Project Documentation: Luxury Housing Bangalore

**Step 1: Load Raw Dataset**

* File name: **Luxury\_Housing\_Bangalore.csv**
* Total rows in raw file: **~101,000**
* *Reason:* First step in any project is to bring the raw data into Python (using Pandas). Only after loading, we can clean and process it.

**Step 2: Python — Data Cleaning & Feature Engineering**

We started with the raw file Luxury\_Housing\_Bangalore.csv (around 101,000 rows). Our goal was to prepare a clean dataset ready for database insertion and analysis.

1. **Remove Duplicates**
   * About 1,000 duplicate rows were found.
   * These were deleted, leaving 100,000 unique rows.
   * *Reason:* Duplicates give wrong results like inflated counts or revenue.
2. **Drop Unwanted Columns**
   * Columns such as *Property\_ID* and *Buyer\_Comments* were removed.
   * *Reason:* These fields are not useful for analysis.
3. **Fix Inconsistent Formats**
   * Price column (*Ticket\_Price\_Cr*) had symbols like “₹” and “Cr”. These were removed, leaving only numeric values.
   * Spelling mistakes in *Configuration* and *Micro\_Market* were corrected.
   * *Reason:* Clean formats are required for calculations and avoiding duplicate categories.
4. **Handle Missing Values**
   * *Unit\_Size\_Sqft:* Missing values filled with **median size** grouped by configuration.
   * *Ticket\_Price\_Cr:* Missing values filled with **mean price** grouped by configuration.
   * *Amenity\_Score:* Missing values filled with **median score** grouped by developer.
   * *Reason:* Null values disturb analysis. Filling with grouped median/mean keeps values realistic.
5. **Change Data Types**
   * *Purchase\_Quarter* converted to proper datetime format.
   * *Reason:* Date columns must be in datetime type to extract year, quarter, and trends.

**Feature Engineering (New Columns)**

We created new columns to help in analysis:

* **Price\_per\_Sqft** → Price per square foot. (Helps compare properties fairly.)
* **Quarter\_Number** → Extracted quarter (1–4) from purchase date. (Helps check seasonal trends.)
* **Booking\_Status** → 1 if booked (Primary), 0 if resale. (Helps measure booking success.)
* **Year** → Extracted purchase year. (For yearly trend analysis.)
* **Year\_Quarter** → Combined Year + Quarter (e.g., 2023Q2). (For quarterly trend analysis.)

**Output**

A cleaned dataset with consistent values, no duplicates, no nulls, and additional useful columns. Saved as a new CSV for further use.

**Step 3: SQL — Load Clean Data into PostgreSQL**

After cleaning, we moved the dataset into a SQL database for structured storage and analysis.

1. **Create Table Schema**
   * Defined column names and proper data types (e.g., numeric, text, date).
   * *Reason:* Ensures clean and structured storage in PostgreSQL.
2. **Insert Data**
   * Loaded the cleaned dataset into PostgreSQL using Python.
   * *Reason:* SQL is efficient for large datasets and integrates with BI tools like Power BI.
3. **Validation Queries**
   * Checked total row count (100,000).
   * Grouped by booking status to confirm Primary vs Resale counts.
   * Calculated average ticket price per developer.

**Output**

Data successfully stored in PostgreSQL, validated with queries, and ready for dashboards/insights.

**Step 4: Power BI — Connect to PostgreSQL and Load Data**

1. Opened Power BI Desktop.

2. Went to Get Data and selected PostgreSQL Database.

3. Entered server name, database name, username, and password.

4. Selected the Luxury\_House table (inserted in Step 2).

5. Loaded the data into Power BI successfully.

**Output**

The Luxury\_House dataset from PostgreSQL is now successfully loaded into Power BI.

**Step 5: Power BI — Visualization and Insights**

**1. Market Trends**

**How we built it:**

* Added Line Chart.
* X-axis: Quarter\_Number or Year\_Quarter.
* Y-axis: Booking Count (count of Booking\_Status).
* Legend: Micro\_Market.

**Insight:**

* Some micro-markets show steady growth in bookings, while others fluctuate.
* Helps identify booming areas quarter by quarter.

**2. Builder Performance**

**How we built it:**

• Added Bar Chart

• X-axis: Sum(Ticket\_Price\_Cr), Avg(Ticket\_Price\_Cr).

• Y-axis: Developer\_Name

**Insight:**

* Shows which builders have the highest sales revenue.
* Comparison of revenue vs average ticket size highlights premium vs mid-range builders.

**3. Amenity Impact**

**How we built it:**

• Added Scatter Plot.

• X-axis: Amenity\_Score.

• Y-axis: Booking Conversion Rate (successful bookings).

• Legend: Developer\_Name

• Bubble Size: Project Count.

**DAX Calculation:**

Project\_Count = DISTINCTCOUNT('public luxury\_house\_data'[Project\_Name])

**Insight:**

* Higher amenity scores generally improve booking chances.
* Few exceptions highlight cases where price or location dominates.

**4. Booking Conversion by Micro-Market**

**How we built it:**

• Added Stacked Column Chart.

• X-axis: Count of Booking Status

• Y-axis: Micro\_Market.

• Legend: Booking\_Status

**Insight:**

* Some micro-markets have high conversion rates (buyers finalize deals quickly).
* Others show low conversion due to high prices or poor infrastructure.

**5. Configuration Demand**

**How we built it:**

• Added Pie Chart.

• Legend: Configuration

• Values: Booking Count.

**Insight:**

* 3BHK and 4BHK dominate demand.
* Niche demand for 2BHK (budget buyers) and 5BHK (luxury premium segment).

**6. Sales Channel Efficiency**

**How we built it:**

• Added 100% Stacked Column Chart.

• Y-axis: Sales\_Channel.

• X-axis: count of Booking\_Status

• Legend: Booking\_Status

**Insight:**

* Certain channels (e.g., Online/Direct Sales) perform better than brokers.
* Developers can invest more in the most efficient sales channels.

**7. Quarterly Builder Contribution**

**How we built it:**

* Added Matrix Table.
* Rows: Developer\_Name.
* Columns: Quarter\_Number.
* Values: Sum(Ticket\_Price\_Cr).

**Insight:**

* Reveals builder dominance by quarter.
* Helps spot seasonal patterns and consistent performers.

**8. Possession Status Analysis**

**How we built it:**

* Added Clustered Column Chart.
* X-axis: Possession\_Status
* Y-axis: count(Booking\_Status).
* Legend: Buyer\_Type.

**Insight:**

* Ready-to-move properties attract NRI and end-user buyers.
* Under-construction projects are popular with investors.

**9. Geographical Insights**

**How we built it:**

* Used Map Visualization.
* Location: Micro\_Market (or lat/long if available).
* Size: count(Project\_Name)

**Insight:**

* Certain zones of Bangalore show dense luxury housing projects.
* Investors can focus on hotspots with strong demand.

**10. Top Performers (Matrix Table)**

**How we built it:**

* Matrix Table created.
* Rows: Developer\_Name.
* Values: Successful Booking and Total Revenue.

**DAX Calculation:**

Total Revenue = SUM('public luxury\_house\_data'[Ticket\_Price\_Cr])

Successful Bookings =

CALCULATE(

COUNTROWS('public luxury\_house\_data'),

'public luxury\_house\_data'[Booking\_Status]= 1

)

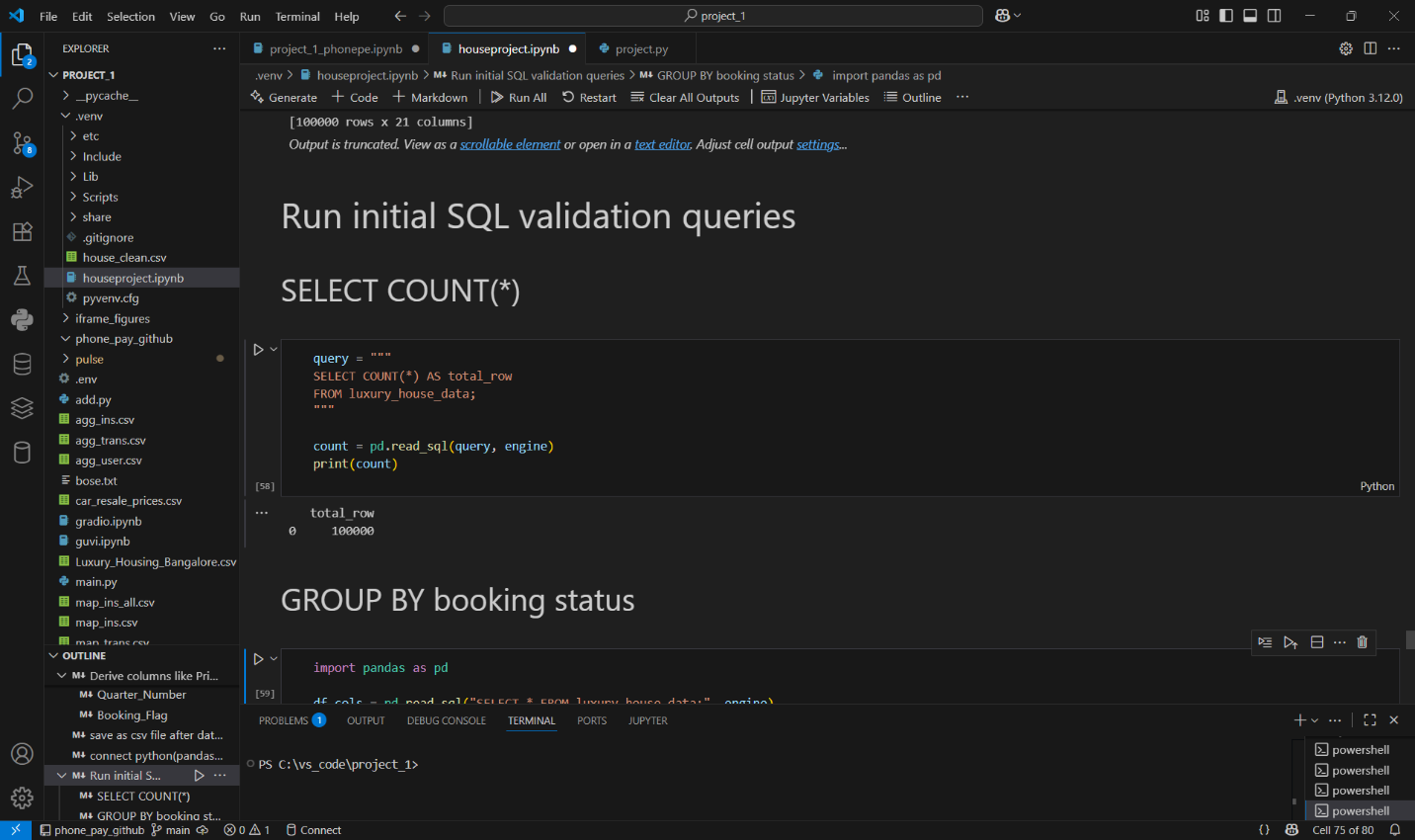
**Insight:**

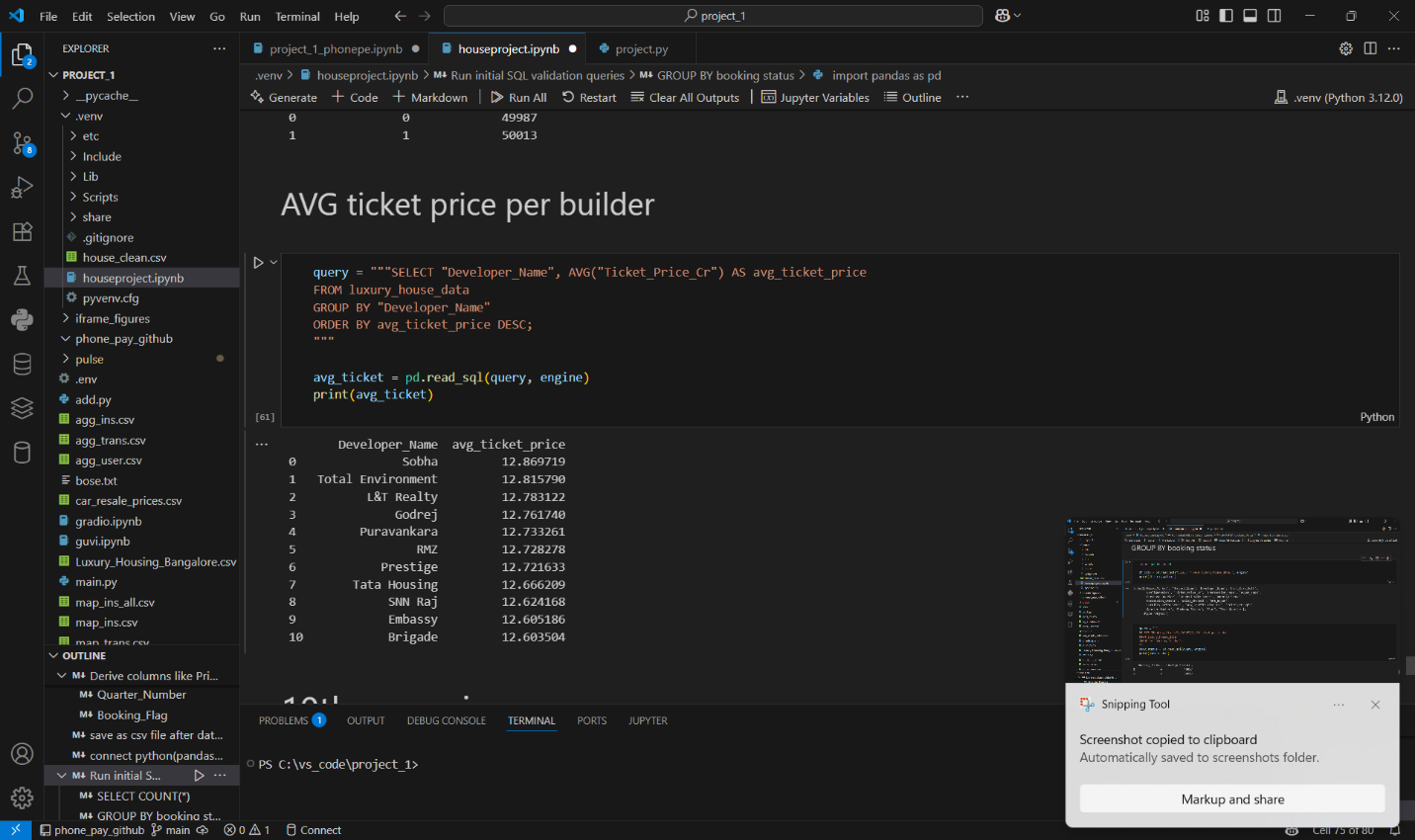
* Top 5 builders contribute the largest share of market revenue.
* They also maintain high booking success, proving strong brand reputation.

SQL Screenshot

A screenshot of a computer program

AI-generated content may be incorrect.





Power BI Screenshot

