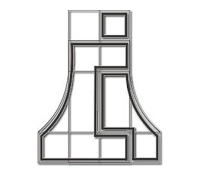


**Facultad de Ingeniería Civil**

**Ingeniero Topógrafo Geomático**



**Programación II**

**Ing.: Sebastián González Zepeda**

**“API de DENUE para la Localización de Negocios”**

**3°B**

**Andrea Rodríguez Santos**

**Juan Osvaldo García Corona**

**Coquimatlán, Colima; Fecha de Entrega: 29/Noviembre/2018**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Juan García Corona, Andrea Rodríguez Santos**

Facultad de Ingeniería Civil, [juanosvaldo\_garcia@ucol.mx](mailto:juanosvaldo_garcia@ucol.mx), [andrea\_rodriguezsantos@ucol.mx](mailto:andrea_rodriguezsantos@ucol.mx).

**Resumen**

En este proyecto se desarrolló un programa en el cual se pudiera observar en un mapa de folium con popup, todos los negocios contenidos en la base de datos del INEGI (API de DENUE), el cual arroja los establecimientos que están dentro de las coordenadas y radio que se indiquen. Para poder realizar este proyecto se utilizó el notebook de jupyter (Python), con apoyo de los módulos de urllib2, requests y folium, esto para poder llevar a cabo todos los procesos requeridos para lograr los objetivos establecidos.

**Abstract**

This project developed a program in which you could see on a map of folium with popup, all businesses contained in the database of the INEGI (API de DENUE), which throws the settlements that are within the coordinates and radius indicated. The notebook of jupyter (Python), was used to carry out this project with urllib2, requests and folium modules support, this is to be able to carry out all the processes required to achieve the objectives set.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Introducción**

Se preguntarán ¿qué es? o ¿qué será? API de DENUE, API de DENUE es un servidor del INEGI el cual proporciona información sobre más de 5 millones de establecimientos económicos a nivel nacional, por entidad federativa y municipio es decir de pequeños y grandes negocios, entonces API de DENUE es una base de datos que se está actualizando constantemente.

Puedes utilizar la API para crear aplicaciones que muestren la información directamente de las bases de datos del INEGI en el preciso momento en que se actualiza.

En este proyecto se pretende lograr extraer la información de la API que se requería (en este caso sería la información que yo necesito de ahí) de cada actividad o establecimiento económico, de cualquier tipo ya sean tiendas de ropa, consultorios, talleres, cualquier tipo de establecimiento que generen ingresos económicos, entonces lo primero que deberíamos hacer es ingresar a la página de API de DENUE (http://www.beta.inegi.org.mx/servicios/api\_denue.html), de la cual obtendremos la información necesaria, pero para poder solicitar la información , una vez dentro de ese link de la API, te diriges a la parte de “guía para desarrolladores” observas donde se encuentra token y le das clic ahí para solicitar un token, entonces el INEGI te proporciona un token, un token es una clave, una vez que se obtenga el token deberás de ingresarlo al final de la url como de la siguiente forma : http://www3.inegi.org.mx/sistemas/api/denue/v1/consulta/buscar/camiones/21.85717833,102.28487238/250/[aquí va tu Token].

Esa url que se genera la debes de ingresar al Python con el módulo de urllib2 la cual permite leer lo que contiene dicha url, después con el módulo de requests te apoyaras para mandar a pedir la información y te genere un json. La extracción de la información del json se puede hacer de varias formas pues mandara a llamar la información que requiere declarando variables y por su índice, o creando un bucle el cual te simplificara el proceso, sin necesidad

de estar mandando imprimir cada variable. Una vez que extrajiste la información que requieres para mapear en folium, que en este caso necesitábamos las coordenadas y el nombre del establecimiento.

**Metodología experimental**

El servidor del INEGI nos proporciona datos en formato Json, entonces una vez que obtienes esta información se procede a utilizar la consola de canopy Python o el notebook de jupyter para trabajar y poder extraer la información ahí. Entonces para poder desarrollar el proyecto se utilizaron 3 módulos los cuales son: “urllib2, requests y folium”.

Realiza una consulta de todos los establecimientos que cumplan las condiciones definidas.

Parámetros de entrada:

Condición

Palabra(s) a buscar en el nombre del establecimiento, razón social, calle, colonia, clase de la actividad económica, entidad federativa, municipio y localidad.

Para buscar todos los establecimientos se deberá ingresar la palabra "todos".

Coordenadas

Par de coordenadas que definen el punto en el mapa a partir del cual se hará la consulta alrededor. El formato de las coordenadas es latitud y longitud en geográficas decimales.

Distancia

Cantidad de metros a partir de las coordenadas que definen el radio de búsqueda. La distancia máxima es de 5 000 metros.

* Este es el código que manejamos en equipo.

import urllib2

import folium

import requests

lat= float(raw\_input("Latitud: ")) #esta es una peticion que le pedira al usuario, que en este caso es latitud

lon= float(raw\_input("Longitud: ")) #esta es una segunda peticion que le pedira al usuario

radio=int(raw\_input("Radio: ")) #de igual manera esta es una peticion que le pedira al usuario ingresar el radio

token= "c6c1fff0-ff8f-4f6e-8917-0460fd84d434"#Esta clave me la proporciono el INEGI

url= "http://www3.inegi.org.mx/sistemas/api/denue/v1/consulta/buscar/todos/"

url+= str(lat) + ","+str(lon)+"/"+str(radio)+"/"+token

url

#importar requests

response= requests.get(url) #este comando devolvera la llamada o peticion de response

results= response.json()

f = results # f es una variable que contiene el resultado, es decir que contiene la informacion del json

s= len(f) #se declara una nueva variable en donde len , es la longitud, es decir todo lo que contiene f

#Se crea el objeto mapa, con el modulo de folium

mexico = folium.Map(location=[22, -99],zoom\_start=5)

folium.Marker([lon, lon], popup= "centro",icon = folium.Icon(color ='red')).add\_to(mexico)

for i in range (s):

folium.Marker(f[0]["Latitud"], f[1]["Longitud"]), f[2]["Nombre"],folium.Icon((color('green'))).add\_to(mexico)

mexico

1. El código que se mostrara a continuación tiene un cambio en el código le aplique una instrucción donde le pido que me grafique la latitud y longitud, una vez que ingreso los datos junto con el radio me da un diferente resultado que en el primer código que manejamos del equipo

import urllib2

import folium

import requests

import matplotlib.pyplot as plt

lat= float(raw\_input("Latitud: ")) #esta es una peticion que le pedira al usuario, que en este caso es latitud

lon= float(raw\_input("Longitud: ")) #esta es una segunda peticion que le pedira al usuario

radio=int(raw\_input("Radio: ")) #de igual manera esta es una peticion que le pedira al usuario ingresar el radio

token= "c6c1fff0-ff8f-4f6e-8917-0460fd84d434"#Esta clave me la proporciono el INEGI

url= "http://www3.inegi.org.mx/sistemas/api/denue/v1/consulta/buscar/todos/"

url+= str(lat) + ","+str(lon)+"/"+str(radio)+"/"+token

url

#importar requests

response= requests.get(url) #este comando devolvera la llamada o peticion de response

results= response.json()

f = results # f es una variable que contiene el resultado, es decir que contiene la informacion del json

s= len(f) #se declara una nueva variable en donde len , es la longitud, es decir todo lo que contiene f

#Se crea el objeto mapa, con el modulo de folium

mexico = folium.Map(location=[22, -99],zoom\_start=60)

folium.Marker([lon, lon], popup= "centro",icon = folium.Icon(color ='red')).add\_to(mexico)

plt.plot(lon,lon)

plt.show()

El resultado al que llegue se muestra en este grafico a continuación:

Latitud: 27.2722

Longitud: -101.9458

Radio: 300



No muestra las direcciones ni los establecimientos ya que en la instrucción era para graficar la latitud y longitud es decir puedo concluir que con la instrucción mostrada

plt.plot(lon,lon)

plt.show()

no sirve de nada para la graficacion de los datos que estamos utilizando

**Manejo de Datos**

Los datos que se manejaran en este proyecto son de una API DENUE en el cual los podrán consultar en la siguiente dirección electrónica:

<http://www.beta.inegi.org.mx/servicios/api_denue.html>

Los cuáles serán utilizados en el proceso de realización del código establecido para este proyecto.

El total de datos están relacionados con los comercios económicos que generan ingresos los cuales se establece que se cuenta con más de 5 millones de establecimientos de este tipo a nivel nacional

**Resultados**

Se espera tener los datos mapeados, es decir en el código su proceso será introducir datos como lo es es latitud y longitud junto con un radio en el cual establecidos esos datos el programa tendrá la función de correr esa orden y por consecuencia se daráSerán mapeados y aparecerán por iconos en el cual al tocar mostrara el tipo de establecimiento, nombre, latitud y longitud.

**Conclusiones**

**En** este nuevo código me di a la tarea de mostrar que una instrucción cambiar por completo el análisis de los datos aunque los datos sean los mismos para cada programa por ejemplo en el primer código que se muestra es el que trabajamos en equipo para el proyecto integrador y el segundo código cambie algunas variables y la instrucción de graficar los datos y me di cuenta que cambia mucho la graficacion de los datos y no te los da como en el primer código que se muestra los negocios que generan ingresos, en el primer código el único problema que nos arrojó es que algunas variables afectaron para que el mapa no saliera con el nombre de los negocios mapeados y es por eso que no se muestra la imagen en cambio en el código tuneado se obtuvo el resultado mostrado en una imagen jpg.

**Bibliografías**

* <http://www.beta.inegi.org.mx/servicios/api_denue.html>