Documentation on Coding Challenge

Prepared by:

Ankit Arora

Contents

Problem

This program takes an input in the form of a .txt file(comma separated) and converts .txt file into an JSON output format.

Solution

Libraries are imported to start with like pandas , phonenumbers and regular expression. Using read.csv , data is read from the .txt file delimiting them by commas and assigning the columns names as C1 to C5 just for easy reference.

Below functions and classes to do the manipulation needed on the data.

Class: entries

This is a class to define the type of variables used and sequence of columns can be used to standardize input records.

Function: validate_phone_number

This function is going to utilize phone numbers library and check if there is a valid US number in the input record. Return True if valid number is found else return False.

Function: format_phone_number

This function is going to take validated phone number as input, all the extraneous characters like '(', ')', '-', is removed from the string and formatted into a standard US phonenumber as xxx-xxx-xxxx.

Function: clean_data

This function is going to take validated phone number as input and strips all the extra whitespaces from the input records.

Function: process_data

This function is using a loop, to iterate through the rows in the dataframe and validating the phonenumbers in columns 3rd , 4th and 5th.

- If a valid phonenumber is found in 3rd column then its matching the format “Lastname, Firstname, phonenumber, color, zipcode”, and converting to the stardard class object accordingly.
- Else if a phonenumber is found in 4th column, the program is checking if the 5th column exists. There are some cases identified where firstname and lastname is not separated by comma, so, name is split by space.
- Same logic is applied to convert the record by validating 4th and 5th columns as valid phonenumbers.
- Else the record is considered an error record and assigned the error record index to “errors” list.
- All the correct data is transformed and converted to a standard class “entries” list.

Total time complexity taken for the for loop is O(n).

All these entries listed are sorted by lastname and firstname. After the sorting of data, data is added into the dictionary. Same way “errors” List is also converted to Json and added to a dictionary using a key, value pair.

This converted JSON object is written to an output file with indent 2.

Instruction to run

A Python program is created, python program can be run either by command line

>>>Python Guidewire_Ankit_solution.py

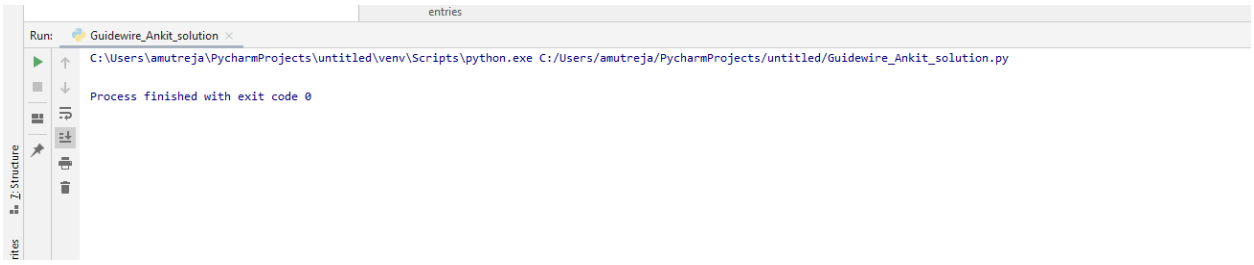
Or directly from Pycharm

All the test cases can be run using the below statement

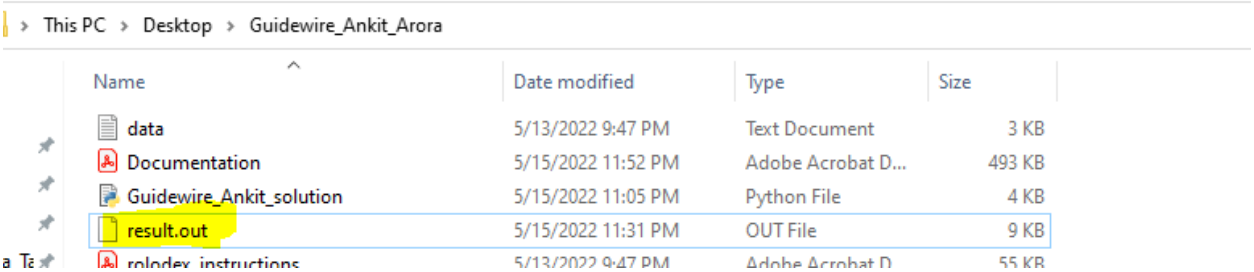
>>> pytest .\test_Guidewire_Ankit_solution.py -s -vv

Execution results:

Execution using pycharm

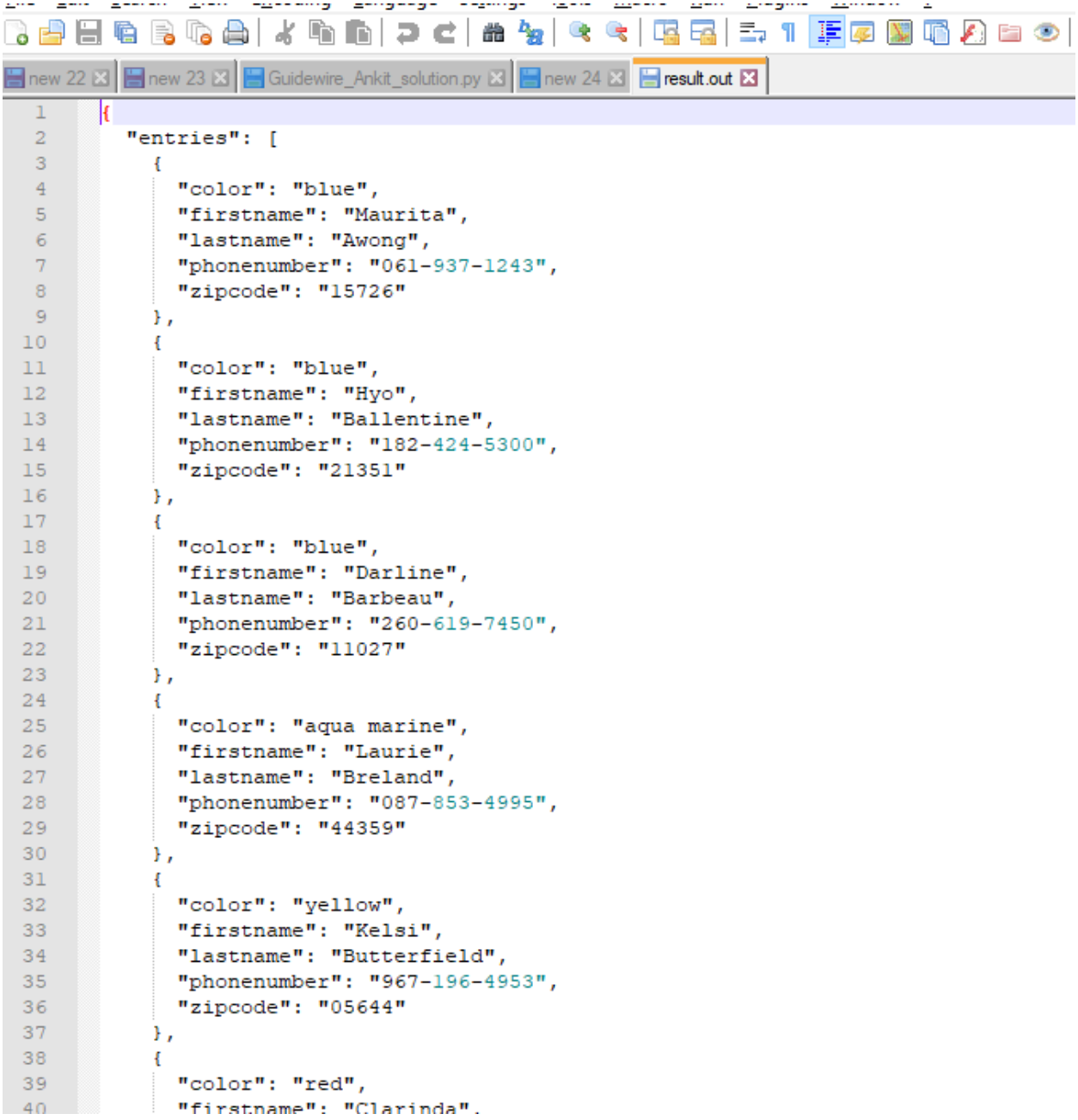


"result.out" JSON file



JSON output file snippet and output

Output Snippet:



JSON file

Attached is the output full JSON file

Test cases

Below test cases are created to validate the solution.

test_validate_phone_number

This test validates phone numbers and it will validates the length of the phone number. It will assert True if the length is 10 digits else assert False.

test_format_phone_number

This test format phone numbers into standard US format. It will assert True if the string is correctly converted to US format xxx-xxx-xxxx else assert False.

test_clean_data

This test cleans the phone numbers into standard US format. It will assert True if the string is converted to standard US format xxx-xxx-xxxx after removing whitespaces else assert False.

test_process_data_errors

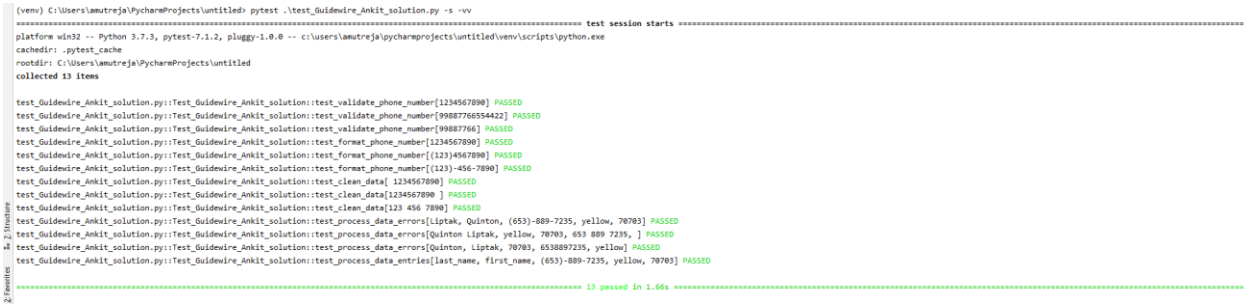
This test will read read_csv understands binary stream as well, use it to test subset of data instead of whole file.

test_process_data_entries

This test converts the sample data into JSON format and assert True if the value matches expected_JSON object else assert FALSE.

Test cases results:

All the test cases passed



```
(venv) C:\Users\amutreja\PycharmProjects\untitled> pytest .\test_Guidewire_Ankit_solution.py -s -vv
============================================================================================================ test session starts ============================================================================================================
platform win32 -- Python 3.7.3, pytest-7.1.2, pluggy-1.0.0 -- c:\users\amutreja\pycharmprojects\untitled\venv\scripts\python.exe
cachedir: .pytest_cache
rootdir: C:\Users\amutreja\PycharmProjects\untitled
collected 13 items

test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_validate_phone_number[1234567890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_validate_phone_number[99887766554422] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_validate_phone_number[99887766] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_format_phone_number[1234567890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_format_phone_number[(123)4567890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_format_phone_number[(123)-456-7890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_clean_data[ 1234567890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_clean_data[1234567890 ] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_clean_data[123 456 7890] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_process_data_errors[Liptak, Quinton, (653)-889-7235, yellow, 70703] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_process_data_errors[Quinton Liptak, yellow, 70703, 653 889 7235, ] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_process_data_errors[Quinton, Liptak, 70703, 6538897235, yellow] PASSED
test_Guidewire_Ankit_solution.py::Test_Guidewire_Ankit_solution::test_process_data_entries[last_name, first_name, (653)-889-7235, yellow, 70703] PASSED

============================================================================================================ 13 passed in 1.66s =============================================================================================================
```