

▼ Getting Data with APIs

Sonal Joshi

```
# Installing packages which are not present by default
#!pip install praw geojson
```

▼ (Que 1) We chose OpenStreetMap API.

- OpenStreetMap (OSM) is a free and open-source map of the world which contains geographic data such as roads, buildings, etc.
- The OpenStreetMap API is a web service that provides read and write access to the OSM data.
- Overpass API is a read-only API that allows users to query OSM data in a more flexible and efficient way than the OSM API.
- Overpass API provides a way to extract a custom selection of data from OSM based on different criteria like geographical region, tags, date/time, and much more. Overpass API supports multiple output formats, including XML, JSON, and GeoJSON.

```
# Importing the required packages
import requests
import json
import geojson
```

▼ (Que 2) We will get the details about road network, buildings and other features for New Brunswick using Overpass API and query parameters.

Access is public for reading or querying data using overpass, so there is no need for any API key. To modify data, an account would be required to do so.

```
# Define the API URL
url = "http://overpass-api.de/api/interpreter"

# Define the query to obtain details about the road network, buildings, and other features in NB
query = """
[out:json];
area["name"="New Brunswick"]->.boundaryarea;
(
  way["highway"](area.boundaryarea);
  relation["highway"](area.boundaryarea);
  way["building"](area.boundaryarea);
  relation["building"](area.boundaryarea);
);
out center;
"""
```

```
# Send the query to the Overpass API to get response
response = requests.get(url, params={'data': query})
```

```
# Response variable returns the response code like 200, 400, 404, 504 etc. Ideally it should be 200.
response.json()
```

```

nodes : [3018710885,
3018710886,
3018710887,
3018710888,
3018710889,
3018710890,
3018710891,
3018710892,
9002327436,
4425308127],
'tags': {'bicycle': 'yes',
'highway': 'footway',
'incline': 'up',
'name': 'Sacred Path',
'surface': 'brick'}},
{'type': 'way',
'id': 297971592,
'center': {'lat': 40.4868402, 'lon': -74.4373561},
'nodes': [3018620943, 3018710894, 3018710896, 3018710895],
'tags': {'highway': 'path'}},
{'type': 'way',
'id': 297971593,
'center': {'lat': 40.4866775, 'lon': -74.437748},
'nodes': [3018710896,
3018710897,
3018710903,
3018710898,
3018710899,
3018710900,
3018710901,
3018620937],
'tags': {'highway': 'path'}},
{'type': 'way',
'id': 297971595,
'center': {'lat': 40.4867271, 'lon': -74.4371288},
'nodes': [3018710894, 3018710904, 3018710929],
'tags': {'highway': 'path'}},
...]}

```

response

<Response [200]>

- ▼ (Que 3) To get data in geojson, out body geom and [out:json] is passed in params. For text, .text method is used.

```

# Query params to get output in geojson
geojson_query = ""
[out:json];
area["name"="New Brunswick"]->.boundaryarea;
(
  way["highway"](area.boundaryarea);
  relation["highway"](area.boundaryarea);
  way["building"](area.boundaryarea);
  relation["building"](area.boundaryarea);
);
out body geom;
""

```

```

# Send the query to the Overpass API
geojson_response = requests.get(url, params={'data': geojson_query})

```

geojson_response

<Response [200]>

```

# To convert the response to text. This can be used with json or XML response too.
geojson_response.text

```

```

{'\n "version": 0.6,\n "generator": "Overpass API 0.7.60.2 e4cf9f7a",\n "osm3
s": {\n "timestamp_osm_base": "2023-05-02T19:35:23Z",\n "timestamp_areas_b
ase": "2023-04-25T08:53:24Z",\n "copyright": "The data included in this docum
ent is from www.openstreetmap.org. The data is made available under ODbL."\n
},\n "elements": [\n\n{\n "type": "way",\n "id": 11648485,\n "bounds": {\n
"minlat": 40.5053361,\n "minlon": -74.4518320,\n "maxlat": 40.5056240,\n
"maxlon": -74.4515126\n },\n "nodes": [\n 4394181032,\n 6910757547,\n
104131645\n ],\n "geometry": {\n { "lat": 40.5053361, "lon": -74.4515126

```

Writing the outputs to different file formats.

▼ (Que 4) Getting filtered data.

We have filtered data using:

- City - Fredericton
- Street Name - King Street, Regent Street etc.
- Tags - Highway, One Way etc.

```
# API query params for filtering on basis of city in Fredericton
city_query = ""
[out:json];
area["name"="Fredericton"]->.boundaryarea;
way(area.boundaryarea)["highway"];
out center;
""
```

```
# Send the request to the Overpass API and get the response
city_response = requests.get(url, params = {"data": city_query})
```

```
# Converts the city_response object to json format
city_response.json()
```

```

    'lit': 'no',
    'name': 'Rue Saint-Roch',
    'sidewalk': 'none',
    'surface': 'asphalt'}},
...]}

```

▼ Filtering data for a particular street in Freddy

```

# Street query params for Fredericton
street_query = ""
[out:json];
area[name="Fredericton"]->.fd;
node(area.fd)["addr:street"]="King Street";
out;
""

# Send the HTTP request to the Overpass API and retrieve the JSON response
street_response = requests.get(url, params={'data': street_query})

# Returns the response in json format.
street_response.json()

{'version': 0.6,
 'generator': 'Overpass API 0.7.60 f4c14d41',
 'osm3s': {'timestamp_osm_base': '2023-05-02T19:35:23Z',
 'timestamp_areas_base': '2023-05-02T15:01:55Z',
 'copyright': 'The data included in this document is from www.openstreetmap.org. The data is made available under ODbL.'},
 'elements': [{'type': 'node',
 'id': 410783668,
 'lat': 45.9607648,
 'lon': -66.6404447,
 'tags': {'addr:city': 'Fredericton',
 'addr:housenumber': '551',
 'addr:postcode': 'E3B 3T9',
 'addr:province': 'NB',
 'addr:street': 'King Street',
 'addr:unit': 'C.P. 414',
 'amenity': 'bank',
 'atm': 'yes',
 'brand': 'National Bank',
 'brand:en': 'National Bank',
 'brand:fr': 'Banque Nationale',
 'brand:wikidata': 'Q634298',
 'drive_through': 'no',
 'name': 'National Bank',
 'name:en': 'National Bank',
 'name:fr': 'Banque Nationale',
 'official_name': 'National Bank of Canada',
 'official_name:en': 'National Bank of Canada',
 'official_name:fr': 'Banque Nationale du Canada',
 'opening_hours': 'Mo 09:30-16:00, Tu 09:30-17:00, We 09:30-17:00, Th 09:30-19:00, Fr 09:30-17:00',
 'operator': 'National Bank',
 'phone': '5064588129'}},
 {'type': 'node',
 'id': 410785666,
 'lat': 45.9625772,
 'lon': -66.6441891,
 'tags': {'addr:city': 'Fredericton',
 'addr:housenumber': '379',
 'addr:street': 'King Street',
 'amenity': 'restaurant',
 'cuisine': 'pizza',
 'name': 'Jack's Pizza',
 'opening_hours': 'Su-Tu 10:00-23:00, We-Sa 10:00-03:00'}},
 {'type': 'node',
 'id': 2330169379,
 'lat': 45.9620139,
 'lon': -66.643459,
 'tags': {'addr:street': 'King Street',
 'amenity': 'cafe',
 'check_date': '2023-02-05',
 'name': 'Coffee and Friends',
 'opening_hours': 'Mo-Fr 09:00-18:00, Sa 10:00-18:00',
 'wheelchair': 'yes'}},
 {'type': 'node',
 'id': 2624615357,
 'lat': 45.9605764,
 'lon': -66.6409606,
 'tags': {'addr:city': 'Fredericton',
 'addr:housenumber': '546',

```

▼ Filtering data for json data with all streets or roads having highway in their tags

```
# Query params for getting highway details in Fredericton
highway_query = """
[out:json];
area[name="Fredericton"]->.nb;
way(area.nb)["highway"];
out tags;
"""

# Define the API parameters
params = {
    "data": highway_query,
    "out": "tags",
}

# Send the HTTP request to the Overpass API and retrieve the response
highway_response = requests.get(url, params=params)

# Returns the list in json format.
highway_response.json()
```

▼ (Que 5) Streaming APIs: Collect Twitter or any other streaming API data for 1 minute on your favourite topic and save it in a JSON file

```
import praw
import time
import warnings
```

```
# Filter out the warning message on console
warnings.filterwarnings("ignore", category=UserWarning, module="praw")
```

```
# Create a Reddit instance
reddit = praw.Reddit(client_id='####', client_secret='####', user_agent='gge/1.0 (sonaljo13@gmail.com)', check_for_async=False)

# Set the subreddit to fetch comments from
subreddit = reddit.subreddit('AskReddit')
```

```
# Create an empty list to store the comments
comments = []
```

```
# Defining the 1 minute time for live streaming
end_time = time.time() + 60
```

```
# Using the reddit praw methods to stream the live comments from AskReddit group
for comment in subreddit.stream.comments(skip_existing=True):
```

```
    comment_data = {
        'body': comment.body,
        'author': comment.author.name,
        'created_utc': comment.created_utc,
        'subreddit': comment.subreddit.display_name,
    }
```

```
    print(comment_data)
    comments.append(comment_data)
```

```
# Breaks the loop when the 60 seconds condition is met
if time.time() >= end_time:
    break
```

```
{'body': 'You will never tell the difference. We can make an artificial human that's hot AF and moan to you. Better in bed tl
{'body': 'Failed to buy Apple stock in the 90s.', 'author': 'WeaselTeamSix', 'created_utc': 1683056402.0, 'subreddit': 'AskRe
{'body': 'forever Minecraft \U0001faf6', 'author': 'EntertainerOk2368', 'created_utc': 1683056402.0, 'subreddit': 'AskReddit
```

```
{'body': 'Panagios - we had been regulars, going in about once a month for a couple of years. One morning, pre Covid, we we'}
{'body': 'A sandwich short of a picnic.\n\nHis elevator doesn't go to the top floor.\n\nGot a few stitches dropped in her kni'}
{'body': 'Gen x here. The fashion industry is ridiculous and a waste of time & money', 'author': 'astupidfckingname', 'create'}
{'body': 'Sounds like you would love ESO. You should look into it. Is an elder scrolls mmo and it's by far the best MMO I've'}
{'body': 'Guess I become the priest to the emo church', 'author': 'hollowed_confessad', 'created_utc': 1683056424.0, 'subredc'}
{'body': 'They're supposedly lawyers coming up with these laws, too. I know it's not their job, but I would be pissed if I h'}
{'body': 'In my younger days (20-25) I had affairs with two girls at work, not at the same time, I used to email them and we'}
{'body': 'HAHA love that !!', 'author': 'baileybaby397', 'created_utc': 1683056425.0, 'subreddit': 'AskReddit'}
{'body': 'R/wooooooosh', 'author': 'MassiveLegend1864', 'created_utc': 1683056426.0, 'subreddit': 'AskReddit'}
{'body': '*Also* also add: Be aware that love doesn't always manifest in the same ways. Look up that love language stuff you'}
{'body': 'Nice', 'author': 'Ok-Suit-1713', 'created_utc': 1683056427.0, 'subreddit': 'AskReddit'}
{'body': 'Be irresponsible financially, not saving and accruing large debt. Takes literal years to get out of that and then j'}
{'body': 'I just wore a pair to Disney world and walked 8 miles in em, oh boy were they comfy! If you can get over the ugly,'}
{'body': 'I mean, Malazan is a 3 million word essay on compassion, loss, and honor. But yeah, nothing to do with human natu'}
{'body': 'Be obvious and don't wait for us to start it. I personally hate starting conversations but if someone asks me a que'}
{'body': 'That makes sense actually, but I would be against it anyway because I think the death penalty is barbaric', 'author'}
{'body': 'Final fantasy 3/6.', 'author': 'pikkdogs', 'created_utc': 1683056430.0, 'subreddit': 'AskReddit'}
{'body': 'Most people keep suicidal or violent thoughts to themselves, because they'll get involuntarily committed.', 'author'}
{'body': 'The Name of the Wind', 'author': 'diggl719', 'created_utc': 1683056431.0, 'subreddit': 'AskReddit'}
{'body': 'It is a developed country. But compared to most others US is "developing" country.', 'author': 'Whirlwind3', 'creat
```

```
# Prints the data in the variable comments
comments
```

```
'author': 'NoteBlock08',
'created_utc': 1683056426.0,
'subreddit': 'AskReddit'},
{'body': 'Nice',
'author': 'Ok-Suit-1713',
'created_utc': 1683056427.0,
'subreddit': 'AskReddit'},
{'body': 'Be irresponsible financially, not saving and accruing large debt. Takes literal years to get out of that and then
you're still behind.',
'author': 'ReaperVF',
'created_utc': 1683056427.0,
'subreddit': 'AskReddit'},
{'body': 'I just wore a pair to Disney world and walked 8 miles in em, oh boy were they comfy! If you can get over the
ugly, embrace the comfort 😊',
'author': 'Throwaway158265',
'created_utc': 1683056428.0,
'subreddit': 'AskReddit'},
{'body': 'I mean, Malazan is a 3 million word essay on compassion, loss, and honor. But yeah, nothing to do with human
nature. /s And has some of the most memorable characters of anything I've read. You seriously are going to say that
Fiddler and Hedge, Quick Ben and Kalam, Karsa Orlong, Kruppe, Tehol and Bugg, Anomander Rake, Tool, Hellian were all meh
characters? \n\nAnd the Cosmere and Malazan have worldbuilding to rival Tolkien, and the Cosmere is not even half done yet.
\n\nWoT book 1 intentionally mimicked LoTR to give readers in 1990 something familiar to draw them in. Jordan ripped a ton
off from Dune as well, but weaved it all together in a great way.\n\nSanderson's basic style serves the story better and
moves it along (most of the time) There are a minimum 12 more books that he has planned to wrap up the core Cosmere stuff.
Personally I think LoTR suffers from the prose. Not saying it's not good, but it gets in the way of the story (for me). To
quote "Martin" from Epic Rap Battles of History: "We don't need the back story to every fuck tree branch"\n\nTolkien is
foundational to the genre. No one can argue that. But is he the best? Certainly the most widely read, but also been on
shelves the longest. I loved the movies, but the books for me were kinda meh.',
'author': 'ShadowDV',
'created_utc': 1683056429.0,
'subreddit': 'AskReddit'},
{'body': 'Be obvious and don't wait for us to start it. I personally hate starting conversations but if someone asks me a
question I will definitely talk with them (unless I don't like them). \n\nI also personally would need a minute to
comprehend the fact a girl actually said something nice to/about me, as it would be a first lol. I am very good at fricking
up interactions with people, then severely regretting it instantly.',
'author': 'TrainGuy',
'created_utc': 1683056429.0,
'subreddit': 'AskReddit'},
{'body': 'That makes sense actually, but I would be against it anyway because I think the death penalty is barbaric',
'author': 'AggravatingToez',
'created_utc': 1683056429.0,
'subreddit': 'AskReddit'},
{'body': 'Final fantasy 3/6.',
'author': 'pikkdogs',
'created_utc': 1683056430.0,
'subreddit': 'AskReddit'},
{'body': 'Most people keep suicidal or violent thoughts to themselves, because they'll get involuntarily committed.',
'author': 'adrefofadre',
'created_utc': 1683056430.0,
'subreddit': 'AskReddit'},
{'body': 'The Name of the Wind',
'author': 'diggl719',
'created_utc': 1683056431.0,
'subreddit': 'AskReddit'},
{'body': 'It is a developed country. But compared to most others US is "developing" country.',
'author': 'Whirlwind3',
'created_utc': 1683056432.0,
'subreddit': 'AskReddit']}]
```

```
# Save the comments to a JSON file
with open('comments.json', 'w') as f:
    f.write(json.dumps(comments))
```

✓ 0s completed at 4:40 PM

