Project Documentation: Leads Extractor from Master Sheet

---

1. Introduction

This Python script (leads_finder_2.0.py) helps users find business leads from a dataset based on industry and location inputs. It uses fuzzy matching to handle variations in spelling and abbreviations, ensuring accurate results.

2. What the Script Does

1. **Loads a dataset** (CSV file) containing business leads.
2. **Asks the user for inputs:**
    - Number of leads needed.
    - Industry (or industries) of interest.
    - Location (state, country, or address).
3. Applies fuzzy matching to find relevant leads.
4. Filters results based on industry and location.
5. Saves the filtered leads in a new CSV file.

3. How to Use the Script

Prerequisites

- **Python 3.x** installed.
- Required libraries:
    - pandas (for data handling).
    - fuzzywuzzy (for fuzzy matching).
- A **dataset file** (Sheet1.csv) in the Data folder or set your own path

- A location_data.py file containing location data (states and countries).

4. Steps to Run

1. Place the dataset in the Data folder.
2. Run the script in the terminal:

3. Enter inputs when prompted:
    1. Number of leads (e.g., 10).
    2. Industry (e.g., restaurants, cafes).
    3. Location (e.g., California, US).
4. Confirm or exclude matched industries/locations.
5. Results are saved in a new CSV file inside the Data folder.

4. Key Features

1. Fuzzy Matching

- Handles typos, abbreviations, and partial matches (e.g., "Cali" matches "California").
- Used for both industries and locations.

2. Location Normalization

- Supports country aliases (e.g., "USA" → "United States").
- Matches states, countries, and postal codes.

3. Interactive Filtering

- Users can exclude mismatched industries/locations before saving results.

4. Dynamic CSV Export

- Filename includes the industry and location for easy identification.

5. Code Explanation

1. Importing Libraries

```python
from fuzzywuzzy import fuzz  # For fuzzy string matching
import pandas as pd          # For data handling
```

2. Loading Data

```python
sheet = pd.read_csv(r"Data\Kenneth Hansen - Sheet1.csv")  # Reads the dataset
```

3. Location Handling

- **country_aliases**: Converts abbreviations to full country names (e.g., "US" → "United States").
- **normalize_country():** Ensures consistent country naming.
- **match_location():** Uses fuzzy matching to find states/countries from user input.

4. User Inputs

```python
no_of_leads = int(input("Enter the number of leads you want.\n"))
user_input = input("What specific industry? (comma-separated for multiple):\n")
user_state = input("In which state, country, or address do you want us to find leads?\n")
```

5. Fuzzy Matching Logic

- **Industries:** Compares user input with dataset industries (fuzz.partial_ratio).
- **Locations:** Checks for state/country matches using predefined data (north_america).

6. Filtering & Exporting Results

- Confirms matches with the user.
- Excludes unwanted entries if needed.
- Saves results in a CSV file (e.g., restaurants_leads_california.csv).

6. Expected Output

After running the script:

1. **Matched industries/locations** are displayed for confirmation.
2. A **CSV file** is created with filtered leads (e.g.,):

Name	Industry	Location	Phone
ABC Café	restaurant	Los Angeles, US	+1 555-1234
XYZ Diner	cafe	San Francisco, US	+1 555-5678

7. Troubleshooting

- **"File not found" error**: Ensure the dataset is in the Data folder and the path is set correctly.
- **No matches found**: Try broader search terms (e.g., "food" instead of "vegan restaurants", or make sure the term you are searching is present in the master data sheet).

- **Module errors**: Install missing libraries (pip install pandas fuzzywuzzy).

8. Conclusion

This script simplifies lead generation by:

- Automating data filtering with fuzzy matching.
- Providing interactive controls for refining results.
- Exporting clean, structured data for further use.

For support, contact the developer.