Programming Assignment 09
-
Dictionaries and Sets

Instructions

This programming assignment consists of **1 programming exercise**.
You have to:

1. **download** the empty Python file on NYU Classes
2. **edit** it according to the assignment
3. **verify** on your computer that it works
4. **upload** it back on NYU Classes (**do not change the filename**)

Exercise 1 - Top Tourist Attractions in China

The `top_tourist_attractions.txt` file (in the same folder as your program) contains some **top tourist attractions** in China and **their location** (Province). Check the format in the file.

This exercise consists of 5 tasks (to be coded in `exercise1.py`):

Task 1

Write a function named `build_attraction_dict` which takes a **filename** (`str` type) as a parameter, and **returns a dictionary**.
The `key` / `value` pairs in the **dictionary** will be:

- `key` → **attraction name** (`str` type)
- `value` → **Province** (`str` type)

read from the file for which the **filename** was given. The file is supposed to have a format similar to `top_tourist_attractions.txt`.

Task 2

Write a function named `add_attraction` which takes a **dictionary as a parameter** and does not return anything.
The function:

- asks the user to **input an attraction name**
- asks the user to **input a province**
- **updates the dictionary** with a new `key` / `value` pair (with a similar format than for task 1)

Task 3

Write a function named `build_province_attraction_dict` which takes a **dictionary as a parameter** and **returns another dictionary**.
The input dictionary is constituted of `key` / `value` pairs similar to the ones built in task 1.
The **returned dictionary** should have `key` / `value` pairs with:

- `key` → **Province** (`str` type)
- `value` → **list of attraction names** (`list` of `str` type) associated to the Province

Task 4

Write a function named `most_attractions` which takes a **dictionary as a parameter** and **returns a set**.
The input dictionary is constituted of `key` / `value` pairs similar to the ones built in task 3.
The **returned set** should contain the **Provinces** which have **at least 3 tourist attractions** (in the input dictionary).

Task 5

Write the `main()` function that:

1. calls `build_attraction_dict` with filename `top_tourist_attractions.txt` as parameter to generate the first dictionary.
2. calls `add_attraction` **one time** in order to update the dictionary. The user will input a new (non-existing) **attraction name**/**Province** pair.
3. calls `build_province_attraction_dict` in order to generate a second dictionary.
4. calls `most_attractions` in order to generate the set of Provinces with at least 3 attractions.
5. prints the Provinces with at least 3 attractions. Display **one Province per line**. See sample examples for the proper format (empty line + Header line).

Sample example 1 with the file `top_tourist_attractions.txt` (the user input is in red, the printed output is in blue (also with the empty line)):

```
Input a new attraction: The Bund
Input the province: Shanghai

Provinces with at least 3 attractions:
Yunnan
Guangxi
```

Sample example 2 with the file `top_tourist_attractions.txt` (the user input is in red, the printed output is in blue (also with the empty line)):

```
Input a new attraction: Tiananmen
Input the province: Beijing

Provinces with at least 3 attractions:
Yunnan
Guangxi
Beijing
```