CSCD 330 - Computer Networks

Lab 2, Web API

Overview:

Write a Python program to check the weather at the address of a company that owns a specified domain name. Your program will have to make several different API calls to accomplish this task.

Instructions:

Complete the above program following the steps in the next section. You must use the APIs listed in those steps. Call this file lab2.py and include your name at the top of the file – #author: YOUR NAME. Your output should match the provided example output. You must also create a README explaining how to run your program AND answering the questions below. Furthermore, you must create a test script in Bash. Call this file test.sh – for this lab, your test script is only expected to run your program on 1 input.

You may only use the following imports:

```
from json import loads
from requests import get
from socket import gethostbyname
from subprocess import getstatusoutput
from sys import argv
```

Steps:

1. URL to IP

The first step is to look up the IP address of the specified domain name. You might recall we did DNS resolution in lab 1.

2. IP to Physical Address

Next, you'll need to get the physical address of the company who registered the IP address from step 1. `whois`, also from lab 1, should be able to accomplish this.

3. Physical Address to lat/lon

Now that you have the physical address, you need to use a web API to find the latitude and longitude for that location. Use the API provided by:
https://geocoding.geo.census.gov/geocoder/Geocoding_Services_API.html

4. lat/lon to Weather

Finally, with the latitude and longitude we can use a web API to pull the weather. Use the API provided by:
https://weather-gov.github.io/api/general-faqs

Questions:

1. What is an API?
2. What is a RESTful API? Were the APIs we used RESTful?
3. What is JSON? Did these APIs use JSON?
4. What is Bash? Have you ever used Bash?

Turn in:

Submit a tarball with the following:
- Your source code (in Python) called lab2.py
- Your test script (in Bash) called test.sh – this must test at least 1 sample input.
- Your README answering the above questions AND explaining how to run your program.

In case you have forgotten:
`tar -czvf lab2_YOURNAME.tar.gz *.py *.sh README`

Example output:

python3 lab2.py google.com

> Sunny, with a high near 75. North northwest wind 6 to 9 mph.

python3 lab2.py yahoo.com

> Mostly sunny. High near 75, with temperatures falling to around 69 in the afternoon. South wind 3 to 7 mph.

python3 lab2.py amazon.com

> A chance of rain showers before 5pm. Partly sunny. High near 53, with temperatures falling to around 51 in the afternoon. North northeast wind around 7 mph. Chance of precipitation is 40