Script Documentation: City Distance and Postal Code Finder using Google Maps API

Overview

This Python script uses the Google Maps API to:
- Calculate driving distance and duration between pairs of cities.
- Retrieve postal codes (ZIP codes) for each city.
- Read and write the results to an Excel file.

It is useful for geographical analysis, logistics planning, or automating the enrichment of city data.

Dependencies

Install required packages:

```
pip install googlemaps pandas openpyxl
```

Project Structure

Function	Description
initialize_gmaps(api_key)	Initializes a Google Maps API client with the provided key.
get_distance_and_duration(gmaps, city1, city2)	Retrieves the driving distance and duration between two cities using the Distance Matrix API.
get_postal_code(gmaps, city)	Retrieves the postal code of a city using the Geocoding API.
load_excel_file(input_file_path)	Loads city pair data from an Excel file into a DataFrame.

save_to_excel(df, output_file_path)	Saves the enriched DataFrame to a new Excel file.
process_distances_and_durations(...)	The main function that iterates over city pairs, fetches distances, durations, and postal codes, and writes the results.

Input Excel Format

The input Excel file (nazvy_miest.xlsx) must contain two columns:

> *| Mesto1 | Mesto2 |*
> *|--------|--------|*
> *| City A | City B |*
> *| City C | City D |*

Output Excel Format

The output Excel file will include:

> *| Mesto1 | Mesto2 | Vzdialenosť autom (km) | Čas cesty (min) | PSČ Mesto1 | PSČ Mesto2 |*

How It Works

1. Reads the input Excel file with city names.
2. Initializes the Google Maps client with the provided API key.
3. For each row:
   - Retrieves driving distance and time between the two cities.
   - Retrieves postal codes for both cities.
   - Fills the results into new columns in the DataFrame.
4. Saves the enriched data to an output Excel file.

API Key Security Note

Your API key is hardcoded in the script:

api_key = "AIzaSyApPAjyICkSHcF4q_sfJlw3kLKrlCLULBQ"

⚠️ This is not recommended for production use. To secure it:
- Store it in an environment variable and read using os.getenv("GOOGLE_MAPS_API_KEY").
- Or use a .env file with python-dotenv.

How to Run

Update the input_file_path, output_file_path, and api_key in the script, then run:

```
python script_name.py
```

Example Output

```
Processing pair: Bratislava - Košice
Distance: 400 km, Duration: 4 hours 10 mins, Postal Code City1: 81101, Postal Code City2: 04001

Results saved to nazvy_miest_s_vzdialenostami_a_casom.xlsx
```

Error Handling

- If a city or address cannot be found, appropriate error messages are printed.
- The script handles exceptions to avoid crashes on individual failures.