

MOD 6 DAY 3 APIs and Geospatial Data Visualization

Tuesday, October 17, 2023 6:12 PM

1 min in we learn how to make a geoapify account

We make a config.py file and add out api key from geoapify

The slide features a decorative background of light blue and white geometric shapes. In the center-left, there's a graphic of three circles connected by lines, forming a small network or tree structure. To the right of this graphic, the text 'Python APIs: Day 3' is displayed in a large, bold, sans-serif font. Below this title is a black horizontal bar containing the text 'Data Boot Camp' and 'Lesson 6.3'. To the right of this text is a white icon of a document with a downward arrow, representing a download or lesson icon. On the far left of the slide, there is very small, faint text that reads '© 2022 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.' On the far right, there is a vertical toolbar with icons for navigation, search, and other presentation functions.

Class Objectives

By the end of today's class, you will be able to:

-  Use the Geoapify API to obtain information about geographic areas.
-  Use the Census API to get population counts, average income, and poverty rates of cities.

Visually represent banking and income data by using GeoViews.

2

30



Activity 1

01-Ins_Geoapify_Geocode

<https://apidocs.geoapify.com/>

We learn how to use a api key

We import api key

We build an end point url

How to get api about map data in url curl js node.js python php java 00:15:00

json dumps

How to get lat and lon (latitude and longitude)



Instructor Demonstration

Geoapify Geocode

6
30
^
v
C
D
S

Geoapify Geocode



Geoapify API geocoding feature converts addresses into latitudinal and longitudinal coordinates.

This process is known as **geocoding**.

Many applications require locations to be formatted in terms of latitude and longitude.

```
"distance": 0,  
"formatted": "Sydney, NSW, Australia",  
"lat": -33.8698439,  
"lon": 151.2082848,  
"place_id": "512088e244aae662405903907f0b57ef40c0f00101f901f5bc570000000000c00208",  
"rank": {  
    "confidence": 1,  
    "confidence_city_level": 1,  
    "importance": 0.8245908962989684,  
    "match_type": "full_match",  
    "popularity": 7.332617625087506
```

7
30
^
v
C
D
S

Activity 2

We set parameters on the location api

We use placea api on geoapify

We run a request from parameters

How to print the name and properties of the first restaurants that appear in search .

0:26:00

How to navigate within APIs

How to read parameters activity 3

The slide features a dark background with diagonal light gray stripes. In the center is a white laptop icon with a teal screen displaying a map. Below the laptop, the text "Instructor Demonstration" is written in a large, bold, teal font. A thin horizontal line separates this from the title "Geoapify Places" in a large, bold, white font. On the right side of the slide, there is a vertical toolbar with various icons: a bookmark, a grid, a number 9, a number 30, arrows for navigation, a circular arrow, a refresh symbol, a magnifying glass, and a question mark.

Activity 3

51:00

03-Stu_Geoapify_Drills

How to use multiples urls and api

We set the parameters

We run request

json.dump

We use geoapify to query locations to search keywords of businesses

bias apify api

How to use conditions as parameters to search maps

```
1 # Geoapify Drills
2
3 In this activity, you will perform API calls to the Geoapify places and geocoding endpoints.
4
5 ## Instructions
6
7 * Complete each of the six drills articulated in the code provided in [geoapify_drills.ipynb](Unsolved/geoapify_drills.ipynb). Feel
  encouraged to look back at the previous examples, but know that you will have to consult the Geoapify API documentation.
8
9 ### Hints
10
11 * See the [Geoapify Geocoding Documentation](https://apidocs.geoapify.com/docs/geocoding/forward-geocoding/#about).
12
13 * See the [Geoapify Places Documentation](https://apidocs.geoapify.com/docs/places/#about).
14
15 ---
16
17 © 2022 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.
18
```

Activity 4

Another way on traverse json

We can use pandas to

...

How to get all rows from columns

```
### try-except  
### loc
```

How to read csv

How to search with parameters

```
### what is pandas
```

Json looks like a python dictionary

We pull data from a column

We access multiple elements from a json 1:13:00

```
### what is a param
```

We create a column with a name and add a value

```
### (. 0. ) mean the element 0 .. So basically the index 0
```





Instructor Demonstration

Exploring Nearest Restaurants in Madrid

Another Way to Traverse JSONs

The Pandas method `iterrows()` returns an index number and the contents of each row.

- Rows can be accessed using `row['column label']`.

With each iteration, the keyword is overwritten.

The `get()` method retrieves results

- If the result exists, the value is retrieved.
- If not, then `"None"` is stored.
- Similar to `try-except`.

```
# Iterate through the types_df DataFrame
for index, row in types_df.iterrows():

    # Get the ethnicity type from the current DataFrame's row
    ethnicity = types_df.loc[index, "ethnicity"]
    # Add the current ethnicity type to the parameters
    params["categories"] = f"catering.restaurant.{ethnicity}"

    # Make an API request using the params dictionary
    restaurant = requests.get(base_url, params=params)

    # Convert the API response to JSON format
    restaurant = restaurant.json()

    # Grab the first restaurant from the results and store the details in the DataFrame
    try:
        types_df.loc[index, "name"] = restaurant["features"][0]["properties"]["name"]
        types_df.loc[index, "address"] = restaurant["features"][0]["properties"]["address_line2"]
        types_df.loc[index, "distance"] = int(restaurant["features"][0]["properties"]["distance"])
    except (KeyError, IndexError):
        # If no restaurant is found, set the restaurant name as "No restaurant found".
        types_df.loc[index, "name"] = "No restaurant found"
        # Set the distance column value to np.nan to allow sorting values
        types_df.loc[index, "distance"] = np.nan

    # Log the search results
print(f"nearest {types_df.loc[index, 'ethnicity']} restaurant: {types_df.loc[index, 'name']}")
```

The `try-except` clause is used with `loc` to store the responses.

Activity 5

05-Evr_Exploring_Airports

Review of what we just save

1:37:00

1:51:00

Import dependecenes

We add colums we are interested

We import url of city info we want

params

We search for airports information

```
> sigmanationn > Desktop > Analysis Project > REPOS > UCB-VIRT-DATA-PT-05-2023-U-LOLC > 01-Lesson-Plans > 06-Python-APIs > 3 > Activities > 05-Evr_Exploring_Airports >
  1  # Exploring Airports in Australia
  2
  3 In this activity, you'll be tasked with obtaining information about some Australian airports. You'll be given a list of cities, and you'll
     need to use the Geoapify Geocoding API and Geoapify Places API to obtain the airports' information.
  4
  5 ## Instructions
  6
  7 * Using [airports.ipynb](Unsolved/airports.ipynb) as a starting point, use the Geoapify Geocoding API, the Geoapify Places API, and Python
     to create a script that retrieves information of some Australian airports in each of the cities found in [Cities.csv](Resources/Cities.csv).
  8
  9 * You should create a DataFrame that has to contain each of the following columns:
 10
 11 * `Lat`
 12
 13 * `Lon`
 14
 15 * `Airport Name`
 16
 17 * `IATA Name`
 18
 19 * `Airport Address`
 20
 21 * `Distance`
 22
 23 * `Website`
 24
 25 * You should review the Geoapify API documentation to identify where is this information located in the JSON response.
 26
 27 * [Geocoding API docs](https://apidocs.geoapify.com/docs/geocoding/forward-geocoding/#about)
 28
 29 * [Places API docs](https://apidocs.geoapify.com/docs/places/#about)
 30
 31 * Finally, save the airports's informations DataFrame as a CSV file called `Airport_Output.csv`.
 32
```



Activity: Exploring Airports in Australia

In this activity, you will obtain information for the airports in major Australian cities.

Suggested Time:

20 minutes

Activity: Exploring Airports in Australia

Instructions

With `airports.ipynb` as your starting point, use the Geoapify Geocoding API, the Geoapify Places API, and Python to create a script that retrieves information of some Australian airports in each of the cities found in `Cities.csv`.

Your final notebook file should contain each of the following headers:

`Lat, Lon, Airport Name, IATA Name, Airport Address, Distance, Website`

Hints

You will need to obtain the latitude (`lat`) and longitude (`lon`) of each airport prior to sending it through the Geoapify Places API to obtain the information.

When using the Geoapify Places API, make sure to use the "airport" category to ensure that the data received is for an airport in the city.

Use a `try-except` to identify airports for which there are missing data..

Activity 6

06-Evr_Geoviews_Maps

GeoView Maps

2:00:00

How to install geo view

(you must install by yourself)

View presentation, download

Geoviews.org

How to turn off warning

dictionary

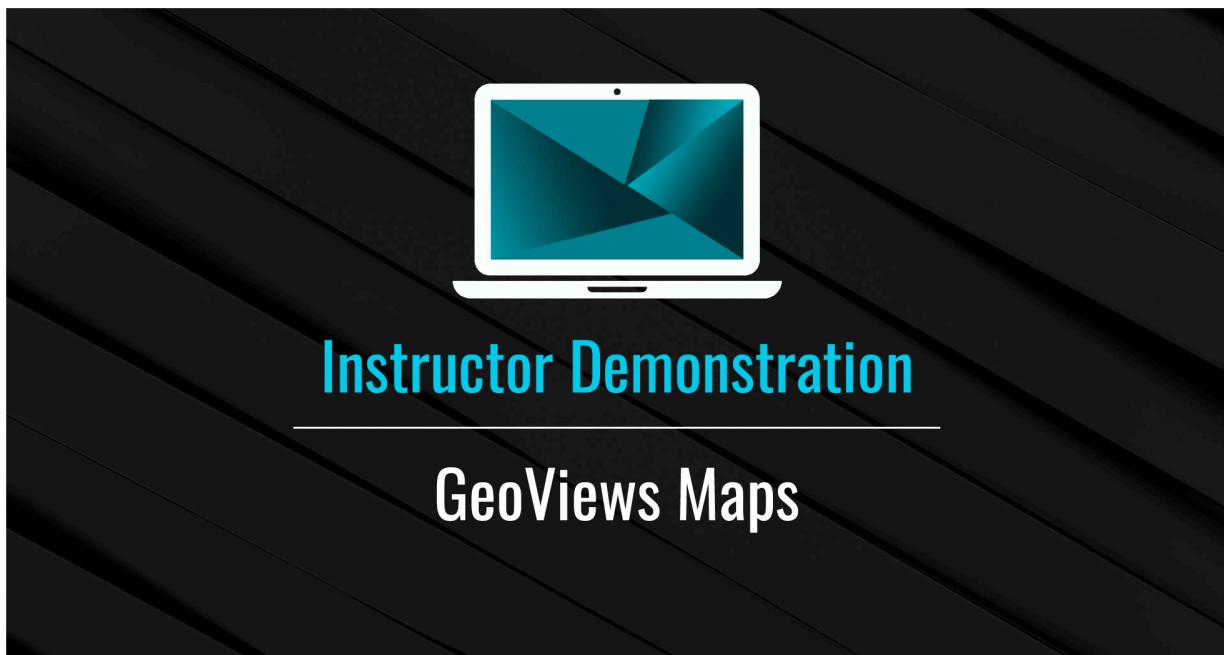
data frame

How to display maps

We change the point colours

geo = true

tiles = osm means map tiles (blocks)



Activity 7

Stu airport maps

2:26:00

We plot data into pandas

Read head of csv

We customize an dconfisur maps by color size title scale frame and scale

```
Users > sigmanationn > Desktop > Analysis Project > REPOS > UCB-VIRI-DATA-PT-05-2023-U-LOLC > 01-Lesson-Plans > 06-Python-APIs > 3 > Activities > 07-Stu_Airports.ipynb
1 # Australian Airports Map
2
3 In this activity, you will create a map based on Australian airports information that you generated before using the Geoapify API.
4
5 ## Instructions
6
7 1. Load airports' data into a Pandas DataFrame.
8
9 2. Create a simple map using GeoViews by adding a point per airport and setting a fixed size at your convenience.
10
11 3. Use GeoViews to create a custom map by setting values for color, size, and a title different than OSM.
12
13 > **Hint:** From the column values, you should decide what columns can be used to set a different color and size for each city.
14
15 ---
16
17 © 2022 edX Boot Camps LLC. Confidential and Proprietary. All Rights Reserved.
18
```



Activity: Australian Airports Map

In this activity, you will create a map based on airport information.

Suggested Time:

15 minutes

Activity 8

2:28:00

08-Ins_Census

Census.gov/developers

We use a wrapper

what is a wrapper

what is a instance

We create a instance



Instructor Demonstration

U.S. Census Demo

U.S. Census API

Instructions:

- Obtain an API key from <http://www.census.gov/developers/>.
- Run `pip install census` in your environment.
- The wrapper provides an easy way to retrieve data from the 2013 Census based on zip code, state, district, or county.
- Each census field (for example, Poverty Count, Unemployment Count) is denoted with a label like `B201534_10E`.
- The results are then returned as a list of dictionaries, which can be immediately converted into a DataFrame.

How We'll Use the U.S. Census API

- The `c.acs5.get` method grabs data on each field.
- `Poverty Rate` is divided by `Total Population` to evaluate `Poverty Rate`.
- The U.S. Census does not explicitly calculate `Poverty Rate`.

```
census_data = c.acs5.get(["NAME", "B19013_001E", "B01003_001E", "B01002_001E",
                         "B19301_001E",
                         "B17001_002E"], {'for': 'zip code tabulation area:*'})

# Convert to DataFrame
census_pd = pd.DataFrame(census_data)

# Column Reordering
census_pd = census_pd.rename(columns={"B01003_001E": "Population",
                                      "B01002_001E": "Median Age",
                                      "B19013_001E": "Household Income",
                                      "B19301_001E": "Per Capita Income",
                                      "B17001_002E": "Poverty Count",
                                      "NAME": "Name", "zip code tabulation area": "zipcode"})

# Add in Poverty Rate (Poverty Count / Population)
census_pd["Poverty Rate"] = 100 * \
    census_pd["Poverty Count"].astype(
        int) / census_pd["Population"].astype(int)

# Final DataFrame
census_pd = census_pd[["zipcode", "Population", "Median Age", "Household Income",
                      "Per Capita Income", "Poverty Count", "Poverty Rate", "Name"]]
```


Activity 9

09-Evr_Banking_Deserts

why and how we import dependemceus

list all data types float sting var

```
5 > signin@linn > Desktop > Analysis Project > REPOS > UCB-VIRI-DATA-PT-05-2023-U-LOLC > 01-Lesson-Plans > 06-Python-APIs > 3 > Activities > 09-Evr_Banking_Deserts > 09-Evr_Banking_Deserts.ipynb
1 # Banking Deserts
2
3 In this activity, your task is to understand if there is a relationship between poverty, age, population, and the number of banks in a given area. To help, we've provided you with Census data for every U.S. zip code. You will also visualize this data set using GeoViews.
4
5 ## Instructions
6
7 1. Retrieve data from the U.S. Census using the Census Python library and the preconfigured labels.
8
9 2. Load the [zip_bank_data.csv](Resources/zip_bank_data.csv) file into a DataFrame. Next, use Pandas to merge this data set with the Census data that you retrieved along the zip code.
10
11 3. Use GeoViews to create a poverty rate map.
12
13     * Use the "Poverty Rate" column to set the point's size. Recall using the `scale` parameter to modify the size appearance.
14
15     * Use the "Zipcode" column to set the point's color.
16
17     * Read the HoloViews documentation and learn how you can use the `hover_cols` parameter to add additional information to the tooltip of a point. Add the "Address" and the "Bank Count" columns.
18
19 4. Compute and print the summary statistics for "Poverty Rate", "Bank Count", and "Population".
20
21 5. Create a scatter plot with linear regression for **bank count** vs. **poverty rate**.
22
23     * Be sure `NaN` values are dropped from the DataFrame.
24
25     * Plot the data points.
26
27     * Plot the linear regression line.
28
29     * Print the R2 value.
30
31 6. Analysis. Write a sentence describing your findings. Were they what you expected? What other factors could be at play?
```



Activity: Banking Deserts

In this activity, you will create a data visualisation to understand how prominent the banking desert phenomenon truly is.

Suggested Time:

20 minutes

28

Activity: Banking Deserts Heatmap

Instructions

Using `GeoViews`, create a poverty rate map:

- Use the "Poverty Rate" column to set the point's size. Recall using the `scale` parameter to modify the size appearance.
- Use the "Zipcode" column to set the point's colour.
- Read the HoloViews documentation and learn how you can use the `hover_cols` parameter to add additional information to the tooltip of a point. Add the "Address" and the "Bank Count" columns.

Print the summary statistics for `Unemployment`, `Bank_Count`, and `Population`.

Create a scatter plot with linear regression for `Bank_Count` vs. `Unemployment_Rate`.

Plot the data points.

Plot the linear regression line.

Print the R^2 value.

Write a sentence describing your findings. Were they what you expected? What other factors could be at play?

