CPSC 1301, Computer Science I Quiz

Quiz 7b

This is a timed test. You have thirty minutes to complete the test. Your deliverable will be a plain text file, that is, an ASCII file with a `.txt` file extension. When you finish the test, upload your deliverable to Canvas. Do not publish your answer to your `git` repository.

You may work with your study partner for this quiz. In fact, working together is strongly encouraged. If you work with a partner, you must each make a seperate submission for credit, but you must also include the names of both authors in your submission.

1 Instructions

This quiz will give you the experience of doing something that you will do very often in your programming career: munge data. You will typically open a data file, read the data into some kind of data structure, and manipulate the data. I have given you the data. You will read it into a data structure. This consists of a `presidents` list. Each element of the list consists of a president's data. The details are given in section 2.

I have given you detailed instructions in the code template. Please read them carefully. There are four tasks. Also note that you will need to use these methods: (1) `int()`, (2) `str.split()`, and (3) `str.replace()`. You may also find `str.isaplha()` and the `len()` function useful. READ THE DOCUMENTATION!

Task 1 Take the data, which is in the form of one long string, and split it based on the end-of-line character. Use `str.split()`. This will result in a list, each element of the list consisting of one line in the file. Since you will have some empty lines and the header to delete, you may choose to delete them in this step. Name the reulting list `presidetsList`.

Task 2 Create an empty list to hold the output. Name this list `presidents`. Iterate through your `presidentsList`, and from each row create a new dictionary. The dictionary should have four elements:

'order' This contains an integer. Use the `int()` function.

'name' This is a dictionary that contains `'firstName'`, `'middleName'`, and `'lastName'`. You will need to delete the double quotation marks; use `str.replace()`.

'birthState' This contains a string. Note that you will need to delete the double quotes.

'birthDay' This contains a list of three integers, in order, year, month, day.

Append each dictionary to your list. Note that these will be anonymous dictionaries that you access by indexing into the list. You may also choose to delete the header line and the empty lines in this step. This is where I took care of this.

Task 3 Print the `presidents` list using a `for` loop. I've given you the code. This is only meant to check your work. You can look at this list and see how you are doing.

Task 4 Print `presidents`. This output should be the final form.

2 Data

The data consists of a file in CSV (comma separated values) format. The first row is the header containing the names of the columns (fields).

- the order of the president, i.e., first, second, third, etc. (int)
- the president's first name (string)
- the president's middle name (string)
- the president's last name (string)
- the president's birth state (string)
- the president's birth year (int)
- the president's birth month (int)
- the president's birth day (int)

The data is something that you would retrieve from a database. This is the data:

```
order,firstName,middeName,lastName,birthState,birthYear,birthMonth,birthDay
1,"George","","Washington","Virginia",1722,2,22
2,"John","","Adams,"Massachusetts",1735,10,30
3,"Thomas","","Jefferson","Virginia",1743,4,13
4,"James","","Madison","Virginia",1751,3,16
5,"James","","Monroe","Virginia",1758,4,28
6,"John","Quincy","Adams","Massachusetts",1767,11,7
```

3 Template

```
#quiz07b.py

# this is the presidents data in CSV (comma separated values) format
# note that this inside a triple quoted string
presidentsData = '''
order,firstName,middeName,lastName,birthState,birthYear,birthMonth,birthDay
1,"George","","Washington","Virginia",1722,2,22
2,"John","","Adams,"Massachusetts",1735,10,30
3,"Thomas","","Jefferson","Virginia",1743,4,13
4,"James","","Madison","Virginia",1751,3,16
5,"James","","Monroe","Virginia",1758,4,28
6,"John","Quincy","Adams","Massachusetts",1767,11,7
'''
print(presidentsData)

# 1 ------------------------------------
# first, split the CSV data into a list using str.split()
# each line of the data should be a separate element of the array
# you will need to split on the end-of-line character (you know what it is)
# name this list presidetsList
presidentsList = pass
# print the list just to make sure it works
print(presidentsList)

# 2 ------------------------------------
# second, iterate through each element of presidentsList and create a dictionary
# create an empty list to hold the data
# append each dictionary to the list
# note that each dictionary will not have a name, it will be anonymous
presidents = []
for line in presidentsList:
    pass
    # each record will be a dictionary with four fields, like this
    # 'order' is an integer
    # 'name' is a dictionary composed of three strings
    #       'firstName', 'middleName', and 'lastName'
```

```
37      # 'birthState is a string
38      # 'birthDay' is a list of three integers, year, month, day
39
40  # 3 ------------------------------------
41  # third, print the presidents
42  # do not include the header line in your data
43  # do not include anything in your data except the presidents
44  # this is just so you can check your work
45  # do NOT include double quotes or empty lines in your presidents!
46  for p in presidents:
47      for key, value in p.items():
48          print("%s => %s" % (key, value))
49      print()
50
51  # 4 ------------------------------------
52  # finally, print te presidents list (note the leading and ending set of brackets [])
53  # your output should be like this:
54  #[{'order': 1, 'name': {'firstName': 'George', 'middleName': '', 'lastName': 'Washington'},
        'birthState': 'Virginia', 'birthDay': [1722, 2, 22]}, {'order': 2, 'name': {'firstName':
         'John', 'middleName': '', 'lastName': 'Adams'}, 'birthState': 'Massachusetts', '
        birthDay': [1735, 10, 30]}, {'order': 3, 'name': {'firstName': 'Thomas', 'middleName':
        '', 'lastName': 'Jefferson'}, 'birthState': 'Virginia', 'birthDay': [1743, 4, 13]}, {'
        order': 4, 'name': {'firstName': 'James', 'middleName': '', 'lastName': 'Madison'}, '
        birthState': 'Virginia', 'birthDay': [1751, 3, 16]}, {'order': 5, 'name': {'firstName':
        'James', 'middleName': '', 'lastName': 'Monroe'}, 'birthState': 'Virginia', 'birthDay':
        [1758, 4, 28]}, {'order': 6, 'name': {'firstName': 'John', 'middleName': 'Quincy', '
        lastName': 'Adams'}, 'birthState': 'Massachusetts', 'birthDay': [1767, 11, 7]}]
55  print(presidents)
```

4 Output

```
[{'order': 1, 'name': {'firstName': 'George', 'middleName': '', 'lastName': 'Washington'},
    'birthState': 'Virginia', 'birthDay': [1722, 2, 22]}, {'order': 2, 'name': {'firstName':
    'John', 'middleName': '', 'lastName': 'Adams'}, 'birthState': 'Massachusetts', 'birthDay':
    [1735, 10, 30]}, {'order': 3, 'name': {'firstName': 'Thomas', 'middleName': '', 'lastName':
    'Jefferson'}, 'birthState': 'Virginia', 'birthDay': [1743, 4, 13]}, {'order': 4, 'name':
    {'firstName': 'James', 'middleName': '', 'lastName': 'Madison'}, 'birthState': 'Virginia',
    'birthDay': [1751, 3, 16]}, {'order': 5, 'name': {'firstName': 'James', 'middleName': '',
    'lastName': 'Monroe'}, 'birthState': 'Virginia', 'birthDay': [1758, 4, 28]}, {'order': 6,
    'name': {'firstName': 'John', 'middleName': 'Quincy', 'lastName': 'Adams'}, 'birthState':
    'Massachusetts', 'birthDay': [1767, 11, 7]}]
```