Is it Essential? V1.0

May 5, 2020

[ ]:
```
##V1.0 goals: bring to full functionality,
```

[1]:
```
!pip install requests
import json
import requests
import warnings
warnings.filterwarnings('ignore')
```

```
Requirement already satisfied: requests in /opt/conda/lib/python3.7/site-
packages (2.12.4)
```

[6]:
```
def naic_lookup(code):
    yr = 2012
    url = f"http://naics.codeforamerica.org/v0/q?year={yr}&code={code}"
    response = requests.get(url)
    data = response.json()
    return data
def geocoding():
    key = "AIzaSyCIZ5647df1-qq4AVq5Jmt9uhLJWSLkmeo"
    address = input("Type in your address. ")
    params = { "key" : f"{key}", "address" : f"{address}"}
    url = "https://maps.googleapis.com/maps/api/geocode/json?"
    response = requests.get(url, params = params)
    one = response.json()
    latlng = (one['results'][0]['geometry']['location'])
    return latlng
def places_lookup(place):
    key = "AIzaSyCIZ5647df1-qq4AVq5Jmt9uhLJWSLkmeo"
    params = { "key" : f"{key}", "input" : f"{place}", "inputtype" :␣
 ↪"textquery", "language" : "english", "locationbias" : "ipbias"}
    url = f"https://maps.googleapis.com/maps/api/place/findplacefromtext/json?"
    response = requests.get(url, params = params)
    candidates = response.json()
    values = (candidates['candidates'])
    IDlist = []
    for placeid in values:
        IDlist.append(placeid['place_id'])
```

```
    return IDlist
def places_details(place_id):
    key = "AIzaSyCIZ5647df1-qq4AVq5Jmt9uhLJWSLkmeo"
    url = f"https://maps.googleapis.com/maps/api/place/details/json?
 ↪key={key}&place_id={place_id}&fields=name,business_status,formatted_address,opening_hours,n
    response = requests.get(url)
    results = response.json()
    return results
```

[18]:
```
search_select = input("Please type in what you will be searching by: 'keyword',␣
 ↪'NAIC code', or 'store name'. Type 'dict' to search essential businesses by␣
 ↪sector. Type 'exit' to quit. ")
while search_select != 'exit':
    if search_select == "keyword":
        with open ("EssentialBusinesses.txt", 'r') as f:
            search1 = input("Please input a keyword to search our list of␣
 ↪essential businesses. Type 'quit' to quit. ")
            search = search1.capitalize()
            for line in f:
                if (f"{search}") in line and ("SIC") in line:
                    print("")
                    print(line)
                    keepgoing = input("Is this the industry you were looking␣
 ↪for? y/n ")
                    if keepgoing == "y":
                        print(f"This industry is considered essential. A *␣
 ↪denotes that there may be some exceptions.")
                        break
                    else:
                        continue
            break
    elif search_select == "NAIC code":
        code = input("Enter a NAICS code, up to 4 digits. ")
        data = naic_lookup(code)
        industry = (data['title'])
        with open ('EssentialBusinesses.txt', 'r') as e:
            if (f"{code}") in e.read():
                essential = 'yes'
            else:
                essential = 'no'
        print(f"This NAIC code refers to: {industry}." )
        print(f"Essential: {essential}")
        break
    elif search_select == "store name":
        place = input("Please input the name of a store and the nearest one to␣
 ↪your address  as determined by IP bias will be returned.")
```

```
        place_id = places_lookup(place)[0]
        results = places_details(place_id)
        hours = results['result']['opening_hours']['weekday_text']
        businessstatus = results['result']['business_status']
        name = results['result']['name']
        print(f"{name} is currently {businessstatus}. If this result is␣
↪'CLOSED_TEMPORARILY', the business is likely nonessential. The new hours are:
↪ {hours}.")
        break
    elif search_select == 'dict':
        with open ('sectors.txt', 'r') as s:
            for line in s.readlines():
                print(line)
        sector = input("Type in one of the above sectors to bring up a list of␣
↪essential businesses in those sectors. ")
        with open ("EssentialBusinesses.txt", 'r') as d:
            for line in d:
                if line.startswith(sector):
                    print("")
                    print(line)
                    for line in d:
                        print (line)
                        if line.startswith('>'):
                            break

    else:
        print("That input wasn't recognized or the program ran into an␣
↪unspecified error. Please try again.")
        break
```

```
Please type in what you will be searching by: 'keyword', 'NAIC code', or 'store
name'. Type 'dict' to search essential businesses by sector. Type 'exit' to
quit. store name
Please input the name of a store and the nearest one to your address  as
determined by IP bias will be returned.Target
Target is currently OPERATIONAL. If this result is 'CLOSED_TEMPORARILY', the
business is likely nonessential. The new hours are: ['Monday: 7:00 AM - 9:00
PM', 'Tuesday: 7:00 AM - 9:00 PM', 'Wednesday: 7:00 AM - 9:00 PM', 'Thursday:
7:00 AM - 9:00 PM', 'Friday: 7:00 AM - 9:00 PM', 'Saturday: 7:00 AM - 9:00 PM',
'Sunday: 7:00 AM - 9:00 PM']
```

[ ]: