

Lab Exercise 9

Assignment Overview

This lab exercise provides practice with dictionaries of lists and sets in Python.

A. Write a program using Dictionaries of lists

Consider the file named “lab9a.py”. Given two files named exactly `continents.txt` and `cities.txt` (no error checking of the file name is needed) of continents, countries and cities. Write a program to read continents, countries and their cities, put them in a nested dictionary and print them (no duplicates are allowed, i.e. no continent should have the same country listed twice and no country should have the same city listed twice). You should ignore empty strings. Note that both files will be formatted the same, but some names will be in both files and some names will only be in one file. The file format will be one header line followed by lines that have names (string) separated by some unknown number of spaces. A continent can have multiple countries. A country can have multiple cities. If a country is in the `cities.txt` file, but not in the `continents.txt` file, it should be ignored. When displaying the dictionaries, the data should be sorted by Continent, by Country, and by city. See sample output below. For output, use this format string for countries “{:>10s} --> ”

Requirements:

- (1) You must use a nested dictionary (dictionary of lists). The keys for the nested dictionaries are `Continent` and `Country`.
- (2) You must use the functions that are provided in the template.

Hints

- (1) Use a nested dictionary (string \rightarrow (Dictionary \rightarrow List<string>))
`Data_map[Continent][Country] \rightarrow List of cities`
- (2) To access the outer dictionary, use the first key, e.g., `Data_map[Continent]`
- (3) To access the inner dictionary, use the second key, e.g.,
`Data_map[Continent][Country]`
- (4) To access a value in a nested dictionary, you have to use both keys and an Index, e.g.
`Data_map[Continent][Country][Index]`
- (5) Check if the continent exists before adding the country. If it doesn't, add it to the dictionary.
- (6) Check if the country exists, before adding the city. If it doesn't, add it to the dictionary.
- (7) Check if the city exists. If it doesn't, add it to the dictionary.

For example, if `continents.txt` contains

Continent	Country
Africa	Tunisia
Europe	Bulgaria
Asia	China
Asia	Japan
Europe	Poland
Europe	Germany
Africa	Nigeria
Africa	Tunisia

and `cities.txt` contains

Country	City
Bulgaria	Sofia
China	Beijing
Japan	Tokyo
Tunisia	Sousse
Poland	Warsaw
Germany	Berlin
Poland	Poznan
Bulgaria	Plovdiv
Nigeria	Abuja
China	Shanghai
Tunisia	Tunis
France	Paris
Japan	Tokyo

the output will be:

Africa:

Nigeria --> Abuja
Tunisia --> Sousse, Tunis

Asia:

China --> Beijing, Shanghai
Japan --> Tokyo

Europe:

Bulgaria --> Plovdiv, Sofia
Germany --> Berlin
Poland --> Poznan, Warsaw

★ **Demonstrate your completed program to your TA. On-line students should submit the completed program (named “lab09a.py”) for grading via the Mimir system.**

B. Write a program using Dictionaries of sets

Rename your file from Part A to “lab9b.py”. Given the same problem in part A, modify the program to replace lists with sets, i.e. use a dictionary of sets instead of dictionary of lists.

Requirements:

- (1) You must use a dictionary of sets.

Hints

- (1) You probably will only have to modify the `build_map` function by replacing list initialization with set initialization and using set methods in place of list methods. (Whether you have to modify the `display_map` function depends on how you implemented it in Part A.)

★ **Demonstrate your completed program to your TA. On-line students should submit the completed program (named “lab09b.py”) for grading via the Mimir system.**