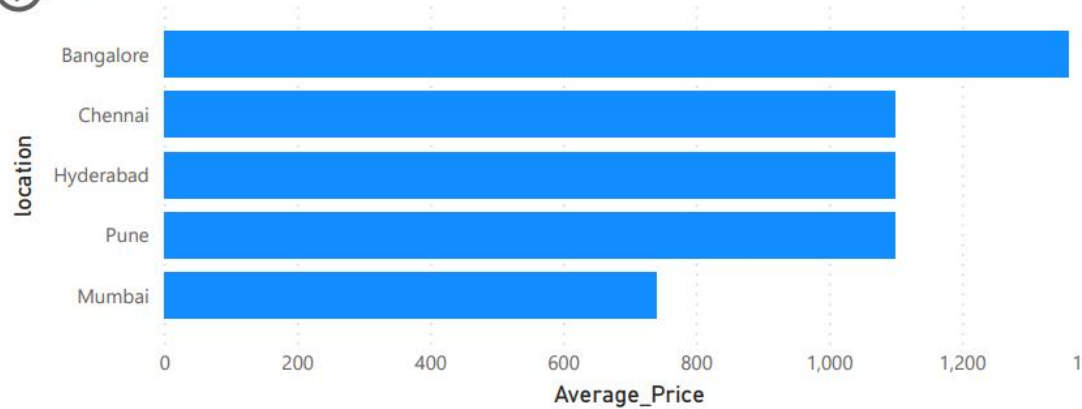
### **Triggers:**

**Flat\_Overview\_Trigger** -> Created a trigger before update on flatoverview table to combine the Possession starts and Launch Date of a flat name.

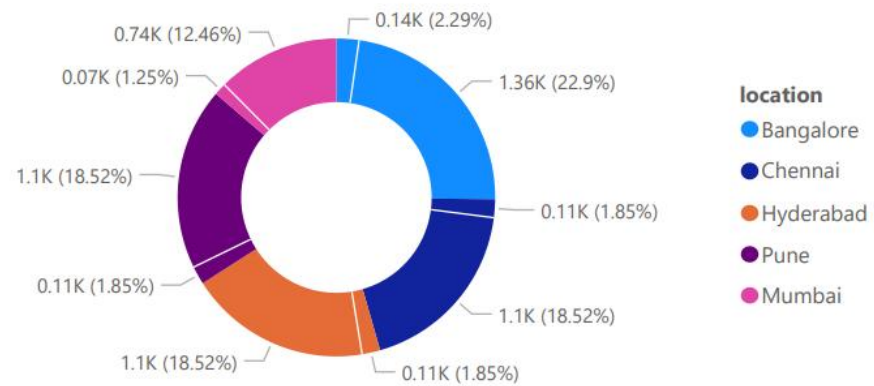
### **Performance Benefits of the Index with compared to Table:**

- 1) Indexes are special lookup tables that the database search engine can use to speed data retrieval.
- 2) An index is a pointer to data in the table.
- 3) Indexes are different types. Single Column indexes, Unique Column indexes, Composite indexes.
- 4) Unique indexes are used not only for performance, but also for data integrity.
- 5) A Unique index does not allow duplicate values to be inserted.

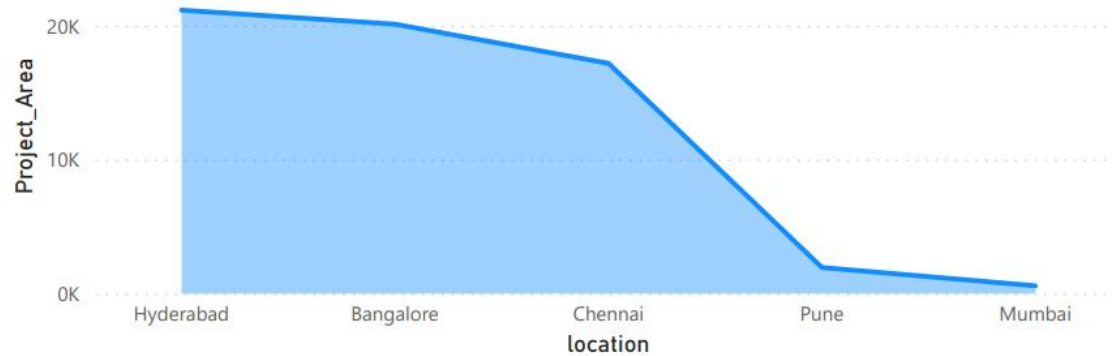
 Average\_Price by location



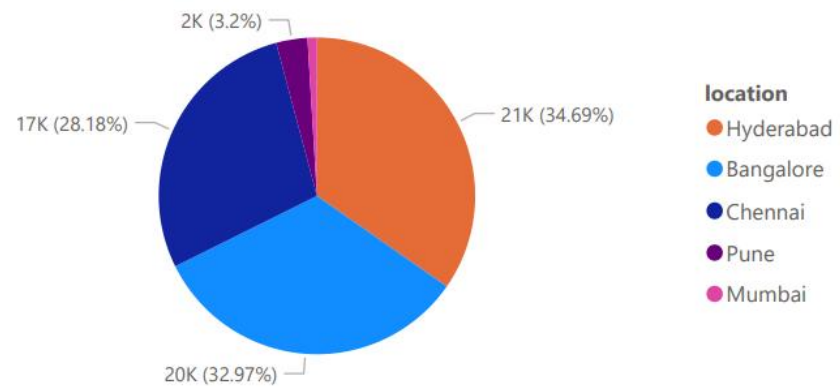
Count of location and Average\_Price by location



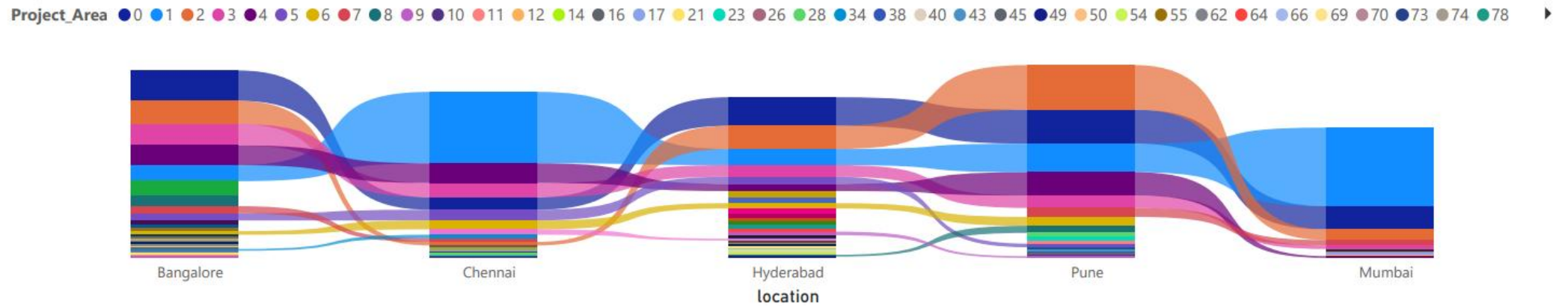
Project\_Area by location



Project\_Area by location



Average\_Price by location and Project\_Area



Average\_Price by location and Project\_Area

