**LEYENDO ARCHIVOS JSON**

Para trabajar con objetos json, se realizan algunos imports:

from urllib.request import Request, urlopen

import json

el primer paso es crear un objeto de tipo Request:

URL = Request('http://globalmentoring.com.mx/api/personas.json')

Luego, se pueden agregar los headers necesarios para la petición:

URL.add\_header(

    'User-Agent', 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:78.0) Gecko/20100101 Firefox/78.0')

Ahora se puede utilizar la siguiente línea:

respuesta = urlopen(URL)

*# print(respuesta) # htt.client.HTTPResponse object*

Para leer el cuerpo de la respuesta:

cuerpo\_respuesta = respuesta.read()  *# cadena en binario*

esto retorna una cadena en bytes que es necesario decodificar

json\_respuesta = json.loads(cuerpo\_respuesta.decode('utf-8'))

ahora el objeto **json\_respuesta** es un diccionario al cual podemos acceder, por ejemplo:

{

'mensaje': 'exitoso',

'personas': [{'edad': '28', 'nombre': 'Juan Perez'},

{'edad': '32', 'nombre': 'Karla Gomez'},

{'edad': '35', 'nombre': 'Carlos Lara'},

{'edad': '22', 'nombre': 'María Esparza'},

{'edad': '40', 'nombre': 'Pedro Santos'}],

'total': 5

}

for e in json\_respuesta['personas']:

    print(e)

OTRO EJEMPLO

Esta sintaxis podría funcionar en todos los casos:

Defino en una línea el objeto de class request. Indicando la url y los headers necesarios para no levantar el error 403 y 406:

request = Request(urlAPI, headers={

                  'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:78.0) Gecko/20100101 Firefox/78.0'})

*# urllib.request.Request*

Ahora genero el objeto de class Httplclient con el método **urlopen()**

httpclient = urlopen(request)

*# HttpClient*

Leo el httpclient, generando un objeto de class bytes:

bytess = httpclient.read()

Por ultimo se conviert a diccionario mediante:

json\_respuesta = json.loads(bytess.decode('utf-8'))