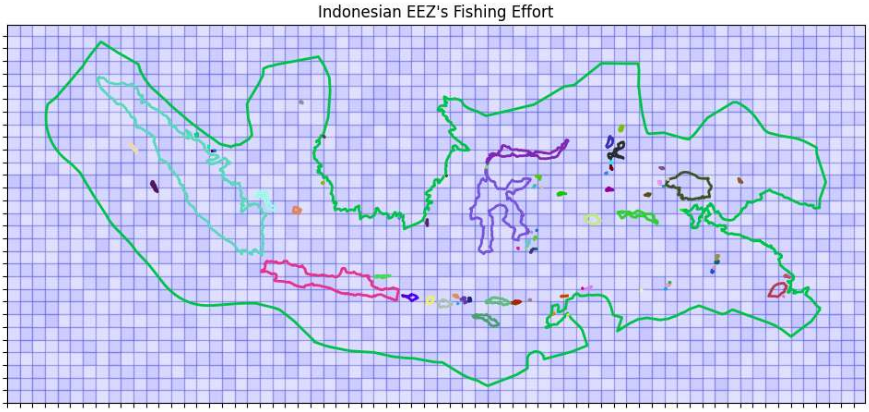
Get List of EEZs

```
import requests
import json


url = 'https://gateway.api.globalfishingwatch.org/v2/datasets/public-eez-areas/user-contex
headers = {
    'Authorization': 'Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCIsImtpZCI6ImtpZEtleSJ9.eyJk
}

response = requests.get(url, headers=headers)
response_data = response.json()
ids = []
iso3s = []
labels = []
for entry in response_data:
    ids.append(str(entry.get('id')))
    iso3s.append(entry.get('iso3'))
    labels.append(entry.get('label'))

indonesian_EEZ = []
print("id | iso3s | labels")
for i in range(len(ids)):
    print(f"{ids[i]} | {iso3s[i]} | {labels[i]}")

    if "Indonesia" in labels[i]:
        indonesian_EEZ.append(ids[i])
        indonesian_EEZ.append(iso3s[i])
        indonesian_EEZ.append(labels[i])


print(indonesian_EEZ)
```

```
8327 | KOR | South Korea
5693 | ESP | Spain
8346 | LKA | Sri Lanka
8355 | SDN | Sudan
8461 | SUR | Suriname
33181 | NOR | Svalbard
5694 | SWE | Sweden
8373 | SYR | Syria
8321 | TWN | Taiwan
8479 | TZA | Tanzania
8332 | THA | Thailand
8392 | TGO | Togo
8449 | NZL | Tokelau
8448 | TON | Tonga
8420 | TTO | Trinidad and Tobago
8381 | BRA | Trinidade
8382 | GBR | Tristan Da Cunha
48946 | FRA | Tromelin Island
8366 | TUN | Tunisia
5697 | TUR | Turkey
26523 | TKM | Turkmenistan
8405 | GBR | Turks and Caicos Islands
8326 | TUV | Tuvalu
5695 | UKR | Ukraine
8360 | ARE | United Arab Emirates
5696 | GBR | United Kingdom
8456 | USA | United States
33180 | USA | United States Virgin Islands
8467 | URY | Uruguay
8313 | VUT | Vanuatu
8433 | VEN | Venezuela
8484 | VNM | Vietnam
8319 | USA | Wake Island
8454 | FRA | Wallis and Futuna
8368 | ESH | Western Sahara
8353 | YEM | Yemen
['8492', 'IDN', 'Indonesia']
```

Get Fishing Effort in Region in a Timeframe

```
import json
import matplotlib.pyplot as plt
from matplotlib.patches import Polygon
import numpy as np

with open('res/geometry.geojson', 'r') as f:
    data = json.load(f)
def plot_coordinates(ax, geometry):
    if geometry['type'] == 'Polygon':
        for ring in geometry['coordinates']:
            x, y = zip(*ring)
            ax.plot(x, y, color=np.random.rand(3,), linestyle='-', linewidth=2)
    elif geometry['type'] == 'MultiPolygon':
        for polygon in geometry['coordinates']:
            for ring in polygon:
                x, y = zip(*ring)
                ax.plot(x, y, color=np.random.rand(3,), linestyle='-', linewidth=2)
    else:
        print("Unsupported geometry type:", geometry['type'])

fig, ax = plt.subplots(figsize=(12, 8))
if data['type'] == 'FeatureCollection':
    for feature in data['features']:
        geometry = feature.get('geometry')
        if geometry:
            plot_coordinates(ax, geometry)
elif data['type'] == 'Feature':
    geometry = data.get('geometry')
    if geometry:
        plot_coordinates(ax, geometry)
elif data['type'] == 'GeometryCollection':
    for geometry in data['geometries']:
        plot_coordinates(ax, geometry)
else:
    print("Unsupported GeoJSON type:", data['type'])


plt.title("Indonesian EEZ's Fishing Effort")
ax.set_aspect('equal')
grid_scale = 0.8
ax.grid(True, which='both', linestyle='--', linewidth=0.5, color='gray', alpha=0.5)
ax.set_xticks(np.arange(min(ax.get_xlim()), max(ax.get_xlim()), grid_scale))
ax.set_yticks(np.arange(min(ax.get_ylim()), max(ax.get_ylim()), grid_scale))
ax.set_xticklabels([])
ax.set_yticklabels([])

for x in np.arange(min(ax.get_xlim()), max(ax.get_xlim()), grid_scale):
    for y in np.arange(min(ax.get_ylim()), max(ax.get_ylim()), grid_scale):
        color = [0, 0, 1]
        alpha = np.random.uniform(0.1, 0.2)
        color.append(alpha)
        ax.add_patch(Polygon([[x, y], [x + grid_scale, y], [x + grid_scale, y + grid_scale]

plt.show()
```


Indonesian EEZ's Fishing Effort