[Solution] CP03

July 17, 2022

0.0.1 01. Capital Quiz

Write a program that creates a dictionary containing the U.S. states as keys and their capitals as values. (Use the Internet to get a list of the states and their capitals.) The program should then randomly quiz the user by displaying the name of a state and asking the user to enter that state's capital. The program should keep a count of the number of correct and incorrect responses. (As an alternative to the U.S. states, the program can use the names of countries and their capitals.)

[1]:

```
import random

capitals = {'Alabama': 'Montgomery', 'Alaska': 'Juneau',

            'Arizona': 'Phoenix', 'Arkansas': 'Little Rock',

            'California': 'Sacramento', 'Colorado': 'Denver',

            'Connecticut': 'Hartford', 'Delaware': 'Dover',

            'Florida': 'Tallahassee', 'Georgia': 'Atlanta',

            'Hawaii': 'Honolulu', 'Idaho': 'Boise',

            'Illinois': 'Springfield', 'Indiana': 'Indianapolis',

            'Iowa': 'Des Moines', 'Kansas': 'Topeka',

            'Kentucky': 'Frankfort', 'Louisiana': 'Baton Rouge',

            'Maine': 'Augusta', 'Maryland': 'Annapolis',

            'Massachusetts': 'Boston', 'Michigan': 'Lansing',

            'Minnesota': 'Saint Paul', 'Mississippi': 'Jackson',

            'Missouri': 'Jefferson City', 'Montana': 'Helena',

            'Nebraska': 'Lincoln', 'Nevada': 'Carson City',
            'New Hampshire': 'Concord', 'New Jersey': 'Trenton',

            'New Mexico': 'Santa Fe', 'New York': 'Albany',

            'North Carolina': 'Raleigh', 'North Dakota': 'Bismarck',

            'Ohio': 'Columbus', 'Oklahoma': 'Oklahoma City',

            'Oregon': 'Salem', 'Pennsylvania': 'Harrisburg',

            'Rhode Island': 'Providence', 'South Carolina': 'Columbia',

            'South Dakota': 'Pierre', 'Tennessee': 'Nashville',

            'Texas': 'Austin', 'Utah': 'Salt Lake City',

            'Vermont': 'Montpelier', 'Virginia': 'Richmond',

            'Washington': 'Olympia', 'West Virginia': 'Charleston',

            'Wisconsin': 'Madison', 'Wyoming': 'Cheyenne'}

correct = 0
wrong = 0
next_question = True
index = 0
user_solution = ''

while next_question:

    state_iterator = iter(capitals)

    # Get a random state name for the question.

    index = (random.randint(1, len(capitals)) - 1)

    for i in range(1 - index):
        temp = state_iterator.__next__()

    current_state = str(state_iterator.__next__())

    # Get user solution.

    user_solution = input('What is the capital of {}? (or enter 0 to quit): '.format(current_state))

    # User wants to quit the game.
    if user_solution == '0':
        next_question = False
        print('You had {} correct responses and {} incorrect responses.'.
 ↪format(correct, wrong))

    # User solution is correct.

    elif user_solution == capitals[current_state]:
        correct = correct + 1
        print('That is correct.')

    # User solution is incorrect.

    else:
        wrong = wrong + 1
        print('That is incorrect.')
```

```
What is the capital of Alabama? (or enter 0 to quit): Alaska
That is incorrect.
What is the capital of Alabama? (or enter 0 to quit): Montgomery
That is correct.
What is the capital of Alabama? (or enter 0 to quit): 0
You had 1 correct responses and 1 incorrect responses.
```

0.0.2 02. Course information

Write a program that creates a dictionary containing course numbers and the room numbers of the rooms where the courses meet. The dictionary should have the following keyvalue pairs:

Course Number (key)	Room Number (value)
CS101	3004
CS102	4501
CS103	6755
NT110	1244
CM241	1411

The program should also create a dictionary containing course numbers and the names of the instructors that teach each course. The dictionary should have the following key-value pairs:

Course Number (key)	Instructor (value)
CS101	Haynes
CS102	Alvarado
CS103	Rich
NT110	Burke
CM241	Lee

The program should also create a dictionary containing course numbers and the meeting times of each course. The dictionary should have the following key-value pairs:

Course Number (key)	Meeting Time (value)
CS101	8:00 a.m.
CS102	9:00 a.m.
CS103	10:00 a.m.
NT110	11:00 a.m.
CM241	1:00 p.m.

[2]:

```
course_room = {'CS101': 3004, 'CS102': 4501, 'CS103': 6755,
               'NT110': 1244, 'CM241': 1411}

course_instructor = {'CS101': 'Haynes', 'CS102': 'Alvarado',
                     'CS103': 'Rich', 'NT110': 'Burke', 'CM241': 'Lee'}

courser_time = {'CS101': '08:00 a.m', 'CS102': '09:00 a.m',
                'CS103': '10:00 a.m', 'NT110': '11:00 a.m',
                'CM241': '01:00 p.m'}

print(course_room)
print(course_instructor)
print(courser_time)
```

```
{'CS101': 3004, 'CS102': 4501, 'CS103': 6755, 'NT110': 1244, 'CM241': 1411}
{'CS101': 'Haynes', 'CS102': 'Alvarado', 'CS103': 'Rich', 'NT110': 'Burke',
'CM241': 'Lee'}
{'CS101': '08:00 a.m', 'CS102': '09:00 a.m', 'CS103': '10:00 a.m', 'NT110':
'11:00 a.m', 'CM241': '01:00 p.m'}
```

0.0.3 03. Find winner of election

Given an array of names of candidates in an election. A candidate name in the array represents a vote cast to the candidate. Print the name of candidates received Max vote. If there is tie, print a lexicographically smaller name.

input = [‘john’,‘johnny’,‘jackie’,‘johnny’, ‘john’,‘jackie’,‘jamie’,‘jamie’, ‘john’,‘johnny’,‘jamie’,‘johnny’, ‘john’]

Output : John.
We have four Candidates with name as ‘John’, ‘Johnny’, ‘jamie’, ‘jackie’. The candidates John and Johny get maximum votes. Since John is alphabetically smaller, we print it.

[3]:

```
votes = ['john','johnny','jackie','johnny', 'john','jackie','jamie','jamie',␣
 ↪'john','johnny','jamie','johnny', 'john']

candidates = list(set(votes))
```

```
max_votes = max(votes, key=votes.count)
print(max_votes.title(), end='.\n')

votes_count = []
for i in range(len(candidates)):
    max_votes = max(votes, key=votes.count)
    votes_count.append(max_votes.title())
    votes.remove(max_votes)


print("We have four Candidates with name as ",end='')
print(', '.join(candidates).title(), end='. ')
print('The candidates {} and {} get maximum votes. Since {} is alphabetically␣
 ↪smaller, we print it.'
      .format(votes_count[0], votes_count[1], max_votes.title()))
```

```
John.
We have four Candidates with name as John, Jamie, Jackie, Johnny. The candidates
John and Johnny get maximum votes. Since John is alphabetically smaller, we
print it.
```

0.0.4 04. Reverse Incremental String Slicing

Sometimes, while working with Pythonn strings, we can have a problem in which we need to perform the slice and print of strings in reverse order. This can have application in day-day programming. So in this exercise write a program that reverse the string in incremental order.

For Example:

```
Orignal String : GIFT

Incremental reverse strings: ['T','TF','TFI','TFIG']
```

[4]:

```
orig_str = 'GIFT'

reverse_str = []
sub = ''

print("Orignal String : ",orig_str)

for ch in reversed(orig_str):
    sub += ch
    reverse_str.append(sub)

print("Incremental reverse strings: ", str(reverse_str))
```

```
Orignal String :  GIFT
Incremental reverse strings:  ['T', 'TF', 'TFI', 'TFIG']
```

0.0.5 05. Alphabetic Telephone Number Translator

Many companies use telephone numbers like 555-GET-FOOD so the number is easier for their customers to remember. On a standard telephone, the alphabetic letters are mapped to numbers in the following fashion:

Letters	Number
A, B, and C	2
D, E, and F	3
G, H, and I	4
J, K, and L	5
M, N, and O	6
P, Q, R, and S	7
T, U, and V	8
W, X, Y, and Z	9

Write a program that asks the user to enter a 10-character telephone number in the format XXX-XXX-XXXX. The application should display the telephone number with any alphabetic characters that appeared in the original translated to their numeric equivalent. For example, if the user enters 555-GET-FOOD the application should display 555-438-3663.

[5]:

```
phone_letters = {'A': '2','B': '2','C': '2','D': '3','E': '3',
                 'F': '3','G': '4','H': '4','I': '4','J': '5',
                 'K': '5','L': '5','M': '6','N': '6','O': '6',
                 'P': '7','Q': '7','R': '7','S': '7','T': '8',
                 'U': '8','V': '8','W': '9','X': '9','Y': '9',
                 'Z': '9'}



number = input('Enter a 10-character telephone number in the format␣
 ↪XXX-XXX-XXXX: ').replace('-', '')

result = []

for n in number:
    if n.isdigit():
        str(result.append(n))

    if n.isalpha():
        value = phone_letters.get(n.capitalize())
        result.append(value)


result.insert(3, '-')
result.insert(7, '-')

# join list in string
```

```
print('Your phone number:', ''.join(result))
```

```
Enter a 10-character telephone number in the format XXX-XXX-XXXX: 555-get-food
Your phone number: 555-438-3663
```