# **Airbnb ETL Project Documentation**

## **ETL Process and Data Transformations**

### 1. Extract

Data is extracted from a CSV file containing Airbnb listings information. The file path is specified in the `start` step of the Metaflow workflow.

### 2. Transform

The transformation step includes several operations:

a) Text Normalization:

- Host names, neighborhoods, and neighborhood groups are converted to title case.

b) Date and Time Processing:

- The 'last\_review' column is parsed into separate date and time components.

- Additional date-related columns are created: year, month, day, and day of week.

c) Room Type Standardization:

- Room types are normalized to consistent formats.

d) Missing Value Handling:

- Missing values in 'reviews\_per\_month' are filled with 0.

e) Feature Engineering:

- A new 'price\_per\_night' feature is calculated.

f) Geospatial Data Rounding:

- Latitude and longitude are rounded to 3 decimal places.

g) Categorical Encoding:

- Room types and neighborhood groups are encoded as categorical variables.

h) Data Deduplication:

- Duplicate rows are removed.

i) Data Type Optimization:

- Certain columns are cast to more efficient data types.

j) Aggregation and Metrics Calculation:

- Average price, reviews, minimum nights, and total listings are calculated per neighborhood.

- These metrics are merged back into the main dataset.

### 3. Load

The transformed data is loaded into a PostgreSQL database:

- A 'listings' table is created if it doesn't exist.

- The transformed DataFrame is written to the 'listings' table, replacing any existing data.

## **Instructions for Running the Metaflow Workflow**

1. Ensure Prerequisites:

- Python 3.7+ is installed

- PostgreSQL is set up with a database named 'airbnb'

- Required Python packages are installed: metaflow, sqlalchemy, pandas, psycopg2

2. Configure the Workflow:

- Open `airbnb\_etl\_flow.py`

- Update the `db\_url` in the `start` step with your PostgreSQL connection details

- Set the correct path for `csv\_path` in the `start` step

3. Run the Workflow:

- Open a terminal or command prompt

- Navigate to the project directory

- Execute the following command:

```

python airbnb\_etl\_flow.py run

```

4. Monitor the Execution:

- Metaflow will display the progress of each step in the terminal

- Watch for any error messages or warnings

5. Verify Completion:

- The workflow should end with the message "ETL workflow completed successfully."

6. Check Results:

- Connect to your PostgreSQL database

- Verify that the 'listings' table has been created and populated with the transformed data

### Troubleshooting

- Database Connection Issues:

- Ensure PostgreSQL is running

- Verify the connection string in `db\_url`

- Check that the 'airbnb' database exists

- CSV File Issues:

- Confirm the file exists at the specified path

- Ensure the file has read permissions

- Package Import Errors:

- Verify all required packages are installed in your Python environment

- Execution Errors:

- Review the error message in the terminal

- Check the corresponding step in the code for potential issues

### Customizing the Workflow

To modify the ETL process:

1. Edit the relevant step in `airbnb\_etl\_flow.py`

2. Add new steps by creating additional methods decorated with `@step`

3. Modify the flow of steps by updating the `self.next()` calls

Remember to thoroughly test any changes to ensure the integrity of the ETL process.