

CS 115 - Introduction to Programming in Python

Lab 04

Lab Objectives: Strings, Files, Modules

Notes:

- You should only use functionality covered in CS115 in your solution.
- Your solution for this lab should not use the `str.split()` method, lists, tuples, dictionaries.
- Include a docstring for your functions. A template for function docstrings is below:

```
"""Summary or Description of the Function

Parameters:
argument1 (type): Description of arg1

Returns:
type: return value
"""
```

1. Download the files `city_districts.txt`, `city_data.txt`. The file `city_districts.txt` contains 2 columns, the first is the city name and the second are the districts in the city. The file `city_data.txt` contains the population and area data for each of the districts in the `city_district` file (in the same order).
2. Create a module, `Lab04_yourname_module.py` that contains the following functions:
 - a. `district_density()`: Takes a string city name and 3 file names as parameters. The first file is an input file containing the city and district names, the second file is an input file containing the population and area data for the districts in the cities. The third parameter is the name of the file to store the results. The function should read the data from the `city_districts.txt` file and for each district in the city whose name is passed as a parameter, calculate the population density using the data from the corresponding row of the `city_data.txt` file. The function should write the district name and population density of each district (in the given city) to the output file with the name passed as a parameter. Each line in the output file should contain the district name followed by the density. The columns should be separated by the comma character.
Sample line from the output file – `ankara_density.txt`:
`Akyurt,89.1`
...
 - b. `find_districts ()`: takes a population density and a file name as parameters. Finds and returns a string containing the names of all districts in the file with a density below the value passed as a parameter. Returns an empty string if no matching districts found.
3. Write a script, `Lab04_yourname_application.py` that does the following (using the functionality defined in the module `Lab04_yourname_module.py`):
 - a. Input a search city from the user (not case sensitive).
 - b. Create a file with the name `searchcity_density.txt`, where `searchcity` is the city input by the user. The file should contain the districts and the population densities of all districts in the given city. If the city does not exist in the `city_district` file, display an appropriate message (see sample run below).
 - i. If the city is found, input a density from the user, and display the names of the districts whose density is below the value entered by the user (display appropriate message if no matching districts found).
 - c. Your program should continue to prompt for search cities until the user enters 'quit' (not case sensitive).
4. Upload `Lab04_yourname_module.py` and `Lab04_yourname_application.py` in a compressed file with the name: `Lab04_yourname.zip`.

Sample Run:

Enter city to search ("quit" to exit): Izmir
Izmir not found...

Enter city to search ("quit" to exit): Ankara

Enter maximum density: 50
Districts in Ankara with population density below 50.0:
Ayas
Bala
Beypazari
Camlidere
Etimesgut
Evren
Gudul
Haymana
Kizilcahamam
Nallihan
Polatli
Sereflikochisar

Enter city to search ("quit" to exit): Istanbul

Enter maximum density: 0.1
No districts in Istanbul with population density below 0.1

Enter city to search ("quit" to exit): Istanbul

Enter maximum density: 10
Districts in Istanbul with population density below 10.0:
Arnavutkoy
Beykoz
Catalca
Silivri
Sile

Enter city to search ("quit" to exit): QUIT
Thank you - Goodbye