Is this CSCD 437
Lab 5
Regular Expressions

SPECIFICATIONS

You are provided two Python3 files. The two files are CSCD437RegExTesting.py and CSCD437RegExMethods.py. You will write code in the CSCD437RegExMethods.py file. You may not change our code in the CSCD437RegExMethods.py file. All you have to do is write regex=and the regular expression for the appropriate function. The regular expressions you must write are specified below. **NOTE: This is python code tabs and spacing matter.**

1. Social Security Number (can be with dashes, whitespace, or no spaces at all) NOTE: dashes and whitespace will only happen between place 3 and 4 and/or between 5 and 6
    - Some Valid Examples
        - 123456789
        - 123-45-6789
        - 123 45 6789
        - 123-45 6789
        - 123 45-6789
        - 12345-6789
    - Some NOT Valid Examples
        - 12-345-6789
        - 123--45-6789
        - 123.45.6789

2. US Phone number – country code area code exchange code subscriber number. You only need to worry about US numbers, anything other than +1 should fail.
    - Parentheses are optional.
    - Space/Dash optional
        - Between area code and number
        - Between exchange code and subscriber number
    - +1 for country code is optional (must contain the + to be valid)
        - Space is the only valid optional character between country code and area code
    - Some Valid Examples
        - (509) 123-4567
        - 509-123-4567
        - 123 456 7890
        - +1 509-123-4567
        - +1 (509) 123-4567
    - Some NOT Valid Examples
        - 1 (509) 123-4567
        - 509.123.4567
        - 123--456-7890

3. Name on a class roster - Last name (optional roman numerals), First name, MI (capitalization is optional, and middle initial is optional). The last name can contain a space or a dash.
    - Some Valid Examples
        - Steiner, Stuart, G
        - Steiner, Stu
        - steiner III, stu, g
        - da Vinci, Leonardo
    - Some NOT Valid Examples
        - John Doe
        - Doe, John $
        - Doe, John
        - Doe John A
        - Doe, John Adam

4. Date in MM-DD-YYYY format
    - You must account for:
        - Separators can be either the dash (-) or the forward slash (/) but not both
        - An optional leading zero.
        - A date that is a leap year date
        - Must be a valid date
    - Some Valid Examples
        - 01/01/2024
        - 01-01-2024
        - 2/29/2024
    - Some NOT Valid Examples
        - 01/01-2024
        - 01 01-2024
        - 03/32/2024
    - NOTE: You may write two helper methods if you need. The method names are defined below. You may add calls to the helper methods in this function before you have the regular expression code. You may not change anything else in any function.
        - is_leap
        - is_valid_date

5. IPv4 Address in ###.###.###.### format
    - You must account for:
        - You must account for numbers between 0 and 255.
        - Leading 0 for addresses less than 100 is acceptable
    - Some Valid Examples
        - 123.123.123
        - 192.168.0.1
        - 033.033.33.33
    - Some NOT Valid Examples
        - 123.456.789
        - 999.999.999.999
        - 123.123.256.256

NOTES

- We have provided CSCD437RegExMethods.py and CSCD437RegExTesting.py.
  - Write your code for the appropriate regular expression in the method provided.
  - You can't change our methods or code in any fashion, other than to add your regular expression code or to call the appropriate helper function.
  - We have provided minimal tests.
  - We have provided everything but the regular expression code. (regex=the expression)
  - You must include a detailed explanation of each regex, meaning you must explain in detail what the regex does/how it works.
- **Your testing MUST be thorough and robust -- you will be heavily penalized if this is not the case. You should have at least 6 tests per problem.**

WHAT TO TURN IN:

- Submit a zip file that contains:
  - CSCD437RegExMethods.py
  - A detailed explanation of each regular expression and the output illustrating your robust testing for each regex. Name the file your last name first letter of first namelab5.pdf each item.
  - We should be able to download your Python file and run it with our CSCD437RegExTesting.py file.
- Name your zip your last name first letter of first namelab5.zip (Example: steinerslab5.zip)